BIBLIOTECA CENTRAL
PUCPR

JOEL LAROCCA NETO

CONTRIBUIÇÃO AO ESTUDO DE TÉCNICAS PARA SUMARIZAÇÃO AUTOMÁTICA DE TEXTOS

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como parte dos requisitos para obtenção do título de Mestre em Ciências.

Área de Concentração:

SISTEMAS INTELIGENTES

Orientador:

Prof. Dr. Celso Antônio Alves Kaestner

Co-orientador:

Prof. Dr. Alex Alves Freitas

Curitiba
2001

Larocca Neto, Joel
Contribuição ao Estudo de Técnicas para Sumarização Automática de Textos. Curitiba, 2001.
109 p.

Dissertação (Mestrado) – Pontifícia Universidade Católica do Paraná. Departamento de Informática.

1.Sumarização Automática de Textos 2.Recuperação de Informações I.Pontifícia Universidade Católica do Paraná. Centro de Ciências Exatas e de Tecnologia. Departamento de Informática II.t

Pontifícia Universidade Católica do Paraná
Centro de Ciências Exatas e de Tecnologia
Programa de Pós-Graduação em Informática Aplicada

ATA DA SESSÃO PÚBLICA DE DEFESA DE DISSERTAÇÃO DE MESTRADO DO PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA APLICADA DA PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ

DEFESA DE DISSERTAÇÃO Nº 48

Aos 18 dias do mês de fevereiro de 2002 realizou-se a sessão pública de defesa da dissertação "**Contribuição ao Estudo de Técnicas para Sumarização Automática de Textos**", apresentada por **Joel Larroca Neto** como requisito parcial para a obtenção do título de **Mestre em Ciências**, perante uma Banca Examinadora composta pelos seguintes membros:

	assinatura	parecer
Prof. Dr. Celso A. A. Kaestner PUCPR (Presidente)		APROVADO
Prof. Dr. Alex Alves Freitas PUCPR	Alex Alves Freitas	Aprovado
Prof. Dr. Julio Cesar Nievola PUCPR	Julio Cesar Nievola	APROVADO
Profa. Dra. Lúcia H. M. Rino UFSCar		Aprovado

Conforme as normas regimentais do PPGIA e da PUCPR, o trabalho apresentado foi considerado APROVADO SEM RESTRIÇÕES (aprovado sem restrições, aprovado com exigências ou reprovado), segundo a avaliação da maioria dos membros da Banca Examinadora acima indicada.

Observações, exigências e/ou restrições da Banca Examinadora, quando houver:

_____

_____

continuar no verso, se necessário

Carlos Maziero

Prof. Dr. Carlos Alberto Maziero, Diretor
Programa de Pós-Graduação em Informática Aplicada
Pontifícia Universidade Católica do Paraná

PPGIA PUCPR

A meus pais, pelos inigualáveis esforços na formação profissional, acadêmica e moral de seus filhos. Vocês são os melhores pais do mundo!

AGRADECIMENTOS

Ao meu orientador e co-orientador, pelo conhecimento, auxílio e dedicação na realização deste trabalho.

Aos meus grandes amigos de Curitiba: André, Denes e Mauro, pelo incrível incentivo, animação e cervejadas necessárias na confecção deste trabalho.

Aos meus colegas de trabalho na Vesta Technologies, pelo incentivo e por realizarem meu trabalho durante o tempo que dediquei a confecção da presente dissertação, ajuda sem a qual este trabalho não seria concluído.

Sumário

Lista de Tabelas

Lista de Figuras

Lista de Abreviações e Siglas

Lista de Símbolos

Resumo

Abstract

Lista de Publicações

Sumário das Publicações e Contribuição do Autor

1 Introdução/Motivação 1

2 Revisão da Literatura 4

2.1 Introdução a sumarização 4

2.2 Revisão de Recuperação de Informações 7

2.3 Avaliação dos Resultados 11

2.4 Técnicas Clássicas de Sumarização 14

2.5 Técnicas Baseadas em Treinamento 18

2.6 Técnicas Baseadas na estrutura do discurso (Coesão e Coerência) 22

2.7 Resultados importantes na sumarização de textos 44

3 Desenvolvendo Novas Técnicas para Sumarização Automática de Textos 52

3.1 Análise das Heurísticas 52

3.2 Os 4 sistemas de Sumarização Propostos 60

4 Experimentos e Análise de Resultados 73

4.1 Formato e métricas de Avaliação 73

4.2 Sistemas de Sumarização Utilizados 77

4.3 Descrição das bases de documentos utilizadas 81

4.4 Resultados dos Experimentos 83

5 Conclusões e direções futuras 98

6 Referências Bibliográficas 100

Apêndice: Exemplo completo de sumarização 103

Lista de Tabelas

Tabela 1 - Resultados do sistema de Cadeias Léxicas 28
Tabela 2 - Resultados do sistema de Árvores de Estruturas Retóricas 37
Tabela 3 - Taxa de acerto dos 11 sistemas contra sumários ideais automáticos com 10% das sentenças do documento 84
Tabela 4 -Resultados do sistema 5 contra sumários ideais automáticos com 10% das sentenças do documento 85
Tabela 5 - Taxa de acerto dos 12 sistemas contra sumários ideais automáticos com 20% das sentenças do documento 86
Tabela 6 - Taxa de acerto dos 12 sistema treinados com sumários automáticos contra sumários ideais manuais com 10% das sentenças do documento 88
Tabela 7 - Taxa de acerto dos 12 sistemas treinados com sumários automáticos contra sumários ideais manuais com 20% das sentenças do documento 89
Tabela 8 - Comparação entre sumários automáticos e manuais 90
Tabela 9 - Taxa de acerto dos 12 sistemas treinados com sumários manuais contra sumários ideais manuais com 10% das sentenças do documento 95
Tabela 10 - Taxa de acerto dos 12 sistemas treinados com sumários manuais contra sumários ideais manuais com 20% das sentenças do documento 96

Lista de Figuras

Figura 1 - Cadeia Léxica/Passo 1 25

Figura 2 - Cadeia Léxica/Passo 2 - Interpretação 1 25

Figura 3 - Cadeia Léxica/Passo 2 - Interpretação 2 26

Figura 4 - Mapa de relacionamentos do texto 29

Figura 5 - Segmentação de texto 30

Figura 6 - Árvore de Estrutura Retórica 35

Figura 7 - Divisões tópicas encontradas pelo Algoritmo *TextTiling* 55

Figura 8 - Resultados do Clustering Aglomerativo em um Documento 56

Figura 9 - Árvore Binária gerada a partir do *Clustering* Aglomerativo 57

Figura 10 - Ramo de árvore com nós salientes 57

Figura 11 - Tela do sistema de sumarização de textos *TF-ISF* 62

Figura 12 - Aplicação do algoritmo TextTiling em um documento 64

Figura 13 - Arquitetura do sistema de sumarização baseado na importância relativa dos tópicos 66

Figura 14 - Sistema Extrator de Características 69

Figura 15 - Sistema Gerador de Arquivos para Classificação 69

Figura 16 – Softwares utilizados nos experimentos 75

Figura 17 - Saída do *Clustering* Aglomerativo entre as técnicas para sumários de 10% 92

Figura 18 - Saída do *Clustering* Aglomerativo entre as técnicas para sumários de 20% 93

Lista de Abreviações e Siglas

ARA	- Answer Recall Average
CMP-LG	- Computation and Language E-Print Archive
DF	- Document Frequency
IDF	- Inverse Document Frequency
MMR	- Maximal Marginal Relevance
RST	- Rhetorical Structure Theory
SUMMAC	- TIPSTER Text Summarization Evaluation
TF	- Term Frequency
TF-IDF	- Term Frequency - Inverse Document Frequency
TF-ISF	- Term Frequency - Inverse Sentence Frequency
TF-TL	- Term Frequency - Term Length

Lista de Símbolos

X .Y	- produto interno dos vetores X e Y
|X|	- tamanho do vetor X
idf(i)	- valor da freqüência inversa do documento para o termo i.
df(i)	- frequência do documento para o termo i.
w(i,d)	- peso da palavra i no documento d
tf(i,d)	- frequência do termo i no documento d
$p(s \in S) \mid F_1, F_2, ..., F_k)$	- probabilidade de uma sentença s percencer ao sumário S, sendo dadas as características F_1, F_2, até F_k.
$\prod$	- Produtório
G^2	- Indica a variação da qual a freqüência de um termo no documento é maior do que o esperado da sua freqüência em toda a base de documentos.
D_i	- documento i
d_{it}	- peso do termo t no documento i
S, R	- conjuntos de documentos
R\S	- diferença dos conjuntos R e S
R = IR(C,Q,θ)	- lista de documentos recuperados por um sistema de recuperação de informações, dados C,Q e um limiar θ, abaixo do qual os documentos não serão recuperados
λ	- coeficiente que regula o uso do MMR.
TP	- verdadeiros positivos
FP	- falsos positivos
TN	-verdadeiros negativos
FN	- falsos negativos
ISF(w)	- frequência inversa na sentença da palavra w

Avg-TF-ISF(s)	- média aritmética do TF-ISF
Max-Avg-TF-ISF	- valor máximo de TF-ISF médio
Threshold-TF-ISF	- valor limiar do TF-ISF
RelTopImp(t)	- importância relativa do tópico t
$Top\ \mathrm{Im}\ p(t)$	- importância do tópico t
σ	- desvio padrão da amostra
P_i	- valor na amostra da medida analisada
$\bar{x}$	- média na amostra da medida analisada
d	- distância enclidiana
s_i	- valor da dimensão i no vetor s

Resumo

O rápido crescimento da quantidade de informação disponível *online* em forma textual não-estruturada necessita de dispositivos que auxiliem o acesso rápido à informação. Neste contexto, a sumarização automática de textos pode ser considerada uma tarefa crucial. Sumarização é o processo de reduzir o tamanho de um texto, preservando o seu conteúdo informativo. Entretanto, produzir um sumário de um texto qualquer é um desafio que requer completo entendimento do texto, fato que está além do estado da arte atual da computação. Apesar disso, diversos sistemas robustos utilizam técnicas estatísticas e baseadas em análise lingüística superficial, de forma independente de domínio, que são capazes de produzir sumários de razoável qualidade para textos de diversos gêneros e estilos.

Este trabalho apresenta uma contribuição nos estudos já realizados para sumarização automática de textos, propondo e comparando quatro diferentes técnicas de sumarização extrativa com enfoque estatístico e de análise lingüística superficial para textos escritos na língua inglesa. As técnicas incorporam diversas heurísticas, todas objetivando classificar a relevância de cada sentença para inclusão no sumário. Deve-se salientar que todas as heurísticas produzem características que são passíveis de serem obtidas de textos não-estruturados, de qualquer domínio e gênero.

As quatro técnicas de sumarização foram avaliadas de forma objetiva, envolvendo a capacidade de replicarem o sumário extrativo "ideal". Como base para comparação, foram utilizados sumários contendo sentenças aleatórias do texto, as primeiras sentenças do texto e produzidos pelo Microsoft Word Summarizer. Os resultados obtidos foram validados em uma base de 100 documentos, contendo sumários extrativos produzidos automaticamente e uma base de 30 documentos, contendo sumários extrativos produzidos manualmente. Adicionalmente, uma análise comparativa entre sumários automáticos e manuais foi realizada e os sumários produzidos por 11 técnicas diferentes foram agrupados por similaridade através do uso de clustering aglomerativo, visando descobrir padrões na geração de sumários automáticos.

Resumo

O rápido crescimento da quantidade de informação disponível *online* em forma textual não-estruturada necessita de dispositivos que auxiliem o acesso rápido à informação. Neste contexto, a sumarização automática de textos pode ser considerada uma tarefa crucial. Sumarização é o processo de reduzir o tamanho de um texto, preservando o seu conteúdo informativo. Entretanto, produzir um sumário de um texto qualquer é um desafio que requer completo entendimento do texto, fato que está além do estado da arte atual da computação. Apesar disso, diversos sistemas robustos utilizam técnicas estatísticas e baseadas em análise lingüística superficial, de forma independente de domínio, que são capazes de produzir sumários de razoável qualidade para textos de diversos gêneros e estilos.

Este trabalho apresenta uma contribuição nos estudos já realizados para sumarização automática de textos, propondo e comparando quatro diferentes técnicas de sumarização extrativa com enfoque estatístico e de análise lingüística superficial para textos escritos na língua inglesa. As técnicas incorporam diversas heurísticas, todas objetivando classificar a relevância de cada sentença para inclusão no sumário. Deve-se salientar que todas as heurísticas produzem características que são passíveis de serem obtidas de textos não-estruturados, de qualquer domínio e gênero.

As quatro técnicas de sumarização foram avaliadas de forma objetiva, envolvendo a capacidade de replicarem o sumário extrativo "ideal". Como base para comparação, foram utilizados sumários contendo sentenças aleatórias do texto, as primeiras sentenças do texto e produzidos pelo Microsoft Word Summarizer. Os resultados obtidos foram validados em uma base de 100 documentos, contendo sumários extrativos produzidos automaticamente e uma base de 30 documentos, contendo sumários extrativos produzidos manualmente. Adicionalmente, uma análise comparativa entre sumários automáticos e manuais foi realizada e os sumários produzidos por 11 técnicas diferentes foram agrupados por similaridade através do uso de clustering aglomerativo, visando descobrir padrões na geração de sumários automáticos.

Abstract

The fast growth of available on-line information in textual form requires systems that provide faster access to information. In this context, Automatic Text Summarization can be considered a crucial task. Summarization is the process of reducing the length of a text, while preserving its informative content. However, producing a summary of any text is a challenge that ideally requires full understanding of the text, which is beyond the current state of the art. Despite this, many robust systems use statistical and shallow linguistic processing, in a domain-independent form, being capable of producing reasonable summaries for texts of different styles and genres.

This work presents a contribution to the research already carried out for Automatic Text Summarization, proposing and comparing four different techniques of extractive summarization for English texts using statistical approaches and shallow linguistic analysis. The techniques use many heuristics, all trying to classify the relevance of each sentence for inclusion in the summary. It must be pointed out that all heuristics produce features that can be extracted from non-structured texts, of any domain and genre.

The four summarization techniques have been evaluated in an objective way, involving the capacity of the system to reproduce the "ideal" extractive summary. As a baseline for comparation, we use summaries containing randomly-selected sentences of text, containing the first sentences of the text, and produced by the Microsoft Word Summarizer. The results have been validated in a corpus of 100 documents, using automatically-produced extractive summaries, and a corpus of 30 documents, using manually-produced extractive summaries. In addition, a comparative analysis between automatic and manual summaries was carried out and the summaries produced by 11 different techniques have been grouped by similarity through the use of agglomerative clustering, trying to discover patterns in the generation of automatic summaries.

Abstract

The fast growth of available on-line information in textual form requires systems that provide faster access to information. In this context, Automatic Text Summarization can be considered a crucial task. Summarization is the process of reducing the length of a text, while preserving its informative content. However, producing a summary of any text is a challenge that ideally requires full understanding of the text, which is beyond the current state of the art. Despite this, many robust systems use statistical and shallow linguistic processing, in a domain-independent form, being capable of producing reasonable summaries for texts of different styles and genres.

This work presents a contribution to the research already carried out for Automatic Text Summarization, proposing and comparing four different techniques of extractive summarization for English texts using statistical approaches and shallow linguistic analysis. The techniques use many heuristics, all trying to classify the relevance of each sentence for inclusion in the summary. It must be pointed out that all heuristics produce features that can be extracted from non-structured texts, of any domain and genre.

The four summarization techniques have been evaluated in an objective way, involving the capacity of the system to reproduce the "ideal" extractive summary. As a baseline for comparation, we use summaries containing randomly-selected sentences of text, containing the first sentences of the text, and produced by the Microsoft Word Summarizer. The results have been validated in a corpus of 100 documents, using automatically-produced extractive summaries, and a corpus of 30 documents, using manually-produced extractive summaries. In addition, a comparative analysis between automatic and manual summaries was carried out and the summaries produced by 11 different techniques have been grouped by similarity through the use of agglomerative clustering, trying to discover patterns in the generation of automatic summaries.

Lista de Publicações

Esta dissertação consiste de uma introdução, apresentação e comparação dos sistemas propostos nas seguintes publicações:

1. Larocca Neto. Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. . Document clustering and text summarization. *Proceedings of 4th Int. Conf. Practical Applications of Knowledge Discovery and Data Mining (PADD-2000)*, 41-55. London: The Practical Application Company. 2000.

2. Larocca Neto, Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. . Generating Text Summaries through the Relative Importance of Topics. *Proceedings of IBERAMIA-2000 (7th Ibero-American Conf. on Artif. Intel.) & SBIA-2000 (15th Brazilian Symp. on Artif. Intel.)*. Sao Paulo, SP, Brazil. Nov. 2000.

3. Larocca Neto, Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. ; Nievola, Júlio C. . Sumarização como um problema de classificação. *Technical Report PPGIA*. Pontifícia Universidade Católica. Jul. 2000.

4. Larocca Neto, Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. ; Nievola, Júlio C. . A trainable algorithm for summarizing news stories. *Proceedings of PKDD-2000 Workshop on Machine Learning and Textual Information Access*. Lyon, France. Sep. 2000.

Sumário das Publicações e Contribuição do Autor

Na publicação 1, é proposto um método simples para realização de sumários automáticos, baseado no uso da representação vetorial de documentos e utilizando a métrica *TF-ISF* (*Term-Frequency – Inverse Sentence Frequency*), uma adaptação do convencional *TF-IDF*, comumente utilizado na Recuperação de Informações [Salton 88]. Sentenças com altos valores de *TF-ISF* são selecionadas para fazerem parte do sumário, a partir de um limiar especificado pelo usuário.

Na publicação 2, é proposto um método mais elaborado que particiona o documento em tópicos (utilizando o conhecido algoritmo Text Tiling [Hearst 93]), que identifica segmentos coerentes do texto. Para cada tópico gerado, o método calcula a relevância relativa no documento, utilizando a noção de TF-ISF e o centróide (vetor médio de todas as sentenças presentes no documento). Finalmente, sentenças são extraídas dos tópicos de acordo com a sua semelhança com o centróide, e em número proporcional à relevância do mesmo. Os resultados foram analisados por uma especialista em lingüística, utilizando 7 documentos da tarefa *Adhoc* do projeto SUMMAC [Mani 98c]. Os sumários produzidos pelo sistema foram comparados com os produzidos pelos sistemas CGI/CMU e Cornell, que foram os sistemas que apresentaram os melhores resultados durante a SUMMAC.

Nas publicações 3 e 4 são propostos dois sistemas de sumarização treináveis, no contexto de uma tarefa de classificação, ao contrário dos sistemas anteriores que utilizavam heurísticas fixas para sumarização. Estas propostas foram elaboradas considerando as evidências de [Kupiec 95], de que um sistema de sumarização extrativo incorpora diversas heurísticas para classificar as sentenças e que as importâncias das heurísticas individuais devem ser aprendidas automaticamente pelo sistema.

Na publicação 3 é proposto um sistema de sumarização que utiliza diversas características baseadas em posição, tamanho e presença de palavras temáticas/relevantes nas sentenças e na identificação de segmentos coesos do texto. O enfoque é utilizar atributos que possam ser obtidos de textos não-estruturados. Os resultados foram analisados em cerca de 183 documentos técnicos da base CMP-LG (utilizada em diversos trabalhos da

área [Teufel 99] [Mani 98a]) e comparados com resultados obtidos por outros pesquisadores. O algoritmo de aprendizagem de máquina utilizado foi o classificador Naive-Bayes, tradicionalmente utilizado em experimentos de sumarização treinável [Teufel 99] [Kupiec 95].

Na publicação 4, o sistema treinável proposto é direcionado para sumarização automática de notícias e faz uso da estrutura argumentativa do texto, através de uma representação em árvore binária obtida do uso de *clustering* aglomerativo, em contraposição à publicação 2, que utilizava uma representação linear gerada pelo algoritmo Text Tiling. As características utilizadas envolvem a detecção dos conceitos principais do texto, grau de conectividade entre as sentenças do documento e ocorrência de nomes próprios, anáforas e marcadores de discurso comuns. Os algoritmos de aprendizagem de máquina utilizados foram o C4.5 [Quinlan 92] e Naive-Bayes. O sistema foi avaliado em uma base com cerca de 800 documentos, obtido dos textos da publicadora *Ziff-Davis* da base TIPSTER [Harman 94].

Em resumo, a contribuição principal do autor é a proposta e comparação de diferentes técnicas de sumarização extrativa com enfoque estatístico e de análise lingüística superficial. As técnicas utilizadas baseiam-se tanto em heurísticas fixas como em treinamento, e foram baseadas nas informações temáticas do texto, métricas conhecidas de Recuperação de Informações, posição e tamanho das sentenças, coesão e estrutura do texto.

1 Introdução/Motivação

O rápido crescimento da quantidade de informação disponível *online* em forma textual não estruturada necessita de dispositivos que auxiliem o acesso rápido a informação. Particularmente, são necessários métodos que auxiliem o julgamento da relevância dos textos em questão. Uma das técnicas que podem ser utilizadas na realização desta tarefa é a sumarização.

Sumarização é o processo de reduzir o tamanho de um texto, preservando o seu conteúdo informativo.

Entretanto, produzir um sumário de um texto qualquer é um desafio que requer idealmente completo entendimento do texto, fato que está além do estado da arte atual da computação. [Brandow 94] [Mitra 97].

Apesar disso, diversos sistemas robustos utilizam técnicas estatísticas e baseadas em análise lingüística superficial, de forma independente de domínio, que são capazes de produzir sumários de razoável qualidade para textos de diversos gêneros e estilos.

Os sumários automáticos também desempenham um papel importante no auxílio à recuperação de informações: em posse do sumário, o usuário é capaz de julgar se o documento completo deve ou não ser analisado.

Embora muitos textos já contenham sumários escritos pelos próprios autores, a geração automática de sumários apresenta algumas características únicas:

- Geração de um sumário com o tamanho especificado pelo usuário. Tal característica permite que o usuário selecione o nível desejado de granularidade, diferente dos sumários fornecidos pelos autores, que são estáticos.
- Criação de ligações entre uma frase do sumário com bloco de texto correspondente.
- Criação de sumários com o foco do usuário, selecionando informações relativas a um interesse particular.

Sumários gerados automaticamente também podem diminuir o grau de interferência exercida pelo sumarizador humano, que é influenciado pela disposição, atitude e opiniões particulares. Assim sendo, a sumarização automática pode resultar em maior consistência e objetividade nos sumários.

Cabe-se também destacar o uso conjunto com a Recuperação de Informações, onde sumários automáticos podem ser utilizados para identificar documentos relevantes em uma coleção, ou sumarizando similaridades e diferenças entre documentos (sumarização multi-documentos).

A área de sumarização está atualmente sendo foco de grande interesse no setor comercial, incluindo a indústria de telecomunicações, mineração em bancos de dados textuais e filtros para sistemas de recuperação de informações na *Web*. O governo americano, através da DARPA, identificou a sumarização de textos como uma área de pesquisa crítica e começou a realizar investimentos e estímulos ao desenvolvimento da área, através de iniciativas com a TIPSTER SUMMAC [Mani 98c], que visa estimular o desenvolvimento do estado da arte na área.

Em síntese, pode-se argumentar que sumários automáticos dificilmente apresentam a mesma qualidade de um sumário produzido por um humano em relação a fluência, coerência e síntese, mas por outro lado podem ser mais úteis para a realização de diversas tarefas devido a sua características dinâmicas.

Esta dissertação tem dois objetivos principais: (a) propor 4 diferentes técnicas de sumarização extrativa com enfoque estatístico e de análise lingüística superficial para textos escritos na língua inglesa; (b) realizar um estudo comparativo entre as 4 técnicas propostas. As técnicas incorporam diversas heurísticas, todas objetivando estimar a relevância de cada sentença para decidir se aquela sentença deve ou não ser incluída no sumário. A limitação da língua é imposta pela maioria das técnicas, que envolvem pré-processamentos disponíveis somente para a língua inglesa.

A dissertação está dividida da seguinte maneira:

No capítulo "Revisão de literatura", a tarefa de sumarização automática de textos é introduzida e diversas técnicas tradicionalmente utilizadas serão explanadas e comparadas.

No capítulo "Desenvolvendo Novas Técnicas para Sumarização Automática de Textos" são explorados os potenciais de utilização e extensão de alguns grupos conhecidos de heurísticas aplicadas a sumarização de textos. No mesmo capítulo, serão explicadas as 4 técnicas de sumarização propostas neste trabalho.

No capítulo "Experimentos e Análise de resultados", as 4 técnicas de sumarização serão avaliadas de forma objetiva, envolvendo a capacidade de replicarem o sumário extrativo "ideal". Para fins destes experimentos, utilizou-se dois tipos de procedimentos para produzir um sumário "ideal": um sumário gerado da forma automática (utilizando técnica explicada mais adiante) e um sumário manual (produzido por um juiz humano).

Adicionalmente, os resultados serão comparados com sumários obtidos por 3 outros métodos, a saber: seleção de sentenças aleatórias do texto, seleção das primeiras sentenças do texto e gerados pelo sistema comercial de sumarização Microsoft Word Summarizer, visando situar os sistemas em relação a ferramentas disponíveis no mercado. As métricas de avaliação utilizadas serão os valores de *precision* e *recall* (definidos na seção 2.1) em sumários contendo 10% do número de sentenças do documento. No caso de sumários gerados de forma automática, uma base de 100 documentos será utilizada na avaliação. Para sumários manuais, a base de documentos utilizada contém 30 documentos. Os sumários automáticos serão comparados com os manuais visando verificar o seu grau de concordância, fator determinante para experimentos futuros utilizando sumários automáticos. Os sumários produzidos pelas 4 técnicas também serão avaliados com relação a similaridades entre si, através do uso de clustering aglomerativo visando descobrir padrões na geração de sumários automáticos.

Finalmente, no capítulo "Conclusões e Direções Futuras", são apresentados os resultados finais e perspectivas do trabalho realizado.

2 Revisão da Literatura

Este capítulo está organizado da seguinte maneira: em "Introdução a sumarização" a tarefa de sumarização automática de textos é introduzida, discriminando-se os principais tipos de sumários existentes e o contexto histórico da área. Na seção "Revisão de Recuperação de Informações" são vistos alguns conceitos básicos da área que apresenta grande interseção com a sumarização de textos, tanto como tarefa combinada quanto na utilização de técnicas e terminologias comuns. Na seção "Avaliação dos Resultados" são abordados os principais tipos de avaliações possíveis e as métricas comumente utilizadas na avaliação. As seções seguintes apresentam diversas técnicas utilizadas em trabalhos realizados na área, agrupadas nas seguintes categorias: "Técnicas clássicas de Sumarização", "Técnicas baseadas em Treinamento" e "Técnicas baseadas na estrutura do discurso (coesão e coerência)". Finalmente na seção "Resultados Importantes na Sumarização de Textos" são abordados alguns experimentos importantes relatados na área e a conferência TIPSTER SUMMAC, o maior evento já realizado para avaliação de sistemas de sumarização automática.

2.1 Introdução a sumarização

Segundo [Spark Jones 99], a tarefa de sumarização pode ser dividida em 3 etapas:

- Criar uma representação do texto origem;
- Transformar a representação do texto origem para uma representação de sumário;
- Gerar o sumário a partir da representação;

Sumários podem ser construídos através de uma análise semântica profunda do texto ou através de análise lingüística superficial do texto.

Existem basicamente 3 tipos de sumários [Morris 92]:

- Indicativo: também chamado de descritivo, apresenta como função descrever sobre o que o texto trata e auxiliar o leitor a decidir se o texto original deve ou não ser lido.

- Informativo: tenta sumarizar a informação do texto de forma que o leitor não necessite consultar o texto original.
- Crítico: Analisa o texto, expressando a opinião do revisor em relação ao texto original.

Devido ao fato de poucas aplicações necessitarem de sumários críticos, e porque a tecnologia necessária para produzir os outros dois tipos está além da atual, a pesquisa em sumarização restringiu-se ao desenvolvimento de sumários informativos.

A área surgiu nos anos 60, através do trabalho de Luhn [Luhn 58]. Nos anos 70 e 80, pouca pesquisa foi desenvolvida e poucos avanços foram obtidos. Nos anos 90, a área renasceu devido a interesses comerciais e governamentais. Porém, quase todos os trabalhos desenvolvidos podem ser classificados como Extração de Texto ou Extração de Fatos.

A extração do texto funciona do modo "O que você vê é o que você tem". Na extração de textos, as fases 1 e 2 mencionadas no início desta seção são uma única fase; segmentos de textos (sentenças ou parágrafos) são tipicamente selecionados através de uma mistura de critérios estatísticos, posicionais ou baseados em palavras chaves.

Na extração de fatos, as fases 1 e 2 também são uma única fase. No processamento inicial, expressões no texto que capturam conceitos são localizadas e uma representação baseada em *frames* ou *templates* é gerada. A geração do sumário envolve a produção de textos em linguagem natural a partir da representação obtida.

Técnicas de extração de textos têm a vantagem da generalidade, pois podem ser aplicadas em quaisquer tipos de texto, mas geralmente produzem uma saída de baixa qualidade. A extração de fatos geralmente apresenta uma saída de melhor qualidade, mas tem a desvantagem de requerer que o tipo de informação sendo utilizada seja explicitamente especificada, sendo utilizada somente em domínios cujas características são previsíveis e bem compreendidas. Não existem evidências fortes que tais sistemas estarão aptos a manipularem textos de forma independente de domínio em um futuro próximo [Brandow 94] [Mitra 97].

Devido à dificuldade constatada, a maioria dos sistemas focalizaram a tarefa de extração de texto, conhecida como sumarização extrativa.

A sumarização por extração pode ser definida como encontrar um subconjunto do documento que é representativo do seu conteúdo, tipicamente atribuindo pontos a sentenças de acordo com a sua relevância e preservando as sentenças com maior pontuação. Estes tipos de sumários não são garantidos de terem coerência narrativa, mas podem ser úteis para um julgamento de relevância e são tratáveis pela tecnologia atual.

Diversas heurísticas foram propostas para sumarização por extração, mas não há nenhum critério muito claro para escolher uma delas. Existem evidências que a combinação de diversas heurísticas deve apresentar melhores resultados.

Algumas destas heurísticas são:

- Presença de palavras indicativas (*cue words*) ou palavras do título do documento na sentença sendo considerada para extração [Edmundson 69].
- Posição da sentença no documento [Edmundson 69].
- Técnicas de Recuperação de Informação: uso de estatísticas baseadas na freqüência de palavras, utilizadas para calcular o grau de importância de uma palavra, ou técnicas de recuperação de passagens no texto [Luhn 58].
- Sentenças importantes existem em posições mais coesas do texto (sentenças mais conectadas do texto). [Barzilay 97][Carbonell 98] [Mitra 97]
- Incorporação da estrutura de discurso (retórica) do texto na criação de sumários [Marcu 99] [Teufel 99].

Um problema inerente da sumarização por extração é que ela não funciona bem em situações aonde se deseja altas taxas de compressão e em documentos muito longos [Teufel 99].

Algumas destas heurísticas serão discutidas em mais detalhes na seção 2.3.

Apesar da maioria dos trabalhos atuais focalizar em extrações ao invés de em resumos propriamente ditos, este enfoque deve se modificar em alguns anos, com o avanço do Processamento em Linguagem Natural. Além disso, novas sub-áreas estão surgindo,

como sumarização de múltiplos documentos, sumarização multimídia e sumarização multi-língua.

2.2 Revisão de Recuperação de Informações

Sistemas de recuperação de informações são utilizados para recuperar documentos ou textos com conteúdo informativo relevante para uma necessidade do usuário. Inicialmente, sistemas de recuperação de informações foram utilizados principalmente por bibliotecas na busca de material bibliográfico. O posterior advento de processadores de texto e CD-ROMs gerou mecanismos efetivos para criação e disseminação de textos em forma eletrônica, motivando o surgimento de técnicas de busca de informações sobre textos completos. A recuperação de informações consiste principalmente de duas diferentes atividades: indexação e busca. A indexação refere-se ao modo como os documentos são representados para posterior recuperação. A busca refere-se ao modo como a representação dos documentos é examinada e o modo como estes são considerados relacionados a uma consulta provida pelo usuário [Witten 94].

A seguir, serão apresentadas algumas técnicas tradicionais da área da recuperação de informações que são tipicamente utilizadas por sistemas de sumarização: pré-processamento, representação vetorial dos documentos e sistema de pesos aplicados a representação vetorial.

Pré-processamento e representação vetorial dos documentos

A representação dos documentos apresenta grande impacto na capacidade do sistema. Devido a severas limitações da tecnologia corrente em processamento robusto de texto, necessita-se definir estruturas que possam ser extraídas do texto automaticamente e a um custo computacional razoável. Por outro lado, a estrutura deve ser rica o suficiente para permitir encontrar informações interessantes.

A forma mais tradicional de representar documentos na área de recuperação de informações é o paradigma *bag-of-words* [Joachims 96], que representa palavras na forma atributo-valor.

Desta forma, cada palavra do documento é tratada como um atributo, e os valores representam o número de ocorrências de cada palavra no documento. Deve-se notar que esta representação assume que a ordem das palavras no documento não é relevante.

A representação vetorial de documentos foi introduzida por Salton [Salton 88] e representa documentos como vetores multi-dimensionais. Cada dimensão do vetor representa a freqüência de ocorrência de um termo no documento.

$$X = (t_1, t_2, ..., t_n),$$

sendo n o total de termos da coleção de documentos

Considerando-se todos os termos possíveis em uma coleção de documentos, um vetor pode ter centenas de dimensões, a maioria delas com o valor zero para cada documento específico. A representação de documentos como vetores sugere uma medida natural de similaridade: o ângulo entre os dois vetores. Quanto menor o ângulo entre os dois vetores, mais similares os documentos são. Se X e Y são 2 vetores n-dimensionais, o angulo θ entre os dois satisfaz:

$$X \cdot Y = |X||Y| \cos\theta$$

aonde X·Y é produto interno, e $|X| = \left(\sum_{i=1}^{n} x_i^2 \right)^{1/2}$ é o comprimento euclidiano do vetor X. O ângulo θ pode ser calculado por:

$$\cos\theta = \frac{X \cdot Y}{|X||Y|} = \frac{\sum_{i=1}^{n} x_i y_i}{\left(\sum_{i=1}^{n} x_i^2 \right)^{1/2} \left(\sum_{i=1}^{n} y_i^2 \right)^{1/2}}$$

Os valores de cosθ variam de 1 para documentos com máxima similaridade e 0 para documentos sem nenhuma similaridade.

Esta métrica é conhecida como similaridade dos cosenos, e é amplamente utilizada na pesquisa de Recuperação de Informações e Sumarização de Textos, aonde tipicamente a

consulta (também considerada um documento) é comparada com cada um dos documentos pertencentes a uma base, e os resultados são ordenados de acordo com a similaridade.

Como um documento apresenta um número alto de palavras únicas, existem 3 processos tradicionalmente aplicados visando reduzir a dimensionalidade excessiva [Witten 94]:

- Stop words

Stop words são termos que ocorrem muito freqüentemente e não devem ser utilizados. Um conjunto de *stop words* é chamado de *stop list*. Os termos *can*, *will*, *do* e *does* são exemplos de *stop words*. Geralmente, sistemas de recuperação de informação utilizam uma *stop list* contendo palavras que não devem ser consideradas na representação vetorial dos documentos.

- Case Folding

Case Folding é o processo de substituição de todos os caracteres para o mesmo formato. Por exemplo, *the*, *The*, *THE* e outras cinco combinações possíveis são convertidas para o termo *the*.

- Stemming

Stemming é a conversão de palavras para a forma de radical, eliminando flexões verbais e plural. O processo de stemming envolve a remoção de um ou mais sufixos, convertendo o termo para uma forma neutra. Por exemplo, os termos *compression* e *compressed* seriam aceitos como equivalentes ao termo *compress*.

Implementações de algoritmos de *stemming* requerem conhecimentos detalhados de linguística, sendo mais utilizado o algoritmo de Porter [Porter 80], originalmente feito para a língua inglesa.

A representação de N-Grams é uma alternativa ao processo de *stemming* e remoção de *stop words*. Um n-gram [Cavnar 94] é um segmento com o comprimento de n caracteres. Por exemplo, a palavra DATA pode ser representada pelos tri-grams (n=3): *_DA, DAT, ATA, TA_* ou pelos quadri-grams(n=4): *DAT, DATA, ATA_*, onde o símbolo "_" representa o espaço existente antes ou depois da palavra. Em comparação ao *stemming* e

remoção de *stop words*, a representação de n-grams tem a vantagem de ser mais robusta – menos sensitiva a erros tipográficos e gramaticais e torna o sistema mais independente de linguagem. Entretanto, a representação de n-grams não é tão efetiva em reduzir o número de dimensões (palavras) de um texto.

Sistema de pesos aplicados a representação vetorial

Ao invés de simplesmente armazenar o número de ocorrências de cada palavra na representação vetorial, algumas heurísticas foram propostas visando diferenciar a importância das palavras, fornecendo valores mais altos para palavras extremamente significantes. Neste contexto, serão abordadas 2 técnicas que podem ser utilizadas para atribuir pesos diferenciados as palavras: o TF-IDF e o TF-TL.

TF-IDF (Term Frequency – Inverse Document Frequency)

Salton explorou diversos métodos de calcular pesos para as palavras. Os pesos de palavras TF-IDF representam o valor de cada palavra na representação vetorial combinando as estatísticas TF(i, d) e DF(i).

O TF(i, d) (*term frequency*) é o número de vezes em que a palavra i aparece no documento d.

O DF(i) (*document frequency*) é o número de documentos no qual a palavra i aparece ao menos uma vez.

O IDF(i) (*inverse document frequency*) pode ser calculado através de DF(i), utilizando-se a seguinte fórmula:

$$IDF(i) = \log\left(\frac{|D|}{DF(i)}\right)$$, onde |D| é a cardinalidade do conjunto de documentos.

O IDF(i) de uma palavra é baixo se esta ocorre em muitos documentos e alto se a palavra ocorre somente em um documento.

O valor de cada dimensão do vetor é calculado através da seguinte fórmula:

$$w(i,d) = TF(i,d) * IDF(i)$$

Esta heurística de pesos para palavras diz que uma palavra é um termo importante de um documento se esta ocorre freqüentemente neste (tf alto). Por outro lado, palavras que ocorrem em muitos documentos são termos pouco importantes devido ao baixo idf.

TF-TL (Term Frequency – Term Length)

Em [Banko 99] é proposta outra técnica para obtenção de pesos para as palavras. O TF-TL multiplica a função monotônica do tamanho do termo pelo função monotônica da freqüência do termo. O TF-TL é um algoritmo eficiente, escalável e apropriado para implementação distribuída porque não requer nenhuma informação externa ao documento. Ao contrário do TF-IDF, ele não apresenta problemas de atualização quando alguns documentos são inseridos ou removidos da base de documentos, e não consome mais recursos quando o número de documentos aumenta. O TF-TL é razoavelmente efetivo, superando alguns métodos de obtenção de pesos para palavras, apesar de não ser tão bom quanto os melhores métodos. Um benefício adicional é o fato dele ser muito simples de implementar e generalizável para n-grams. O TF-TL é baseado em uma variação do pressuposto da linguagem natural que sugere que linguagens influenciam ou restringem o modo como as pessoas pensam. Em contraste, o algoritmo TF-TL assume que o uso da linguagem e considerações de eficiência exercem pressões evolucionárias na linguagem, com o resultado que palavras usadas mais freqüentemente são mais curtas. Analogamente, palavras que são usadas menos freqüentemente (e que são melhores para distinguir documentos ou passagens) tendem a ser maiores, e portanto tem maior valor de comprimento do termo (TL – *Term Length*).

2.3 Avaliação dos Resultados

Sistemas de sumarização podem ser avaliados de diferentes maneiras: sumário versus texto origem, sumarização automática versus sumarização manual ou sumarização automática versus sumarização automática.

Os métodos de avaliação podem ser classificados em 2 categorias básicas [Mani 98c]:

- extrínseca, onde a qualidade do sumário é julgada através do quanto este afeta a realização de outra tarefa, tal como determinar a relevância do documento em um tópico específico, e a resposta de questões baseadas na leitura de um sumário;
- intrínseca, onde a qualidade do sumário é diretamente baseada na análise do mesmo. Isto envolve julgamentos de fluência, abrangência de idéias chaves ou centrais de um texto ou similaridade com um "sumário ideal". Este tipo de análise é extremamente complexa, devido aos seguintes fatores:
  - Um "sumário ideal" é difícil de ser estabelecido, pois pode existir um grande número de sumários que podem sumarizar um documento. Adicionalmente, isto depende das instruções dadas aos juizes humanos: pode-se extrair todas as sentenças que aparentam pertencer ao sumário, ou extrair os melhores 10 ou 20% das sentenças, ou classificar todas as frases como "muito importante", "moderadamente importante" ou "não importante", ou extrair parágrafos ao invés de sentenças. Outra indicação da inexistência de um sumário ideal é a baixa concordância entre humanos com relação as sentenças boas para um sumário. Em [Mitra 97], foi realizado um experimento que demonstra um co-seleção de somente 46% entre sentenças selecionadas por 2 juizes, para 50 artigos de uma enciclopédia. Entretanto, existem diversos estudos que afirmam que embora os juizes não concordem em todas as sentenças a serem incluídas no sumário, eles concordam nas sentenças mais importantes a serem incluídas. Outra possibilidade é combinar a informação de diversos julgamentos (através da unanimidade, votação da maioria ou união).
  - O julgamento de idéias centrais de um texto é algo muito subjetivo, mesmo quando humanos concordam antecipadamente com o conjunto das idéias principais.

A grande dificuldade na análise de resultados em sistemas de sumarização é refletida nas análises efetuadas pelos pesquisadores da área, visto que a maioria dos autores realizam trabalhos em bases não padrões, providas pelo próprio autor, variando em tamanho (8 documentos em [Marcu 99] a 250 documentos em [Brandow 94]), gênero (notícias em [Brandow 94], textos de enciclopédia em [Mitra 97] e artigos acadêmicos em

[Teufel 99]) e critérios de avaliação (comparação com sumários gerados por humanos em [Mitra 97][Barzilay 97], comparação com sumários extrativos obtidos através de uma consulta utilizando o sumário provido pelo autor em [Mani 98a]).

Tal fato torna difícil a comparação de resultados obtidos, bem como a identificação de técnicas promissoras na área.

Algumas exceções são os sistemas que participaram da TIPSTER SUMMAC Evaluation [Mani 98c], que permitiram uma avaliação detalhada de resultados em diversas tarefas.

Métricas de avaliação

Tipicamente, a avaliação de sumários automáticos é feita através da criação de um sumário ideal, escrito por profissionais ou misturando diversos sumários providos por diversos juizes humanos usando métodos como opinião da maioria, união ou interseção. A saída do sumarizador é então comparada com o sumário ideal, utilizando *precision* e *recall* como unidades de medida da taxa de acerto [Barzilay 97], [Marcu 99].

Os valores de *precision* e *recall* são dados por:

$$\text{Precision}=TP/(TP+FP)$$

$$\text{Recall} = TP/(TP+FN)$$

sendo,

TP = verdadeiros positivos (o número de sentenças incluídas no sumário ideal que foram corretamente selecionadas pelo sistema)

FP = falsos positivos (o número de sentenças não incluídas no sumário ideal que foram incorretamente selecionadas pelo sistema)

TN = verdadeiros negativos (o número de sentenças não incluídas no sumário ideal que não foram selecionadas pelo sistema)

FN = falsos negativos (o número de sentenças incluídas no sumário ideal que não foram selecionadas pelo sistema)

Intuitivamente, o valor de *precision* expressa "o quão correta é a saída do sistema", ou seja, a proporção de sentenças corretas que foram selecionadas pelo sistema. O valor de *recall* expressa "o quanto das respostas corretas o sistema selecionou", ou seja, a proporção das sentenças corretas que o sistema selecionou com relação a todas as sentenças que deveriam ser consideradas corretas.

Deve-se notar que em casos aonde o sistema seleciona o mesmo número de sentenças que o "sumário ideal" (geralmente ocorre em casos aonde o juiz humano e o sistema selecionaram percentuais fixos do número de sentenças, como 10% ou 20%), os valores de *precision* e *recall* são idênticos, e os sistemas utilizam simplesmente o termo taxa de acerto. Este tipo de abordagem foi utilizado em [Kupiec 95] e [Teufel 99].

Na SUMMAC [Mani 98c], foi proposta a utilização do *F-score*, uma métrica que combina valores de *precision* e *recall* da seguinte forma:

$$Fscore = 2 * Precision * Recall / (Precision + Recall)$$

2.4 Técnicas Clássicas de Sumarização

Motivado pela economia de recursos humanos na sumarização, e na obtenção de sumários objetivos, não influenciados pela interpretação pessoal, Luhn [Luhn 58] propôs o primeiro sistema automático de sumarização, utilizando métodos estatísticos para a extração de sentenças relevantes de um documento. A técnica utilizada é baseada na freqüência das palavras significantes contidas no texto e sua posição na sentença. Os textos eram processados da seguinte maneira:

- Efetua-se a remoção de palavras comuns de alta freqüência (pronomes, preposições e artigos) e palavras de baixa freqüência.
- As palavras eram agrupadas através de um cálculo de similaridade entre palavras utilizando o número de letras diferentes entre elas: se este número era menor ou igual a 6, as palavras eram consideradas iguais.
- Em cada sentença, eram procurados agrupamentos contendo palavras significantes separadas por não mais do que 4 a 5 palavras não significantes. De acordo com os

agrupamentos encontrados, as sentenças eram classificadas pelos maiores valores de importância e selecionadas através de um ponto de corte (*Cutoff*) no valor de relevância.

Luhn não apresentou nenhuma análise de resultados detalhada, limitando-se a demonstrar dois exemplos de "*auto-abstracts*" indicativos.

A técnica era relativamente simples, e tinha a desvantagem de não considerar a semântica do texto e muitas das soluções propostas foram substituídas por equivalentes mais poderosos, como *stemming* e uso de freqüências de palavras em vários documentos. Porém, Luhn apresentou diversas possibilidades na área da sumarização automática que tiveram grande impacto em pesquisadores da área, tais como:

- Criação de resumos de tamanho igual a um percentual do tamanho do documento original.
- Criação de sumários de tamanho variável, de acordo com o nível de detalhamento desejado.
- Resumos com ênfase especial em um interesse do usuário, propondo uma solução através de palavras bônus.
- A possibilidade do sistema de sumarização rejeitar um artigo que seja muito genérico para ser sumarizado.
- A aplicação do algoritmo nas subdivisões dos artigos, o que motivou diversas técnicas baseadas na segmentação dos textos.
- Sistemas de sumarização poderiam ser utilizados na geração de termos de indexação na área de Recuperação de Informações.

Edmundson [Edmundson 69] propôs um método de sumarização que utilizava novas características para gerar o sumário, argumentando que embora métodos estatísticos gerassem sumários de qualidade suficiente para motivar trabalhos futuros, o uso de somente métodos estatísticos era inadequado. Ele propôs 4 métodos básicos, todos gerando pesos que indicavam a relevância da sentença e eram posteriormente combinados. Os métodos são:

- **Cue Method (C):** Pressupõe que a relevância da sentença é baseada na presença de palavras como "significante", "impossível" e "difícil". Para realizar a tarefa, utilizou-se um dicionário de palavras que pertenciam a 3 categorias básicas: palavras *bonus*, que adicionavam evidência positiva na inclusão da sentença no sumário, palavras *stigma*, que adicionavam evidência negativa na inclusão da sentença no sumário e palavras nulas, que eram irrelevantes no julgamento da sentença. O dicionário foi construído selecionando-se palavras através de métodos estatísticos e refinamentos em uma base heterogênea de 100 documentos. Edmundson observou que: (a) palavras comparativas, advérbios de conclusão e superlativos tinham grande probabilidade de serem consideradas palavras *bônus*; (b) anáforas e palavras de detalhe insignificante tinham grande probabilidade de serem consideradas palavras *stigma* ; (c) números, pronomes, artigos e adjetivos eram geralmente palavras nulas. O peso final da sentença em relação ao método era dado pela soma dos pesos de cada palavra da sentença.

- **Key Method (K)**: Pressupõe que as palavras relevantes de alta freqüência são úteis para determinar a relevância da sentença no sumário. Edmundson selecionou todas as palavras que não foram escolhidas para participar no método anterior com freqüência maior que um limiar estipulado. As palavras recebiam pesos positivos iguais a sua freqüência. O peso da sentença em relação ao método era dado pela soma dos pesos de todas as palavras da sentença.

- **Title Method (T)**: Pressupõe que a relevância da sentença era baseada na presença de palavras do título ou nome de seções do documento. Para realizar a tarefa, Edmundson construiu um glossário com todas as palavras do título, subtítulo e seções do documento que não eram nulas. O peso da sentença em relação ao método era dado pela soma dos pesos de todas as palavras da sentença presentes no glossário.

- **Location Method (L)**: Pressupõe que sentenças que ocorrem em certas seções do documento ou que ocorrem no começo ou no fim do documento ou do parágrafo são positivamente relevantes ao sumário. Para realizar a tarefa, Edmundson utilizou um dicionário de seções contendo palavras como "Introdução", "Propósito" e "Conclusão". O peso da sentença em relação ao método era dado pela soma dos pesos da posição da sentença no documento e no parágrafo.

Edmundson utilizou pesos reajustáveis para cada um dos métodos e selecionou um conjunto de documentos para treinamento e teste. Deve-se notar que os pesos foram ajustados manualmente, de forma empírica. Para cada documento, um sumário extrativo manual foi realizado por juizes. A base de treinamento era heterogênea, continha 200 documentos. A base de testes continha 200 documentos na área de química. Os sumários produzidos deveriam apresentar um tamanho de 25% das sentenças do texto original e foram comparados com sumários gerados selecionando-se sentenças aleatoriamente.

O peso final da sentença era dado por:

Peso Final = a1.C+a2.K+a3.T+a4.L, onde,

a1,..., a4 são números positivos inteiros

Executando-se todas as combinações entre os diferentes métodos, Edmundson chegou a conclusão que os métodos *Cue*, *Title* e *Location* produziram os melhores resultados na base de treinamento e que o método *Key* produziu o pior resultado comparando-se os métodos isoladamente.

Nos dados de teste, foi conduzido um experimento que julgava a similaridade subjetiva entre os sumários automáticos, sumários realizados por juizes humanos e sumários gerados aleatoriamente. Os valores possíveis de similaridade eram: sem similaridade, similaridade moderada, considerável similaridade, completa similaridade e não aplicável. Para um conjunto de 40 documentos escolhidos aleatoriamente da base de teste foram obtidos os seguintes resultados:

- sumários automáticos versus sumários manuais: similaridade média de 66% (44% de sentenças iguais)
- sumários aleatórios versus sumários manuais: similaridade média de 34% (25% de sentenças iguais)

O trabalho de Edmundson influenciou muito da pesquisa futura na área, especialmente em sistemas baseados em treinamento, como [Kupiec 95], [Teufel 99] e [Mani 98a], e desencorajou o uso da freqüência de palavras como método de sumarização.

2.5 Técnicas Baseadas em Treinamento

Em alguns trabalhos realizados na área, pesquisadores investigaram se uma análise em um conjunto de documentos pode induzir métodos ou conceitos a serem utilizados na sumarização. Tal fato é realçado pela dependência de estratégias de sumarização de acordo com o gênero do texto, visto que sumários de notícias, artigos técnicos e artigos de enciclopédia apresentam características diferentes entre si. Na análise efetuada, os sumários extrativos produzidos por juizes humanos ou obtidos através de outros métodos, como o alinhamento de sumários fornecidos pelo autor com sentenças do texto, podem ser utilizados para aprender regras ou padrões para sumarização automática.

O trabalho que mais motivou a pesquisa na área de sumarizadores treináveis foi [Kupiec 95]. Kupiec propôs o problema de sumarização como um problema estatístico de classificação, cuja necessidade é criar uma função que classifique a probabilidade de uma sentença ser incluída no resumo. Posteriormente, as sentenças são ordenadas de acordo com a probabilidade obtida e o sumário extrativo é gerado. As características utilizadas no sistema para obter a função de classificação, todas de natureza discreta, eram:

- **Sentence Length Cut-off**: Sentenças curtas tendem a não serem incluídas nos sumários. Dado um limiar, por exemplo 5 palavras, a característica é verdadeira para todas as sentenças maiores que o limiar e falsa em caso contrário.
- **Fixed-Phrase**: Sentenças contendo qualquer frase de uma lista, por exemplo "Esta carta...", "Em conclusão..." ou ocorrendo imediatamente depois de um título de seção contendo palavras como "conclusão", "resultados" e "discussão" são mais prováveis de serem incluídas no sumário. A característica é verdadeira para sentenças que contêm qualquer uma das 26 frases indicativas selecionadas, ou para sentenças que seguem títulos de seções que contém palavras específicas.
- **Paragraph**: Verdadeira para sentenças presentes nos 10 primeiros parágrafos e para os últimos 5 parágrafos do documento. Sentenças em um parágrafo são distinguidas de acordo com a presença no começo, meio e fim do parágrafo.
- **Thematic Words**: As palavras relevantes mais freqüentes são definidas como palavras temáticas. Um pequeno número de palavras temáticas são selecionadas e cada sentença

é classificada em função da freqüência das referidas palavras. A característica é binária, sendo verdadeira para as sentenças que apresentam maior número de palavras temáticas.

- **Uppercase Word**: Pressupõe que nomes próprios são geralmente importantes para determinar a relevância de uma sentença para sumarização. Todas as sentenças são classificadas de acordo com o número de palavras em maiúscula (excluindo a primeira palavra de cada sentença e abreviaturas comuns (Kg, F,...)). A característica é binária, sendo verdadeira para as sentenças que apresentam maior número de palavras em maiúscula.

Kupiec utilizou o classificador Naive-Bayes, que calcula a probabilidade de uma sentença ser incluída no resumo como:

$$p(s \in S) \mid F_1, F_2, ..., F_k) = \frac{\prod_{j=1}^{k} P(F_j \mid s \in S) P(s \in S)}{\prod_{j=1}^{k} P(F_j)},$$

sendo:

s a sentença

S o conjunto de sentenças no sumário

F_i o valor da i-ésima característica na sentença s

K o número de características

Para calcular as probabilidades, Kupiec utilizou uma base de treinamento com 188 documentos de 21 publicações diferentes. Os documentos não continham sumários providos pelo autor, de forma que profissionais na criação de resumos produziram sumários indicativos com 3 sentenças em média. Deve-se notar que os sumários produzidos não eram extrativos e um segundo processo de obtenção de sumários extrativos teve que ser realizado em 2 etapas:

- Seleção automática: um programa verificava o alinhamento de sentenças do sumário e do documento. Desta forma, cerca de 79% das sentenças do sumário foram associadas automaticamente.

- Seleção manual: associação efetuada por juizes humanos, sendo que 9% das sentenças do resumo foram consideradas não passíveis de serem associadas a sentenças do texto.

Os resultados foram calculados utilizando validação cruzada: cada publicação era selecionada como base de teste e as demais como base de treinamento. O melhor resultado obtido foi uma taxa de acerto de 42% de similaridade entre as sentenças selecionadas pelo sistema e os sumários extrativos, utilizando uma combinação das características *Paragraph*, *Fixed-Phrase* e *Sentence Length Cut-off*. Para comparação dos resultados, Kupiec sugere a utilização das primeiras sentenças do documento que apresentam valores verdadeiros para a característica *Sentence Length Cut-off*, excluindo títulos de seções e outras informações irrelevantes. Esta técnica obteve uma taxa de acerto de 24%. A melhor característica isolada é *Paragraph*, com 33% de acerto.

Em [Mani98a] é proposto um método de sumarização baseado em aprendizagem de máquina sobre um conjunto de documentos contendo resumos fornecidos pelos autores. O sistema produz tanto sumários genéricos quanto com foco do usuário, e pretende produzir regras de sumarização que possam ser compreendidas e editáveis por humanos.

As características utilizadas foram agrupadas em 3 classes:

- Locacionais: exploram a estrutura do texto. Deve-se observar que as características deste grupo só podem ser extraídas de textos estruturados.
  - **sent-loc-para**: indica se a sentença ocorre no começo, meio ou fim do parágrafo.
  - **Para-loc-section**: indica se a sentença ocorre no começo, meio ou fim da seção.
  - **sent-special-section**: assume o valor 1 se a sentença ocorre na introdução, 2 na conclusão ou 3 em outra seção.
  - **depth-sent-section**: assume o valor 1 se a sentença ocorre em uma seção de nível 1, até 4 se a sentença ocorre em uma seção de nível 4.
- Temáticas: indicam o conteúdo temático das sentenças.
  - **Sent-in-highest-tf**: *tf* médio da sentença.
  - **Sent-in-highest-tf.idf**: *tf-idf* médio da sentença.

- **Sent-in-highest-G^2**: G^2 médio da sentença. Indica a variação da qual a freqüência de um termo no documento é maior do que o esperado da sua freqüência em toda a base de documentos.
- **sent-in-highest-title**: número de menções a títulos de seções ou termos do título do documento.
- **sent-in-highest-pname**: número de menções a nomes próprios.

- Coesão: envolvem relações entre palavras, indicando o quão conectado o texto é.
  - **sent-in-highest-syn**: número de sentenças únicas com uma ligação de sinônimos com a sentença corrente.
  - **Sent-in-highest-co-occ**: número de sentenças únicas com uma ligação de co-ocorrência de palavras com a sentença corrente.

Os resultados foram analisados em um conjunto de 198 artigos técnicos da base CM-PLG [Teufel 99], com seus sumários providos pelo autor. Os sumários extrativos foram produzidos da seguinte maneira:

- Cada sentença do texto é codificada como um vetor.
- As sentenças são então classificadas de acordo com a sua similaridade com o sumário fornecido pelo autor.
- As mais similares c% (onde c é a taxa de compressão) sentenças são selecionadas como exemplos positivos de treinamento (sentenças que fazem parte do sumário extrativo), e as demais como exemplos negativos.

O processo acima descrito resultou em 27803 vetores de treinamento, que foram reduzidos para 900 vetores únicos, com 182 positivos e o restante negativos. Dos vetores negativos, selecionou-se somente 214, para balanceamento dos dados de treinamento.

Os algoritmos de treinamento utilizados foram:

- SCDF: técnica de regressão múltipla que cria uma função linear que maximiza a discriminação entre os exemplos.

- C4.5 Rules: produz regras a partir da árvore de decisão produzida pelo C4.5 [Quinlan 93], um algoritmo bastante conhecido na área de aprendizagem de máquina.
- AQ15c: indutor de regras que otimiza as regras de acordo não apenas com sua precisão preditiva, mas também com o número e simplicidade das condições.

Os resultados foram avaliados por uma medida comumente utilizada em recuperação de informações, chamada de *F-Score* [Mani 98c]. Os valores obtidos para sumários genéricos para os referidos algoritmos foram: SCDF (62%), AQ15c(52%) e C4.5 Rules(69%).

Nas regras de classificação produzidas pelo C4.5Rules, as características temáticas e de coesão foram referenciadas principalmente em regras para a classe negativa, enquanto que as locacionais e as que envolviam freqüências dos termos nas regras da classe positiva. Os resultados do sistema praticamente não se alteraram com mudanças na taxa de compressão de 5% a 30%.

Um exemplo das regras de sumarização encontradas pelo algoritmo:

```
if the sentence is in the conclusion and it is a high tf-idf sentence
then it is a summary sentence
```

2.6 Técnicas Baseadas na Estrutura do Discurso (Coesão e Coerência)

Um sumário não é apenas um conjunto de sentenças, cada uma representando uma informação saliente do documento. Ao contrário, o documento apresenta uma organização interna, que reflete no sumário a necessidade de incorporar a estrutura do texto. Na área de sumarização, as técnicas utilizadas para capturar a estrutura do discurso do texto envolvem a distinção entre coerência e coesão.

O conceito de coesão no texto envolve a relação entre palavras e expressões indicando o grau de conexão do texto. As relações de coesão são indicadas através de anáforas, elipses, conjunções e relações léxicas como sinônimos, hiperônimos e hipônimos. Sinônimos são palavras com o mesmo sentido ou quase com o mesmo sentido, mas com formas diferentes. São sinônimas as palavras que podem substituir-se uma a outra num único enunciado isolado. Ex: Bonito, belo, lindo são sinônimos, denotativamente,

embora cada uma das palavras tenha conotações próprias. Hipônimos são palavras que se relacionam pelo sentido dentro de um conjunto, ligando-se por afinidade ou por um ser parte do outro. Assim, são hipônimos: pão em lugar de alimento (Ganharás o pão com o suor deo teu rosto.), Camões em lugar da obra (Gosto de ler Camões); almas em vez de pessoas (Havia na cidade vinte mil almas.). Quando os hipônimos de um conjunto são todos incluídos em termo de sentido que os abrange, este termo chama-se hiperônimo. Ex.: animal é hiperônimo de cão, gato, gato e burro.

O conceito de coerência envolve a descoberta de relações de argumentação entre as sentenças ou cláusulas do documento. Por exemplo, as palavras "Apesar", "Por exemplo" indicam determinados tipos de relações entre as cláusulas envolvidas. Estas relações determinam a estrutura argumentativa do texto, que são responsáveis por fazê-lo coerente.

Coesão

Algumas técnicas utilizadas para computar a coesão no texto são:

- Co-ocorrência de palavras ([Mitra 97], [Carbonell 98], [Sanderson 98]);
- Co-referência ([Baldwin 98]);
- Cadeias Léxicas ("*Lexical Chains*") ([Barzilay 97]);

Abaixo segue a descrição de alguns trabalhos envolvendo as técnicas apresentadas.

Em [Barzilay 97] é apresentada uma técnica de sumarização utilizando Cadeias Léxicas. Cadeias Léxicas são seqüências de palavras relacionadas que indicam tópicos conectados do texto. O primeiro modelo de Cadeias Léxicas proposto data de 1991, e utilizou o *Roget's Thesaurus* na construção das cadeias. O maior problema da técnica proposta é que ela não requer que as palavras apareçam no mesmo sentido.

Barzilay propôs a utilização da *WordNet* [Miller 90] na criação de cadeias léxicas que são capazes de resolver o sentido correto da palavra. A *WordNet* é uma rede semântica de representação de conhecimento contendo mais de 118.000 formas de palavras diferentes, além de ligações para sinônimos, hipônimos e hiperônimos. Cada conjunto de palavras

relacionadas semanticamente na *WordNet* é denominado *Synset*. Aproximadamente 17% das palavras da *WordNet* são polissêmicas, ou seja, apresentam mais de um sentido.

Para a criação de cadeia léxicas, o texto é primeiramente segmentado através do algoritmo *TextTiling* [Hearst 93] e são construídas cadeias para cada segmento. Para cada sentença, os substantivos simples e compostos são identificados. Na identificação dos substantivos simples, foi utilizado o *Brill Part-of-Speech* [Brill 92]. Para identificação dos substantivos compostos, Barzilay utilizou um *parser* desenvolvido na Universidade de Bar-Ilan. As cadeias são criadas através de relações divididas nas categorias:

- extra-forte: repetições da mesma palavra;
- forte: ocorre entre 2 palavras conectadas por uma relação da WordNet;
- média-forte: ocorre quando existem conexões entre os *synsets* da palavra com distância maior que 1.

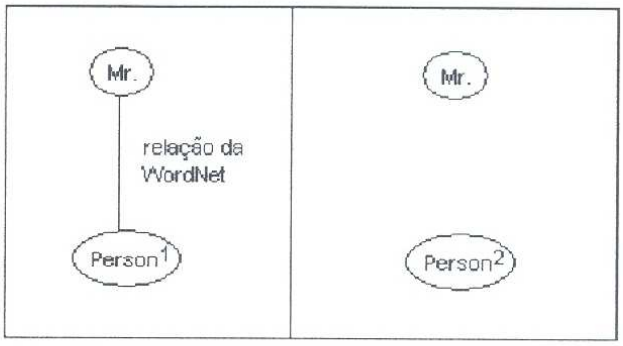
Para cada tipo de relação, existe um distância máxima para que a palavra seja considerada pertencente a mesma cadeia:

- extra-forte: sem limite;
- forte: 7 sentenças;
- média-forte: 3 sentenças.

Para demonstrar a criação de uma cadeia léxica, seja o seguinte exemplo:

Mr. Kenny is the **person** that invented an anesthetic **machine** wich uses **micro-computers** to control the rate at which an anesthetic pumped into the blood. Such **machines** are nothing new. But his **device** uses two **micro-computers** to achieve much closer monitoring of the **pump** feeding the anesthetic into the patient.

O primeira palavra encontrada é *Mr.*, que só apresenta um único sentido, segundo a WordNet. A segunda palavra é *Person*, que apresenta 2 sentidos: "*person, individual, someone*" ou "*gramatical category of pronouns and verb forms*". A escolha pelo sentido da palavra *person* divide a cadeia em 2 interpretações diferentes, sendo que no primeiro sentido da palavra *person* existe uma relação entre os termos na WordNet:



Person 1) person, individual, someone
Person 2) grammatical category of pronouns and verb forms

Figura 1 - Cadeia Léxica/Passo 1

A próxima palavra é *machine*, que tem 5 sentidos. no primeiro sentido "*an efficient person*", é relacionado aos sentidos de *person* e *Mr*, embora seja o sentido incorreto para a sentença.

Mas quando as palavras *micro-computer*, *device* e *pump* são inseridas, o número de possibilidades aumenta. As interpretações consideradas mais fortes são vistas na figura abaixo:

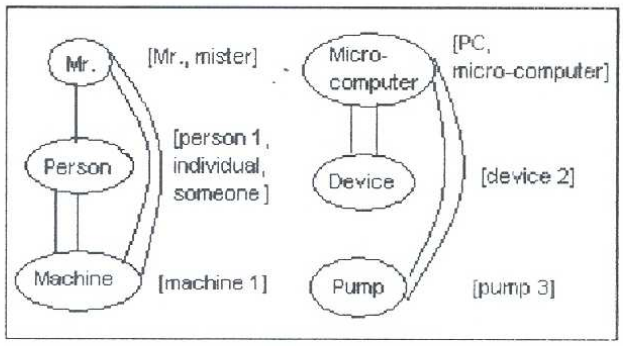


Figura 2 - Cadeia Léxica/Passo 2 - Interpretação 1

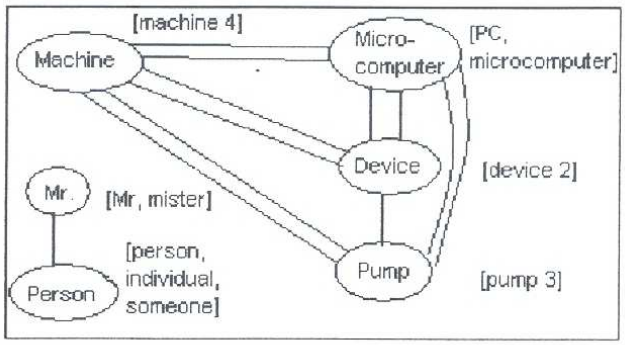


Figura 3 - Cadeia Léxica/Passo 2 - Interpretação 2

Considerando o princípio que o texto é coesivo, define-se que a melhor interpretação é a que apresenta maior número de conexões. Neste caso, a segunda interpretação é selecionada, que determina o sentido correto para a palavra *machine*. O valor de uma cadeia é determinado pelo número e peso das relações entre membros da cadeia, experimentalmente definidos como: 10 para repetições e sinônimos, 7 para antônimos e 4 para hipônimos e holônimos. O algoritmo computa todas as combinações possíveis, mantendo cada uma sem contradição. Quando o número de combinações é muito grande, acima de um determinado limiar, as interpretações fracas são podadas.

Inicialmente, as cadeias são computadas em separado. Depois, as cadeia podem ser misturadas dentro de um mesmo segmento se existirem membros de um mesmo *synset*, ou um nó é hipônimo do outro em um caminho cujo comprimento é menor que um limiar especificado. As cadeias também podem ser misturadas entre segmentos diferentes, mas somente se elas contiverem ao menos uma palavra em comum no mesmo sentido.

Para gerar sumários através de cadeias léxicas utiliza-se um algoritmo de 2 passos:

- **Passo 1**: encontrar as cadeias léxicas mais fortes.

Segundo uma análise empírica, alguns atributos considerados bons para classificar a importância de uma cadeia léxica são:

- tamanho: número de ocorrência dos membros da cadeia.
- Índice de homogeneidade: número de ocorrência distintas dos membros da cadeia dividido pelo tamanho da cadeia.

O valor total da força de uma cadeia léxica é dado por:

Força(cadeia)=tamanho x índice de homogeneidade

e as cadeias mais fortes são:

Força(cadeia) > Média(Força) + 2 * DesvioPadrão(Força)

Em um teste realizado em 5 documentos de 1055 palavras, o processo acima seleciona 5 de 32 cadeias léxicas.

- **Passo 2**: Extrair as sentenças mais significantes.

  Foram propostas 3 heurísticas:

  - Heurística 1: Para cada cadeia léxica forte, escolher a sentença que contém a primeira aparição de um membro da cadeia no texto.
  - Heurística 2: Para cada cadeia léxica forte, escolher a sentença que contém a primeira aparição de um membro representativo da cadeia no texto. Membros representativos da cadeia são palavras que apresentam freqüência na cadeia acima da média.
  - Heurística 3: Para cada cadeia léxica forte, encontrar a unidade do texto aonde a cadeia está mais concentrada e extrair a sentença com a primeira aparição da cadeia na unidade. A concentração da cadeia nas unidades é computada através do número de ocorrências dos membros da cadeia no segmento dividido pelo número de substantivos no segmento.

Todas as heurísticas selecionam somente 1 sentença por cadeia. A heurística que produziu melhores resultados foi a segunda. A terceira heurística, apesar de mais elaborada, apresentou resultados piores.

Para avaliação do método, Barzilay comparou sumários extrativos construídos por juizes humanos com o algoritmo de sumarização baseado em cadeias léxicas e com o Microsoft Word Summarizer, a ferramenta de sumarização que acompanha o Microsoft Office 97.

Para tanto, foi utilizada uma base de 40 documentos, sendo que para cada documento foram gerados 10 sumários, produzidos por 5 juizes humanos. Cada juiz produziu 2 sumários: com 10% e 20% do número de sentenças. O "resumo ideal" foi selecionado com base na votação da maioria dos juizes. Barzilay reportou uma taxa de concordância entre juizes humanos muito maior do que a média: cerca de 96% para sumários de 10% e 90% para sumários de 20%. Os números são bastante altos se comparados com outros experimentos (25% em [Mitra 97] e 71% em [Marcu 99]).

Os resultados, medidos em termos de precision e recall foram:

	Word Summarizer		Cadeias Léxicas	
	Precision	Recall	Precision	Recall
sumários 10%	33	37	61	67
Sumários 20%	32	39	47	64

Tabela 1 - Resultados do sistema de Cadeias Léxicas

Apesar do algoritmo obter bons resultados, alguns problemas permanecem, a saber:

- Sentenças longas têm maior probabilidade de serem selecionadas.
- Não existe nenhum controle no tamanho e nível de detalhe do sumário, visto que somente uma sentença é selecionada de cada cadeia léxica.

Em [Mitra 97] é proposta uma técnica de sumarização utilizando coesão baseada em parágrafos como a unidade de extração. Segundo o autor, o parágrafo provê mais contexto, sendo que os problemas de legibilidade e coerência são no mínimo amenizados. O sistema utiliza a representação vetorial de documentos, tipicamente utilizada em sistemas de recuperação de informações.

Conforme discutido na seção 2.2, cada texto ou segmento de texto é representado como um vetor de pesos/termos na forma $D_i=(d_{i1}, d_{i2}, ..., d_{it})$, onde d_{ij} representa o peso de importância do termo d_j no documento D_i. Os termos podem ser palavras ou frases e os pesos são calculados levando em conta as ocorrências dos termos no documento individual e na coleção de documentos como um todo.

Assumindo a representação vetorial, é possível calcular a similaridade entre dois vetores através do coseno do ângulo entre os mesmos, sendo que este fornece o valor 1 para vetores idênticos e 0 para vetores disjuntos.

A relação entre os parágrafos é determinada através de um mapa de relacionamentos do texto (*text relationship map*), sendo que parágrafos são nós e parágrafos relacionados são unidos por arcos. Os nós são unidos por arcos baseando-se na similaridade numérica entre cada par de parágrafos. Tipicamente usa-se um limiar e todos os parágrafos cuja similaridade exceder o limiar são conectados. Pode-se afirmar que os parágrafos conectados são "semanticamente relacionados" pois apresentam um número considerável de termos em comum.

Um exemplo de um mapa de relacionamentos do texto segue abaixo:

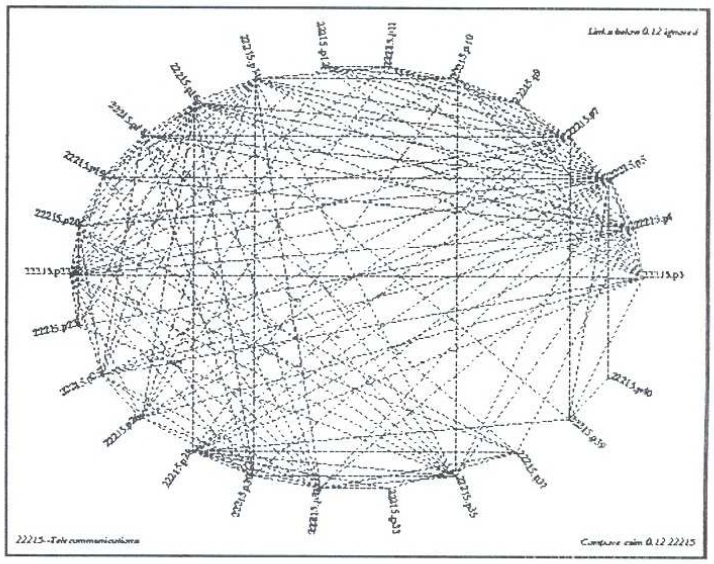


Figura 4 - Mapa de relacionamentos do texto

O mapa de relacionamentos do texto pode ser utilizado para decompor o documento em segmentos, verificando parágrafos com muitas conexões entre si, mas poucas conexões a outros parágrafos. Um exemplo de uma segmentação de texto pode ser visto abaixo:

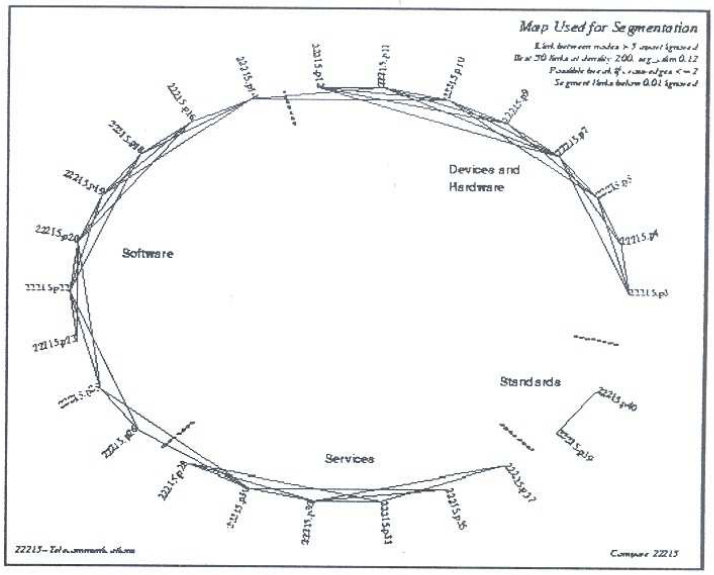


Figura 5 - Segmentação de texto

Os mapas de relacionamentos do texto podem ser utilizados na geração de sumários, através da identificação de parágrafos importantes. Vários critérios podem ser utilizados para associar importância a parágrafos. Em [Mitra 97], são sugeridos 4 caminhos no mapa que selecionam frases para o sumário. Os 4 caminhos são:

- *Global Bushy path*: Selecionar os n parágrafos mais conectados no mapa, onde n é o número de parágrafos desejados no sumário.
- *Depth-first path*: O parágrafos selecionados pelo caminho *Global Bushy path* são altamente conectados a outros parágrafos, mas não necessariamente entre si, o que pode gerar sumários incoerentes e com má legibilidade. Para minimizar o problema, o caminho *Depth-first path* seleciona um nó inicial (tipicamente o primeiro nó ou o nó mais conectado) e a cada passo visita o nó mais similar. Desta forma, o sumário não apresenta transições abruptas, mas todos os aspectos do texto podem não estar presentes no sumário.
- *Segmented Bushy Path*: Constrói *Global Bushy paths* separados para cada segmento do texto e concatena os parágrafos selecionados na ordem do texto. No mínimo 1 parágrafo é selecionado para cada segmento; o restante do resumo é formado selecionando-se mais nós altamente conectados de cada segmento na proporção do seu tamanho.

- *Augmented Segmented Bushy Path*: Seleciona sempre o primeiro parágrafo de um segmento, baseado na idéia que o autor introduz um novo assunto na primeira linha.

Os resultados foram comparados em uma base com 50 artigos de enciclopédia, com um número de parágrafos entre 13 e 48, sendo que 2 sumários manuais foram produzidos para cada documento. O sistema automático e os sumários manuais apresentavam o mesmo número de linhas. Os juizes humanos tiverem um baixo grau de concordância nos sumários extrativos gerados: somente 46% das sentenças eram iguais. Quatro métodos para analisar os resultados do sistema foram propostos:

- Avaliação Otimista: utilizar para a análise de resultados o resumo manual com o qual o sumário automático apresentou maior número de sentenças coincidentes.
- Avaliação Pessimista: utilizar para a análise de resultados o resumo manual com o qual o sumário automático apresentou menor número de sentenças coincidentes.
- Intercessão: calcular a intercessão das linhas entre os sumários manuais e computar o percentual dos parágrafos incluídos no sumário automático.
- União: calcular a união das linhas entre os sumários manuais e computar o percentual dos parágrafos incluídos no sumário automático.

Todos os sumários selecionavam a primeira linha do documento, que apresentava grande possibilidade de ser incluída no sumário. Os sumários automáticos foram comparados com sumários aleatórios e sumários que selecionavam os 20% primeiros parágrafos. Os melhores resultados foram obtidos utilizando-se o caminho *Global Bushy Path*. Entretanto, sumários obtidos selecionando-se os 20% primeiros parágrafos obtiveram resultados comparáveis ao sistema, como observado em [Brandow 94].

Em [Carbonell 98] é apresentado um método estatístico para recuperação de informações e sumarização de um ou múltiplos documentos. A técnica é denominada MMR (*Maximal Marginal Relevance*), e é basicamente um método para combinar o conceito de relevância de uma consulta com o conceito de novidade/não-redundância na informação.

O MMR no contexto da recuperação de informações é utilizado em casos aonde existe um vasto número de documentos potencialmente relevantes e redundantes uns com os outros, ou contendo informações parcialmente ou completamente duplicadas.

Um documento tem alta relevância marginal se ele é tanto relevante a uma consulta e contém mínima similaridade com documentos selecionados anteriormente. O MMR pode ser definido como:

$$MMR = Arg \max_{D_i \in R \setminus S} \left[\lambda(Sim_1(D_i, Q) - (1-\lambda) \max_{D_j \in S} Sim_2(D_i . D_j)) \right], \text{ sendo}$$

Q, uma consulta

R = IR(C,Q,θ), é a lista de documentos recuperados por um sistema de recuperação de informações, dados C, uma coleção de documentos, Q e um limiar θ, abaixo do qual os documentos não serão recuperados

S, conjunto de documentos em R já selecionados como sendo relevantes

R\S, é a diferença dos conjuntos R e S

Sim_1 é uma métrica de similaridade entre os documentos D_i e D_j.

Sim_2 pode ser a mesma métrica de similaridade ou diferente

λ, um coeficiente que regula o uso do MMR. Quando λ=1, o comportamento é similar a sistemas tradicionais de recuperação de informação. Quando λ=0, o sistema busca a máxima diversidade entre os documentos recuperados.

Um experimento foi conduzido com cinco usuários para determinar a diferença entre o método comum de classificação de documentos em recuperação de informações e o MMR para efetuar buscas de documentos. Cerca de 80% preferiram o MMR.

No contexto da sumarização de documentos, o MMR pode ser utilizado para selecionar sentenças relevantes do documento. Na SUMMAC [Mani 98c], a maior avaliação já realizada de sistemas de sumarização, a Universidade Carnegie Mellon obteve

os melhores resultados nas tarefas *AdHoc* e *Question-Answering*, que envolviam a geração de sumários baseados em consulta, com um sistema baseado no uso do MMR. O algoritmo funcionava da seguinte maneira:

- Segmentar os documentos em sentenças.
- Usar o MMR com a métrica de similaridade dos cosenos e re-classificar as sentenças em resposta a uma consulta do usuário.

O MMR também pode ser utilizado para sumários genéricos, que representam a idéia principal do documento, sem a existência de uma consulta. Neste caso, pode-se utilizar o centróide do documento como consulta, segundo recomendações do autor.

O MMR é uma técnica simples porém robusta, visando maximizar a relevância ao mesmo tempo que evita a redundância com as informações já obtidas. Deve-se notar que o MMR é mais indicado para sumarização multi-documentos, aonde o nível de redundância é muito maior.

Em [Sanderson 98] é proposto um algoritmo simples baseado em coesão para geração de sumários com foco do usuário (deseja-se sumarizar um texto com enfoque em uma consulta fornecida), que participou da SUMMAC [Mani 98c] na tarefa *Adhoc*. O algoritmo de sumarização utilizava o sistema de recuperação de informações INQUERY, que continha 2 módulos relevantes na tarefa:

- *Best Passage Operator*

Retorna a região do documento que melhor atende a uma consulta. O processo executa uma operação de recuperação similar à busca de documentos, classificando segmentos de tamanho igual, sobrepostos no documento. Para sumarização, retira-se o melhor segmento seqüencial de tamanho igual a 10% do documento.

- LCA (*Local Context Analysis*)

É uma forma de expansão automática de consulta, que obtém palavras relacionadas em uma base de documentos fornecida separadamente. Para sumarização, foi utilizada para expandir os tópicos com 70 palavras adicionais.

A base de documentos utilizada foi a seção de artigos de jornais da TREC. Antes da avaliação da SUMMAC, resultados preliminares foram obtidos em uma base de treinamento na qual o sistema foi comparado com sumários contendo os 10% iniciais de cada documento. Comprovou-se neste experimento que o LCA não melhorou os resultados do sistema. O autor especula que tal fato se deve ao fato que as consultas já eram longas (cerca de 38 palavras), ao passo que normalmente o LCA é utilizado em consultas muito menores.

Na avaliação da SUMMAC da tarefa *Adhoc*, o sumarizador obteve a terceira posição em termos de taxa de acerto, com um *f-score* (ver seção 2.3) de 59, um fato surpreendente para um sumarizador que ignora a estrutura do texto e utiliza um método muito simples para seleção de passagens no texto.

Coerência

Algumas técnicas utilizadas para computar a coerência no texto são:

- Geração de árvores da estrutura retórica do texto ([Marcu 99]).
- Obtenção da estrutura retórica com granularidade de sentenças ou segmentos baseada em treinamento ([Teufel 99] [Chuang 00]).
- Obtenção da estrutura de discurso a partir de observação empírica e preenchimento de *templates* ([Strzalkowski 99]).

Abaixo segue a descrição de alguns trabalhos envolvendo as técnicas apresentadas.

Em [Marcu 99] é proposto um sistema baseado na Teoria da Estrutura Retórica (*Rhetorical Structure Theory* (RST)), uma das mais populares teorias de discurso na pesquisa de Geração de Linguagem Natural. O centro da teoria é a noção de relações retóricas, que é a relação existente entre dois segmentos não sobrepostos de texto, chamados de Núcleos (*Nucleus*) e Satélites (*Satellite*). Núcleos expressam o que é essencial na argumentação do texto, ao contrário de satélites. Núcleos também são compreensíveis independentemente de satélites, mas não vice-versa. Como exemplo, pode-se citar a relação retórica de evidência existente entre um núcleo e o satélite: o núcleo apresenta uma informação que o escritor do texto acredita que será insuficiente para ser aceita pelo leitor.

Por conseqüência, o satélite conterá informações mais detalhadas, visando convencer o leitor da afirmação realizada. Relações retóricas podem ser expressas por árvores de estrutura retórica (*RS-Trees*), aplicando-se relações individuais a segmentos que variam em tamanho de uma cláusula ao texto completo. Uma árvore de estrutura retórica é uma árvore binária com unidades elementares como folhas, onde cada folha é um núcleo ou satélite, e nós internos, que recebem como nome a relação retórica detectada. Recentes avanços na lingüística computacional permitem derivar a estrutura retórica para textos irrestritos. Um exemplo de uma árvore de estrutura retórica pode ser visto abaixo, sendo que núcleos são representados por caixas sólidas e satélites por caixas tracejadas.

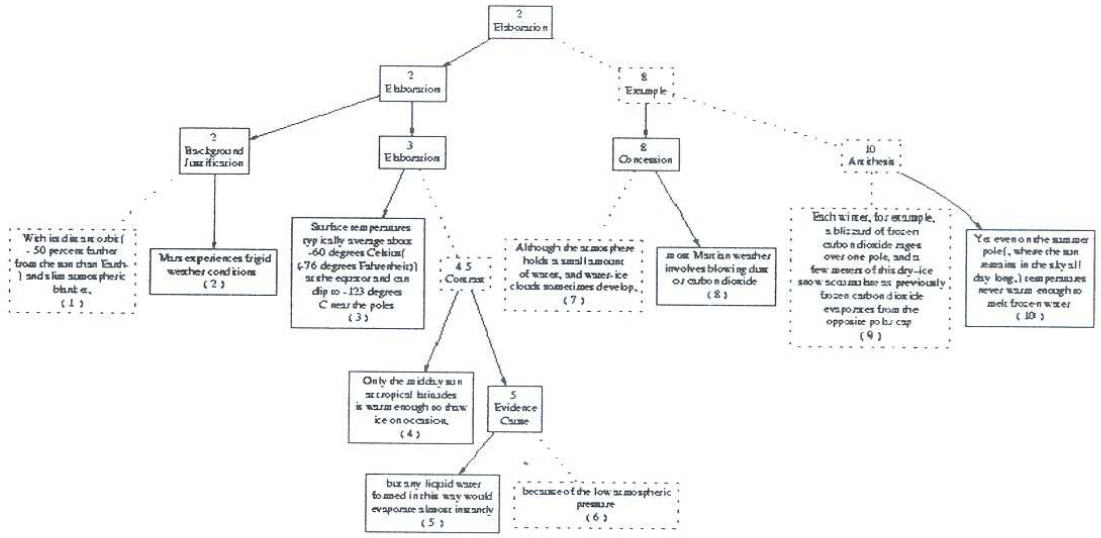


Figura 6 - Árvore de Estrutura Retórica

Cada nó interno na árvore de estrutura retórica tem um conjunto promocional (*promotion set*) que é dado pelas unidades salientes ou promocionais daquele nó. As unidades salientes são as mais importantes no segmento de texto correspondente. Elas são determinadas de maneira *bottom-up*, como se segue:

- A unidade saliente de uma folha é a própria folha.
- As unidades salientes de um nó interno são dadas pela união das unidades salientes dos filhos imediatos do referido nó que são núcleos.

Pesquisadores em lingüística computacional especulam que o núcleo da árvore de estrutura retórica constitui uma sumarização adequada do texto.

Se repetidamente aplicarmos o conceito de saliência a cada nó da estrutura, pode-se induzir uma ordem parcial de importância para todos as unidades do texto. A intuição é que unidades textuais que estão nos conjuntos promocionais dos nós do topo são mais importantes que nós encontrados na base.

O grau de importância de cada nó na estrutura pode ser definido como uma função recursiva, como se segue:

$$s(u,D,d) = \begin{cases} 0 & \text{se D é nulo} \\ d & \text{se u} \in \text{prom(D)} \\ d-1 & \text{se u} \in \text{paren(D)} \\ \max(s(u,C(D),d-1)) & \text{em caso contrário} \end{cases}$$

sendo

prom(D) : retorna o conjunto promocional do nó D

paren(D): retorna as unidades pais do nó D

C(D): retorna as sub-árvores filhas do nó D

Na árvore vista acima, a ordem de importância obtida para os nós utilizando a função é:

2 > 8 > 3, 10 > 1, 4, 5, 7, 9 >...,

significando que o nó 2 é o mais importante da árvore, seguido o nó 8, e assim por diante.

Marcu conduziu um experimento utilizando 5 artigos curtos da revista "Scientific American", considerados por ele bem escritos, com tamanhos de 161 a 725 palavras.

O autor inicialmente separou as unidades retóricas do texto manualmente e pediu a 13 juizes humanos que avaliassem a importâncias das unidades, tendo em vista um sumário potencial. Os juizes deveriam associar a cada sentença uma pontuação com 3 valores possíveis: muito importante, moderadamente importante e não importante. Adicionalmente,

foram selecionados mais 2 juizes, analistas de lingüística computacional com sólido conhecimento de estrutura retórica. Analisando-se a concordância dos juizes, obteve-se os seguinte valores:

- concordância em relação a sentenças muito importantes: 65%
- concordância em relação a sentenças moderadamente importantes: 58%
- concordância em relação a sentenças não importantes: 73%

Pode-se perceber que os juizes apresentaram um grau alto de concordância, comparado a outros experimentos como [Mitra 97], [Mani 98c] e outros.

Os resultados do sistema foram então comparados com os obtidos pelos juizes a nível de sentenças e cláusulas. Adicionalmente, o programa Microsoft Summarizer foi utilizado para comparação de base. Os resultados obtidos foram:

Sistema	Precision	Recall
Word Summarizer (cláusulas)	25,64%	27,77%
Word Summarizer(sentenças)	38,70%	41,37%
RST(cláusulas)	52,77%	50%
RST(sentenças)	65,51%	67,85%

Tabela 2 - Resultados do sistema de Árvores de Estruturas Retóricas

O experimento confirma que árvores da estrutura retórica podem ser utilizadas para extrair unidades textuais salientes em um nível comparado a humanos, e devem apresentar resultados mais "legíveis" que outros métodos, devido ao maior grau de compressão do texto, apesar de não resolver problemas como anáforas.

Em [Teufel 99], o interesse é desenvolver técnicas para sumarizar textos longos, como artigos de revistas, com 20 ou mais páginas. Nestes casos, a extração típica de sentenças não funciona bem, gerando um grande número de fragmentos desconexos. A causa do problema é que a organização lógica e retórica é perdida e se torna difícil escolher as frases potenciais para inclusão no sumário. Tendo em vista o problema apresentado, é proposta uma técnica que pretende selecionar o material para o sumário mantendo a informação sobre a estrutura retórica geral do texto. Entretanto, a estrutura retórica geral de um documento é muito complexa e difícil de modelar automaticamente, o que motivou a extração apenas da informação retórica a um nível suficiente para permitir a determinação

da contribuição retórica de todas as sentenças aptas a serem incluídas no sumário, sem modelar conhecimentos específicos de domínio. Para este fim apenas características como meta-comentários ("nós apresentamos um método para...") e estruturas retóricas a um nível macro podem ser utilizadas. Neste contexto, o sumário é construído como um *template* argumentativo, onde *slots* representam regras retóricas como GOAL, BACKGROUND, METHOD, etc. A base utilizada era uma coleção de artigos técnicos nas áreas de ciências cognitivas e lingüística computacional (CM-PLG), o que direcionou o trabalho para uma sumarização de documentos que seguem a estrutura Introdução, Método, Resultados e Discussão.

Segundo estudos, a maioria dos pesquisadores em lingüística concorda que sumários informativos de artigos técnicos devem conter:

- Propósito ou problema;
- O escopo ou metodologia;
- Os resultados;
- Conclusões e recomendações.

No trabalho, foram consideradas as seguintes unidades argumentativas:

Background, Topic/Aboutness, Related Work, Purpose/Problem, Solution/Method, Result, Conclusion/Claim.

Para construção do sumarizador automático, foi utilizado uma técnica treinável, similar a [Kupiec 95], que necessita de um conjunto de documentos que forneça respostas "corretas" a serem "aprendidas" pelo sistema, denominado *gold standard*. A tarefa de produção de sumários extrativos com anotação retórica foi realizada em 2 fases de classificação:

- Fase 1: O objetivo é extrair sentenças para composição do sumário, tentando separar sentenças que carregam regras retóricas de sentenças irrelevantes, que são a maior parte do texto, gerando um sumário intermediário.

- Fase 2: O objetivo é identificar a regra retórica correta de cada sentença candidata em uma das 7 categorias propostas.

A base de documentos utilizada apresenta sumários providos pelo autor. Porém, os sumários desejados devem ser extrativos, e somente 31% das sentenças do sumário provido pelo autor foram automaticamente alinhadas com sentenças do texto (ao contrário de 79% em [Kupiec 95]). Para correção do problema, juizes humanos selecionaram sentenças relevantes e anotaram o estado argumentativo de cada uma das sentenças do resumo extrativo. Foram gerados 2 *gold standards*:

- A, contendo somente as sentenças alinhadas automaticamente (509 sentenças).
- B, contendo as sentenças alinhadas automaticamente mais as sentenças marcadas por juizes humanos (948 sentenças).

Na base toda, existia um total de 1.172 regras retóricas para 948 sentenças, ou seja, algumas sentenças apresentavam mais de uma regra retórica.

Para treinamento do sumarizador, 7 características/heurísticas foram utilizadas: 4 similares a [Kupiec 95]: *Indicator Quality*, *Relative Location*, *Sentence Length* e *Tematic Words* e 3 novas:

- **Indicator Quality**: Indica meta-comentários no texto. Os meta-comentários foram identificados através de uma lista de 1728 frases. A maioria deles apresentam indícios positivos de inclusão no sumário, mas existem termos negativos também. Apresentava 5 valores, de –2 a 2.
- **Indicator Rhetorics**: Tenta modelar a contribuição retórica das frases. Apresenta 16 valores distintos, incluindo combinações das 7 unidades consideradas, mais o valor ZERO para frases que não predizem nenhuma regra retórica específica.
- **Header Type**: Representa a divisão retórica da sentença, especificando a divisão na qual a sentença aparece no texto ("Introdução", "Conclusão", etc). Uma lista de seções típicas foi utilizada, sendo que cada seção é associada a um dos 15 valores possíveis: 14 valores típicos e o valor "não-típico", no caso da sentença aparecer em seções que contêm nomes específicos de domínio.

Os resultados foram avaliados em relação a co-seleção entre as sentenças pertencentes aos *gold standards* e os resultados do sistema, utilizando validação cruzada. Deve-se notar que os resultados apresentam *precision* e *recall* com o mesmo valor, pois o número de sentenças extraídas é igual ao número de sentenças no *gold standard*.

Na fase 1 (extrair sentenças para o sumário), a melhor característica individual foi a *Indicator Quality*, com 54,4% de taxa de acerto. O melhor resultado foi obtido utilizando-se todas as características, com exceção da *Indicator Rhetorics*, com 66% de taxa de acerto. Como base para comparação dos resultados foi utilizada a seleção das n primeiras sentenças, que obteve uma taxa de acerto de 28%. Com relação à utilização dos dois *gold standards*, chegou-se a conclusão que sumários apresentavam qualidades similares. Na fase 2 (identificar a regra retórica de cada sentença no sumário), os melhores resultados foram obtidos com a combinação das características *Indicator Rethorics*, *Location* e *Title*, com 64,2% de taxa de acerto. Como base de comparação dos resultados foi utilizada a heurística de selecionar a regra retórica de maior ocorrência para todas as sentenças (*Solution*), que obteve uma taxa de acerto de 40%.

Em [Chuang 00] é proposto um sumarizador treinável que divide as sentenças do documento em segmentos delimitados por marcadores especiais e suas respectivas relações retóricas. O método de segmentação é o mesmo utilizado em [Marcu 99]. As características foram agrupadas em 3 grupos:

- Grupo 1: número de parágrafo, posição do segmento no parágrafo, número de palavras bônus, número de palavras do título e freqüência média dos termos.
- Grupo 2: relação retórica (antítese, causa, etc).
- Grupo 3: peso do núcleo (todas as ocorrências na qual um segmento age como núcleo), peso do satélite (idem para satélite) e nível máximo (quantas vezes um segmento é satélite de outro satélite, recursivamente).

No trabalho foram utilizados os algoritmos C4.5, Naïve-Bayes e DistAI, um algoritmo de redes neurais construtivo (a rede é construída dinamicamente, adicionando-se neurônios até a obtenção da melhor taxa de acerto).

Os resultados foram avaliados em uma base contendo 9 documentos de uma base de patentes, utilizando-se somente as seções "*Background of Invention*". Utilizou-se validação cruzada e os resultados foram comparados com os obtidos pelo Microsoft Word Summarizer e uma heurística simples que utilizava pesos fixos para cada característica. A pior taxa de acerto foi obtida pelo Microsoft Word Summarizer (60%) e a melhor pelo DistAI(78%), em média nos 9 experimentos.

Em [Strzalkowski 99] é proposto um sistema para sumarização de notícias baseado na observação empírica da estrutura de discurso dos textos, denominada DMS (*Discourse Macro Structure*). O sumário é criado para refletir os componentes do DMS, que impõe algumas restrições de discurso no processo de geração, evitando a análise completa da estrutura do documento.

O DMS parte da observação que certos tipos de textos (notícias, artigos técnicos, etc) atendem a um conjunto de restrições de estilo e organização que ajudam o autor a comunicar-se. Notícias, por exemplo, tendem a ser construídas hierarquicamente em componentes que se encaixam em 2 categorias básicas: "*what's the news*" e uma categoria "*background*" opcional.

Alguns indícios que ajudam a identificar parágrafos que são seções principais de notícias ("*what's the news*") são:

- Contém palavras que ocorrem freqüentemente no texto (alto *TF-IDF*).
- Contém palavras do título.
- Presença de substantivos compostos no início do parágrafo.
- Contém palavras que ocorrem somente em alguns parágrafos.
- Estão situados próximos ao início do documento.
- Contém nomes próprios de pessoas, locais e organizações ou números literais.
- Apresentam certas expressões como "*In Summary*", "*The point is*", etc.

Alguns indícios que ajudam a identificar parágrafos que são seções de fundo ("*background*") são:

- A presença de referências externas, como anáforas.
- A presença de nomes parciais, especialmente nomes próprios de pessoas.
- A presença de conectivos de discurso, como "*Furthermore*", "*As a result*" , "*But*" e "*Also*".
- Geralmente precedem o parágrafo da seção principal.
- Podem conter mudanças no tempo verbal ou advérbios temporais.
- Apresentam pouca ligação com os demais parágrafos em termos de vocabulário compartilhado [Mitra 97].

Considerando-se os parágrafos principais e de fundo, o algoritmo de sumarização proposto é:

1. O texto é segmento em parágrafos.
2. Uma consulta é construída contendo palavras do título do documento, palavras de conteúdo e palavras freqüentes do texto.
3. Parágrafos principais que apresentem ligações para parágrafos de fundo são unidos.
4. Todas os parágrafos são classificados e ordenados de acordo com a consulta, inclusive os unidos no passo anterior.
5. Normalize os valores obtidos (similaridade com a consulta) pelo tamanho da passagem desejada (tamanho do sumário).
6. Descarte todas as passagens com tamanho igual ou maior a 1,5 do tamanho desejado.
7. Combine as passagens em grupos de 2 ou mais, baseado no conteúdo e tamanho das passagens.
8. Re-classifique as passagens.
9. Ordene as passagens. Todas são candidatas a sumários.
10. Repetir os passos 7 à 9 até que não exista mudança nos valores obtidos (similaridade com a consulta) na classificação por 2 interações consecutivas.

Deve-se notar que o algoritmo de sumarização só adiciona a seção principal de notícia juntamente com a sua seção fundo, pois segundo o autor, o texto pode tornar-se ilegível ou incompreensível caso a seção de fundo não faça parte do sumário. Este ponto de vista difere de [Marcu 99], aonde seções com as mesmas características seriam provavelmente descartadas. Considerando-se isto, pode-se dizer que um dos objetivos principais deste método é a geração de sumários com alta legibilidade e continuidade, fator geralmente ignorado na pesquisa de sumários extrativos.

Comparação entre técnicas de Coesão e Coerência

Em [Mani 98b] foi efetuado um estudo comparativo entre técnicas de coesão e coerência, visando comparar sentenças consideradas salientes por ambos os métodos com sentenças selecionadas por juizes humanos. Os resultados mostram que apesar de técnicas baseadas em coerência apresentarem melhores resultados, o melhor método de coesão pode chegar a 76% de taxa de acerto em relação ao método baseado em coerência. Ambos os métodos apresentam correlação positiva com os julgamentos providos por juizes humanos. Entretanto, métodos baseados em coesão são muito mais "baratos" de serem obtidos, pois não requerem entendimento do texto.

Em [Harabagiu 99] é proposta uma técnica que utiliza estruturas de coesão para inferir a estrutura de coerência do texto. Ao contrário de [Marcu 99], que utilizava bases de documentos grandes e análise lingüística superficial para deduzir regras para detecção da estrutura retórica, o trabalho utiliza-se de bases lexicográficas grandes (*WordNet* [Miller 90]) e algoritmos de busca para gerar a estrutura coesiva do texto, representada através de cadeias léxicas.

A motivação de tal técnica se deve a:

- Suporte empírico que caminhos léxicos entre palavras representam um recurso útil para reconhecimento de relações de coerência.
- Em [Marcu 99] supõem-se que marcadores de discurso ocorrem freqüentemente no texto.

- O método é orientado a dados, sem a necessidade de intuição humana na qual são baseados os estudos em coerência.
- A estrutura gerada não é hierárquica.

Para associar relações retóricas a caminhos léxicos utilizou-se uma base de 30 textos do *Wall Street Journal* aonde marcadores de discurso foram manualmente nomeados com as relações retóricas. É então efetuada uma busca de padrões envolvendo a classe gramatical das palavras (*part-of-speech*) e as relações encontradas na *WordNet*.

Além dos marcadores de discurso, outras associações foram encontradas nos caminhos léxicos, derivadas de 3 categorias de relações de coerência: semelhança, causa-efeito e contiguidade.

Na avaliação de sistema, utilizou-se 40 textos do *Wall Street Journal*, da base do *Treebank Project*. Para cada texto, 3 analistas construíram a estrutura do discurso, tendo acesso ao caminhos léxicos, associando aos marcadores de discurso as relações de coerência. Utilizou-se o voto da maioria (pelo menos 2 juizes deveriam ter escolhido a mesma relação). O algoritmo encontrou cerca de 80% das relações, mas somente 50% delas estavam corretas. Comparado com o algoritmo desenvolvido por [Marcu 99], a concordância foi de 80% das estruturas de coerência idênticas.

2.7 Resultados importantes na sumarização de textos

Alguns resultados anteriores obtidos na área devem ser considerados em novas pesquisas, pois estabeleceram critérios e limitações que auxiliam a avaliação e interpretação dos resultados obtidos por novos sistemas. Segue-se algumas constatações importantes já realizadas.

Em [Rath 61], um experimento foi conduzido com 6 pessoas, que foram solicitadas a selecionar as 20 sentenças mais representativas de 10 artigos da revista *Scientific American*. As 6 pessoas concordaram em média em apenas 1,6 sentenças (8% de concordância) e em 6,4 sentenças (32% de concordância) utilizando a maioria de 5 das 6 pessoas. Em contraste, 5 métodos automáticos de sumarização concordaram em média em

9,2 sentenças das 20 e 17 das 20 utilizando a maioria de 4 dos 5 métodos automáticos. A concordância entre humanos e sistemas é muito baixa.

Em um segundo experimento, foi conduzido um teste com 5 estudantes em extrair as 20 sentenças mais representativas de 6 artigos da revista *Scientific American*. 8 semanas depois este teste foi repetido, e os estudantes foram instruídos a selecionarem as mesmas sentenças. Os estudantes selecionaram as mesmas sentenças em somente 55% dos casos, e identificaram as sentenças selecionadas anteriormente em 64% dos casos. A conclusão do autor é que aparentemente existem muitas sentenças igualmente representativas para um determinado documento.

Em [Brandow 94], desenvolveu-se o sistema ANES e foi conduzida uma avaliação intrínseca na qual os juizes eram encarregados de avaliar a aceitabilidade dos sumários obtidos de 250 artigos de notícias, totalizando um total de 750 sumários (um dos maiores experimentos já realizados na área) . O julgamento foi feito em base na fluência e abrangência do sumário. Os sumários tinham 50, 150 e 250 palavras de tamanho. ANES gerou sumários aceitáveis somente em 68% a 78% dos casos. Em contraste, sumários produzidos selecionando-se somente a parte inicial do texto foram considerados aceitáveis em 87% a 96% dos casos, o que comprova que técnicas triviais podem superar os sumarizadores automáticos em algumas situações.

Em [Morris 92], o autor realizou um experimento na tarefa de perguntas e respostas. O autor utilizou 4 exercícios com 8 questões de múltipla escolha, com uma única resposta correta por questão. Foram realizados experiências com relação a respostas de indivíduos que leram o texto original, sumários extrativos de 20% e 30%, um sumário manual informativo (produzido por profissionais em sumarização, com cerca de 25% de compressão), sem nenhum texto e uma seleção aleatória. Os resultados mostram que a avaliação dos sumários extrativos de 20% a 30% e o sumário manual informativo são comparáveis ao texto original, sendo que somente os casos sem nenhum texto e seleção aleatória apresentaram resultados significantemente baixos.

Tal experimento demonstra a eficiência de sumários extrativos na realização de tarefas aonde o texto completo é normalmente utilizado.

Em [Mochizuki 00] foi realizada uma comparação de 10 algoritmos de sumarização na realização de uma tarefa extrínseca (determinação da relevância do texto a um tópico, similar à tarefa *Adhoc* da SUMMAC). Além disso, as saídas dos diferentes algoritmos foram comparadas entre si, visando determinar algoritmos similares. A avaliação utilizou sumários de 20% do tamanho do texto na base BMIR-J2, uma base de testes para sistemas de recuperação de informação para a língua japonesa. Cerca de 30 juizes participaram da avaliação, composta de 10 consultas e uma lista de sumários. Os critérios utilizados para a avaliação foram: taxa de acerto, tempo requerido, número de vezes que os juizes necessitaram de ajuda do texto completo e legibilidade dos sumários.

Os 10 métodos utilizados foram:

- 1 - Utilizar o texto completo.
- 2 - Utilizar o título do documento.
- 3 - Extrair as primeiras sentenças do documento.
- 4 - Extrair parágrafos relacionados com a consulta, utilizando a similaridade dos cosenos.
- 5 - Extrair sentenças com a maior soma de pesos *TF-IDF*.
- 6 - Produzir sumários com foco do usuário extraindo sentenças com maior soma dos pesos *TF-IDF*, fornecendo maior peso a termos da consulta.
- 7 - Produzir sumários extraindo sentenças baseadas em cadeias léxicas.
- 8 - Produzir sumários com foco do usuário extraindo sentenças baseadas em cadeias léxicas, fornecendo maior peso a termos da consulta.
- 9 - Usar um software comercial de sumarização, incluído em um processador de textos japonês.
- 10 - Extrair passagens usando cadeias léxicas buscando inicialmente as que contenham termos da consulta. A classificação das sentenças se dá de acordo com a pontuação das cadeias incluídas nas passagens e o grau de sobreposição das cadeias.

Os resultados obtidos foram:

- Em *f-score* médio (ver seção 2.2), os métodos 9 e 10 foram melhor que o texto completo.
- Em termos de tempo de leitura, pode-se dizer que todos os métodos exceto o primeiro economizam tempo. O melhor tempo de leitura foi o do método 2, enquanto que os demais demonstraram tempos similares.
- Pode dizer que o método 2 (título do documento) foi o melhor método no geral, pois é o mais rápido e apenas um pouco inferior do que o método 10 em termos de taxa de acerto. Entretanto, nem sempre o título do documento está disponível.
- Em termos de legibilidade, os métodos 3, 7 e 9 obtiveram melhores resultados que o texto completo. O método 10 teve o mesmo resultado que o texto completo. Todos os outros foram inferiores.

Os métodos foram agrupados entre si, construindo-se *clusters* de vetores binários, representando as sentenças selecionadas no documento, através da distância dos cosenos. Os tipos 1 e 2 foram desconsiderados, por não poderem ser comparados com os demais. Foram encontrados 3 grupos:

- Grupo 1: sumários com alta contigüidade entre as sentenças (3, 9, 10)
- Grupo 2: sumários com baixa contigüidade (5, 6, 7, 8) .Cabe-se destacar a pequena variação entre métodos baseados em cadeias léxicas e freqüência dos termos (*TF*).
- Grupo 3: sumários com alta contigüidade, mas com sentenças muito diferentes do Grupo 1 (4).

TIPSTER SUMMAC

O *TIPSTER Text Summarization Evaluation* (SUMMAC) [Mani 98c] foi uma experiência conduzida pelo governo americano para avaliar sistemas automáticos de sumarização de textos, e representa um passo significante no campo. O experimento foi conduzido em maio de 1998, como uma extensão do programa TIPSTER, que visa estimular a pesquisa no estado da arte nas áreas de recuperação de informações e extração

de informações. A avaliação foi aberta a todos os sistemas que desejassem participar, e 16 sistemas efetivamente participaram da avaliação, entre eles: *CGI/CMU* (*Carnegie-Mellon University*), *Cornell University*, *SabIR Research* e *GE Research and Development*.

As bases de dados utilizadas continham na maioria artigos de jornais (notícias), incluindo seções como "Cartas ao Editor". Os dados foram reutilizados da conferência TREC (*Text Retrieval Conference*) e eram reconhecidamente limitados em termos de domínio. Entretanto, o comitê organizador afirma que não havia tempo hábil para construção de uma nova base específica para sumarização.

Os documentos estavam no formato XML e não tinham títulos, e outras *tags* informativas; somente a *tag* TEXT era permitida.

Basicamente 2 tipos de sumários foram utilizados:

- **Fixed-length**: sumários limitados a 10% do número de caracteres total do texto, excluindo espaços.
- **Best**: Sumários de tamanho variado, considerados pelos sistemas como o sumário de tamanho ideal para o texto analisado.

Os participantes tiveram acesso a uma base de treinamento vários meses antes da realização da avaliação. Os participantes podiam submeter exemplos que eram validados. O código que efetuava a contagem de caracteres foi fornecido pelos organizadores, evitando confusões.

Os resumos foram comparados com sumários contendo os primeiros 10% do texto, visto que em [Brandow 94], tais sumários apresentaram melhores resultados que técnicas mais elaboradas.

Os documentos foram avaliados em 4 tarefas, 2 extrínsecas e 2 intrínsecas.

As tarefas foram:

- -**Adhoc**: O foco era em sumários indicativos e com foco no usuário, onde sumários eram guiados por tópicos. O tópico era provido como entrada para o sistema de sumarização e a avaliação determinava se o sumário era efetivo em determinar a

relevância do documento. A base da tarefa *Adhoc* continha 20 tópicos, com aproximadamente 50 documentos por tópico.

- **-Categorization**: O foco era em sumários genéricos, e na verificação se os sumários eram efetivos para categorizar um documento. Os juizes deveriam ler os sumários e utilizá-los para classificar o documento em 5 categorias, ou nenhuma das anteriores. Cada documento pertencia a somente uma das categorias. A base da tarefa *Categorization* apresentava 10 tópicos, com aproximadamente 100 documentos por tópico.

- **Question-Answering**: O foco era em medir a relevância de sumários na resposta de um conjunto de questões de um determinado tópico. O desafio era produzir sumários informativos que abrangessem todos os aspectos do tópico no menor tamanho possível. Foi utilizados um subconjunto de 3 dos 20 tópicos da tarefa *Adhoc*, com cerca de 30 documentos de cada tópico .

- **Acceptability**: O foco era em medir a aceitabilidade de um sumário em relação ao documento, similar a [Brandow 94], utilizando sumários do tipo *"best"*, ou seja, com comprimento variável. O juiz era indagado a responder se o sumário era "bom" ou não, sem que um critério específico de qualidade fosse fornecido. Foram utilizados cerca de 30 documentos selecionados aleatoriamente da base da tarefa *Categorization* (3 documentos por tópico).

A equipe de avaliação era composta de 51 profissionais em análise de informação, que trabalharam durante 1 mês, com cerca de 16 a 20 horas por pessoa.

Os resultados obtidos foram:

Na tarefa *Adhoc*, os sistemas apresentaram resultados muito similares, sendo que o *F-score* variou entre 72 para o melhor sistema (CGI/CMU) e 56 para o pior (ISI). A concordância entre os 14 juizes foi baixa: cerca de 166 dos 1000 documentos analisados (16,6%).

Na tarefa *Categorization*, os sistemas apresentaram resultados mais próximos ainda, com valores de *F-Score* variando de 54 para o melhor sistema (CIR) a 47 para o pior

(CGI/CMU). Os resultados foram baixos e argumentou-se que tratava-se de uma tarefa difícil, pois nem os documentos originais apresentam resultados altos. A concordância entre os juizes também foi baixa: cerca de 19,5%.

Na tarefa *Acceptability*, 4 juizes analisaram cada um dos 420 sumários produzidos. Como os sumários eram do tipo *"best"*, foi introduzido uma medida normalizada que considerava o tamanho do sumário na nota final. Muitos resumos considerados aceitáveis na tarefa de categorização não foram considerados como "aceitáveis" pelos juizes. Cerca de 60% dos resumos foram considerados inaceitáveis. Os resultados normalizados variaram de 45 para o melhor sistema (BT) a 8 para o pior (ISI).

Na tarefa de *Question-answering*, trechos do documento foram considerados como respostas das questões (*answer key*). Desta forma, a avaliação do sistema é relacionada a quantas passagens corretas foram obtidas. Entretanto, a resposta não era objetiva, pois um documento poderia conter a mesma informação em mais de um lugar, ou prover versões mais ou menos específicas da informação. Uma pessoa foi designada como avaliador do tópico, cuja função era preparar perguntas e respostas no texto. As respostas a cada questão foram julgadas como corretas, parcialmente corretas ou perdidas. Os julgamentos foram expressos em ARA (*Answer Recall Average*) e o melhor resultado foi obtido pelo sistema *CGI/CMU*, com 0.71 e o pior, 0.30 para o sistema *Penn*.

No geral, os resultados obtidos foram:

- Sumários de tamanho variável economizam tempo e são tão corretos quanto o texto original.
- Mais de 80% dos sumários produzidos pelos sistemas são inteligíveis.
- Mesmo considerando a grande variedade de técnicas utilizadas, os sistemas são muito similares em termos de acerto.
- Sumários fixos com 10% dos caracteres do texto apresentaram resultados piores que sumários de tamanho variável.
- Os sistemas mais próximos do ideal em termos de acerto e tempo apresentam compressão menor que 20%.

- Existiram grandes variações na aceitabilidade dos sumários produzidos por diversos participantes, mesmo os sumários sendo muito similares na capacidade de realizarem outras tarefas.

- Sumários da tarefa *Adhoc* foram capazes de selecionar passagens do documento que eram relevantes para os tópicos, auxiliando juizes da SUMMAC a selecionar documentos que não foram considerados relevantes aos juizes da TREC (que não tinham ferramentas de sumarização).

- Os julgamentos individuais dos juizes apresentam grau médio a alto de confiança, embora exista um nível moderado de discordância entre juizes nas tarefas *Adhoc* e *Categorization*.

As lições aprendidas foram:

- Argumenta-se que o maior problema foi que a avaliação dos sistemas de sumarização foi restrita a textos curtos de jornais, visto que textos maiores e de diversos gêneros requerem soluções de sumarização mais flexíveis, e talvez exponham maiores diferenças entre os sistemas. Outro ponto salientado foi o grande uso de julgamentos humanos na avaliação, que impedem desenvolvedores de sistemas de continuamente testarem e melhorarem os resultados. Outro resultado não esperado foi a grande variabilidade nos julgamentos, mais pronunciado que outras avaliações, como a TREC.

3 Desenvolvendo Novas Técnicas para Sumarização Automática de Textos

Considerando-se a Revisão de Literatura, pode-se determinar grupos de heurísticas/características normalmente utilizadas em sistemas de sumarização. Na seção 3.1 as heurísticas serão agrupadas e seus potenciais para a tarefa de sumarização serão desenvolvidos, visando à construção de sistemas de sumarização. Em seguida, na seção 3.2, serão introduzidas quatro novas técnicas para sumarização de textos, explorando as heurísticas/características identificadas na seção 3.1.

3.1 Análise das Heurísticas

Os principais grupos de heurísticas identificados são:

- Informações temáticas do texto [Edmundson 69] [Kupiec 95].
- Métricas de Recuperação de informações [Luhn 58] [Banko 99] [Mani 98a].
- Posição e tamanho das sentenças [Edmundson 69] [Kupiec 95] [Teufel 99] [Mani 98a].
- Coesão do texto [Mitra 97] [Carbonell 98] [Barzilay 97] [Mani 98a].
- Estrutura retórica do texto [Teufel 99].
- Análise da composição morfológica do texto [Carbonell 99].
- Análise semântica das palavras do texto [Lin 95].

Deve-se salientar que todas as características foram analisadas considerando-se a restrição de serem passíveis de serem obtidas de textos não-estruturados, de qualquer domínio e gênero.

Informações temáticas do texto

Apesar de constatado em [Edmundson 69] [Kupiec 95] que palavras temáticas/chaves do texto apresentaram os piores resultados isoladamente, não sendo incluídas no conjunto de características com melhores resultados, pode-se argumentar que o método utilizado para a obtenção das mesmas era inadequado: somente a freqüência das palavras foi considerada como determinante. Em [Teufel 99], as palavras temáticas foram

obtidas através da métrica *TF-IDF* e auxiliaram (em pouca quantidade) a performance do sistema, sendo incluídas no conjunto de características com melhores resultados, apesar de apresentarem a menor performance individualmente. No presente trabalho, utiliza-se palavras temáticas dentro de um enfoque mais robusto: palavras-chaves serão obtidas através de técnicas treináveis para obtenção de *key-phrases* de um documento, como visto em [Turney 00] e [Nevill-Manning 99]. Após obtidas, as palavras-chaves serão utilizadas na criação de uma consulta a cada sentença do documento, através da similaridade dos cosenos.

Deve-se notar que características como a presença de *Cue Words/Fixed Phrases*, apesar de apresentarem bons resultados em [Edmundson 69] e em [Kupiec 95] (segunda melhor característica) não são viáveis, pois apresentam dependência de domínio e gênero do texto e portanto serão descartadas do presente trabalho.

Métricas de Recuperação de informações

Desde o primeiro trabalho realizado na área [Luhn 58], a pesquisa em sumarização sempre utilizou estatísticas baseadas na freqüência de palavras ou técnicas de recuperação de passagens no texto.

Em [Mani 98a] foram utilizadas diversas características para treinamento envolvendo métricas comuns de sistemas de recuperação de informação. Particularmente, foram utilizados os valores médios de *TF*, *TF-IDF* além do G^2 médio, que indicava se a freqüência de um termo do documento é maior que o esperado da sua freqüência no conjunto de documentos. As características em questão apresentaram bom desempenho no sistema. Entretanto, métricas mais sofisticadas permanecem inexploradas em sistemas de sumarização baseados em treinamento, sendo utilizadas somente em heurísticas fixas, que limitam o funcionamento da técnica à compreensão do autor no processo de sumarização. Algumas métricas mais sofisticadas que poderiam ser utilizadas são pesos *TF-TL* (que foram utilizados para alinhamento de sentenças de sumários providos pelo autor com sumários extrativos) e *MMR*, que visa maximizar a relevância ao mesmo tempo que evita a redundância com as informações já obtidas.

Posição e tamanho das sentenças

Características envolvendo a posição da sentença no texto ou no parágrafo apresentaram boa performance na maioria dos sistemas [Edmundson 69] [Kupiec 95] [Teufel 99] [Mani 98a]. Neste trabalho serão feitos alguns experimentos envolvendo a característica proposta em [Nevill-Manning 99] para extração de *key-phrases*: utilizar o percentil da sentença; isto é, a posição da sentença do documento em termos percentuais. Esta informação é menos rica da que a utilizada em demais trabalhos, que tipicamente envolvem a posição da sentença nos parágrafos/seções ou em seções conhecidas (como Introdução ou Conclusão). Entretanto, ela é passível de ser obtida de qualquer texto não-estruturado, fator determinante para o trabalho em questão.

Coesão do texto

Diversas técnicas utilizadas em sistemas de sumarização baseiam-se na idéia que o texto é um conjunto coeso de palavras, e que a relevância de sentenças no sumário pode ser obtida através do estudo de segmentos coesos do texto. A maioria das técnicas apresentaram bons resultados, sendo que somente [Mani 98a] apresentou resultados negativos: características de coesão contribuíram pouco na performance geral do sistema. Algumas das características sugeridas para representar a coesão no presente trabalho são:

- conexão entre parágrafos: utiliza a mesma idéia do caminho Global Bushy Path, proposto em [Mitra 97].
- cálculo da similaridade de cada sentença com o centróide do documento: Pressupõe que sentenças coesas apresentam alta similaridade com o centróide do documento ou de segmentos do documento. Em [Radev 00] técnicas similares são usadas para sumarização multi-documentos.

Estrutura retórica do texto

A análise completa da estrutura retórica do texto é muito complexa de ser obtida e requer completo entendimento do texto. Porém, métodos baseados em entendimento

superficial obtiveram sumários de boa qualidade. Em [Teufel 99] os sumários são analisados de maneira a preservar a estrutura organizacional do texto.

Algumas técnicas de coesão podem ser utilizadas para obter a estrutura organizacional do documento de forma linear, como visto no algoritmo *Text Tiling* [Hearst 93]. O *Text Tiling* apresenta bons resultados, como vistos no artigo a seguir (linhas pretas representam as divisões tópicas encontradas pelo algoritmo *Text Tiling*):

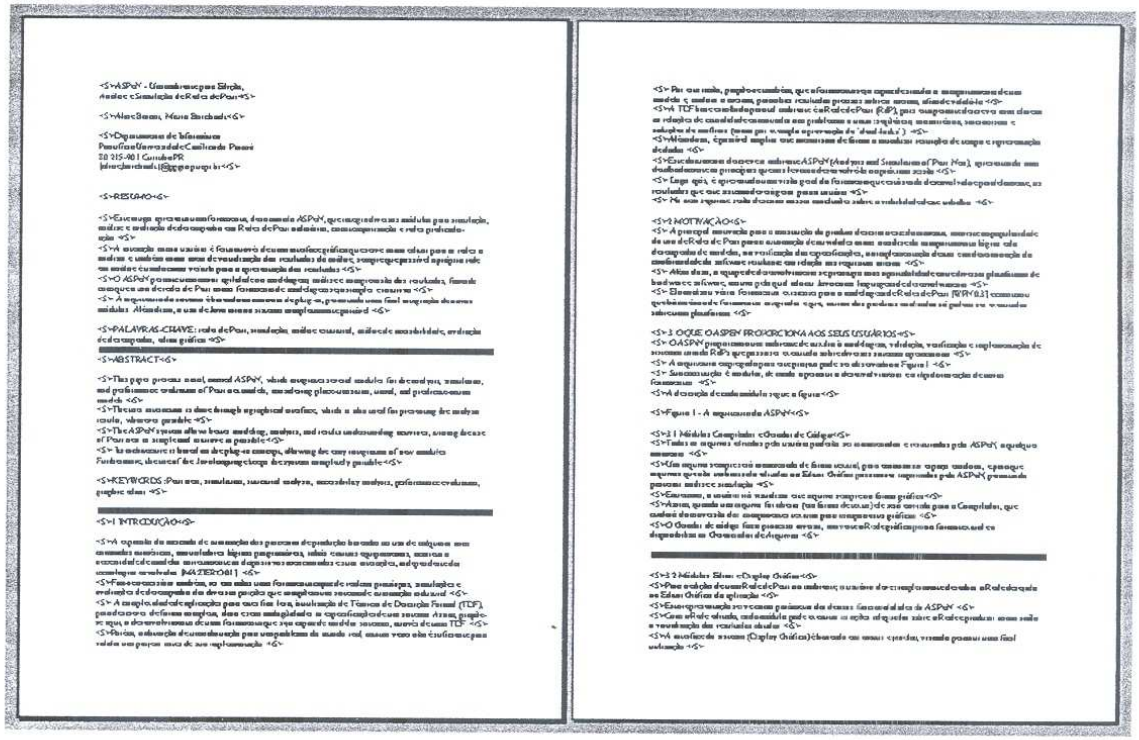


Figura 7 - Divisões tópicas encontradas pelo Algoritmo *TextTiling*

Apesar de nem toda a estrutura organizacional do documento ser capturada, os resultados são bons o suficiente para serem utilizados para dividir o texto em segmentos coerentes.

A proposta do presente trabalho é tentar extrair a referida estrutura utilizando árvores geradas através de *clustering* aglomerativo, como visto em [Yaari 97]. A árvore obtida utiliza técnicas de coesão, pois segmentos de texto similares são repetidamente agrupados pela sua similaridade léxica, até o nível mais alto (todo o documento). O algoritmo funciona da seguinte maneira:

- Cada unidade elementar (geralmente parágrafos) é representada como um vetor de pesos *TF-IDF*.
- A partir do primeiro nível (aonde todos os parágrafos são vetores isolados), aplica-se um teste de proximidade entre cada parágrafo e seus vizinhos esquerdo e direito, e une-se o parágrafo com seu vizinho mais similar. O processo é repetido até que todos os parágrafos estejam unidos.

O teste de proximidade utilizado é a conhecida similaridade dos cosenos. Algoritmos comuns de *clustering* aglomerativo têm complexidade da ordem de $O(n^2)$. Entretanto, o algoritmo proposto, por computar o teste de similaridade somente com os parágrafos vizinhos, apresenta complexidade O(n).

Um exemplo do algoritmo aplicado a um texto de notícias extraído da base da tarefa *Categorization* da SUMMAC [Mani 98c] pode ser visto na figura abaixo.

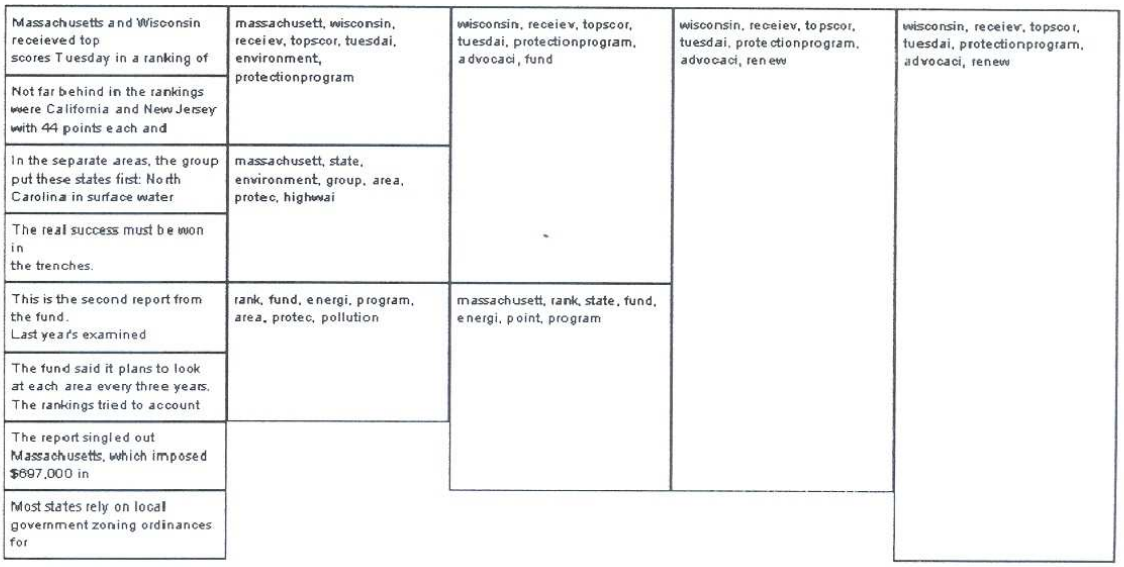


Figura 8 - Resultados do Clustering Aglomerativo em um Documento

No desenho, grupos de 2 parágrafos do texto são representados como caixas brancas; enquanto que caixas cinzas representam agrupamentos de parágrafos (*clusters*) e contêm o conjunto de palavras com maior ocorrência no agrupamento em questão. O *clustering* aglomerativo da figura acima apresenta 5 níveis. Utilizando a representação de árvore binária, a figura ficaria como:

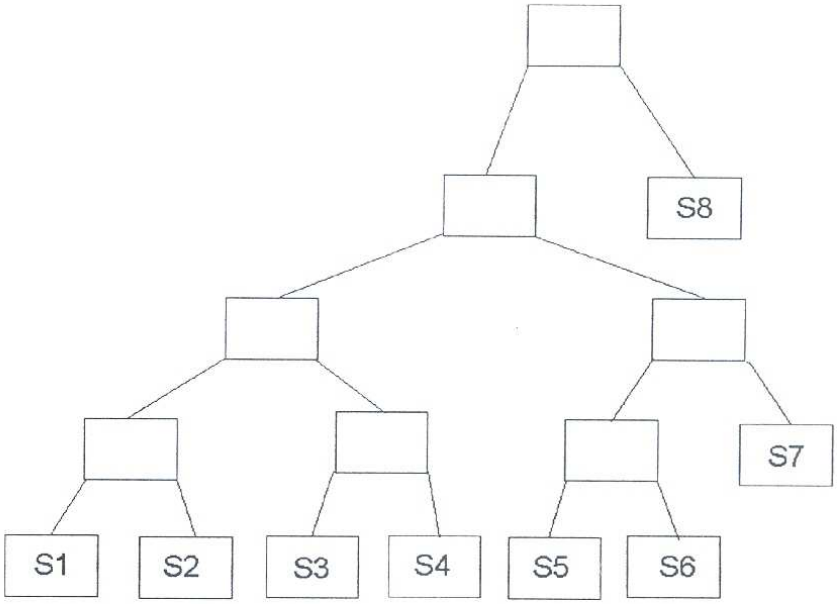


Figura 9 - Árvore Binária gerada a partir do *Clustering* Aglomerativo

Uma outra idéia que poderia ser introduzida na árvore é o conceito de saliência do nó: "Em cada nível da árvore, calcular a saliência do nó em relação ao nó pai".

O valor da saliência poderia ser calculado através de alguma métrica como o *MMR* [Carbonell 98], que considera a similaridade do nó com seu antecessor, ao mesmo tempo que evita a redundância com relação a seus nós irmãos. Se um limiar fosse aplicado no valor de saliência do nó, de forma que abaixo de um determinado valor o nó fosse considerado redundante, teríamos, por exemplo, estruturas como a vista na figura abaixo:

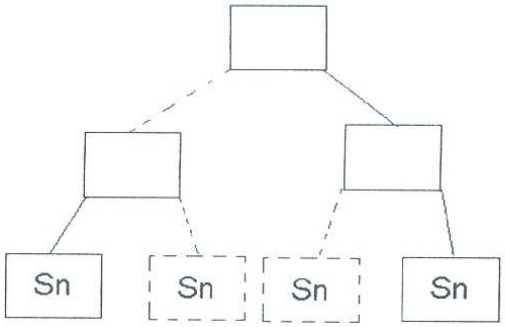


Figura 10 - Ramo de árvore com nós salientes

A partir da profundidade do nó na árvore e da medida de saliência, poder-se-ia determinar uma ordem parcial entre todos os nós, o que geraria valores de relevância para possível inclusão da sentença no sumário.

Nós com maior profundidade são a princípio mais redundantes, de forma que a relevância da sentença seria inversamente proporcional a profundidade.

Deve-se notar que a idéia proposta é por si só um algoritmo de sumarização, mas pode ser combinada com outras características.

Em [Mani 98b] as árvores produzidas através de *clustering* aglomerativo e árvores de estrutura retórica [Marcu 99] foram comparadas para 7 documentos. O resultados demonstram que as árvores produzidas pelo *clustering* aglomerativo apresentam uma segmentação de granularidade alta (nível de parágrafo) da estrutura tópica do texto, enquanto árvores da estrutura retórica apresentam uma segmentação de granularidade fina (nível de cláusula). Para os teste efetuados, árvores geradas por coerência tem muito menos variabilidade em número de folhas. As árvores são de um modo geral muito diferentes, mas ocasionalmente existem alguns pontos de similaridade.

Análise da composição morfológica do texto

Algoritmos de *part-of-speech* [Brill 92] permitem determinar a classe gramatical de palavras em uma sentença (substantivo, adjetivo, verbo, artigo, etc). Em [Carbonell 99] foi realizada uma análise empírica das propriedades de sumários utilizando-se uma base de notícias com artigos do *Los Angeles Times* e *Reuters*. Na análise constatou-se diversos padrões, como:

- Sentenças do sumário contêm mais artigos indefinidos que sentenças que não pertencem ao sumário.
- Pronomes pessoais, títulos como "*Dr.*", "*Mr.*" e "*Mrs.*", negações, verbos auxiliares, números ordinais escritos com dígitos ou palavras, conjunções e preposições tendem a ser bons indícios de evidência negativa no sumário.

Desta forma, a classe gramatical das palavras parece estar altamente relacionada com a relevância da sentença no sumário. Neste contexto, propõe-se a criação de

características que expressem a proporção de cada classe gramatical na sentença, pois estas aparentemente são boas discriminadoras entre sentenças que devem e que não devem pertencer ao sumário.

Análise Semântica das Palavras do Texto

A *WordNet* é um dicionário contendo substantivos, verbos, adjetivos e advérbios. Os substantivos em especial são representados como uma rede semântica de conceitos.

Uma busca por hiperônimos (generalizações) da palavra "*computer*" na *WordNet* retornaria:

```
Sense 1
computer, data processor, electronic computer, information processing system
    => machine
      => device
        => instrumentality, instrumentation
          => artifact, artefact
            => object, physical object
              => entity, something

Sense 2
calculator, reckoner, figurer, estimator, computer
    => expert
      => person, individual, someone, somebody, mortal, human, soul
        => life form, organism, being, living thing
          => entity, something
        => causal agent, cause, causal agency
          => entity, something
```

Deve-se notar que a *WordNet* contém os diversos sentidos que uma palavra apresenta. Uma busca por hipônimos (especializações) da palavra "*computer*" na *WordNet* retornaria:

```
Sense 1
computer, data processor, electronic computer, information processing system
    => analog computer, analogue computer
    => digital computer
    => node, client, guest
    => number cruncher
    => pari-mutuel machine, totalizer, totaliser, totalizator, totalisator
    => server, host

Sense 2
calculator, reckoner, figurer, estimator, computer
```

```
=> number cruncher
=> statistician, actuary
```

Em [Lin 95], a *WordNet* foi utilizada para identificar as idéias centrais do texto através de generalizações e especializações na hierarquia de substantivos. Controlando-se um determinado limiar pode-se variar o nível de generalidade dos conceitos extraídos do texto. A *WordNet* também foi utilizada na obtenção de cadeias léxicas [Barzilay 97].

Como heurística para sumarização de textos, a *WordNet* poderia ser utilizada para fornecer a densidade dos conceitos principais do texto contido em cada sentença. Ao invés de uma simples comparação de palavras, operações de generalização e especialização seriam utilizadas para fornecer pesos intermediários a palavras normalmente consideradas distintas por métodos como similaridade de vetores de pesos *TF-IDF*. O processo poderia ser descrito como uma "busca semântica de conceitos na sentença".

3.2 Os 4 sistemas de Sumarização Propostos

Selecionando-se algumas das possibilidades enumeradas acima, foram desenvolvidos 4 sistemas de sumarização: 2 baseados em heurísticas fixas e 2 baseados em treinamento. Os sistemas apresentam grau crescente de complexidade. Segue-se uma descrição detalhada dos sistemas.

Sistema 1: Sumarização de textos baseada em pesos TF-ISF

O sistema funciona da seguinte maneira [Larocca 00a]:

- Inicialmente, as sentenças são separadas do documento. O fim da sentença é definido como um ".", seguido de espaço ou avanço de linha, desde que a sentença apresente um tamanho maior ou igual a um tamanho mínimo especificado pelo usuário (parâmetro "*Min Line Size*").
- Os pré-processamentos tradicionais são aplicados nas sentenças: *case-folding*, remoção de *stopwords* e *stemming* [Witten 94]. Para remoção de *stopwords*, utilizou-se uma lista de 524 palavras, obtidas do código fonte da biblioteca

BOW, desenvolvida pela Universidade de *Carnegie Mellon*. Para efetuar o processo de *stemming*, utilizou-se o conhecido algoritmo de Porter [Porter 80].

- Cada sentença é representada como um vetor de pesos *TF-ISF* (*term frequency - inverse sentence frequency*). A computação do *TF-ISF* para cada palavra é similar à computação do *TF-IDF* [Salton88]. A diferença é que a noção de "documento" do *TF-IDF* é substituída pela noção de sentença. Similarmente, o "número de documentos" é substituído pelo número de sentenças no documento. Então a medida *TF-ISF* de uma palavra w na sentença s, denotada por *TF-ISF(w,s)* é computada pela seguinte fórmula:

$$TF\text{-}ISF(w,s) = TF(w,s) * ISF(w)$$

sendo,

TF(w,s): o número de vezes que a palavra w ocorre na sentença s.

ISF(w): é dada pela fórmula:

$$ISF(w) = \log(|S| / SF(w)),$$

sendo,
SF(w): número de sentenças na qual a palavra w ocorre.

- Para cada sentença s, é computado o *TF-ISF* médio, denotado por *Avg-TF-ISF(s)*, calculado pela média aritmética do *TF-ISF(w,s)* sobre todas as palavras w da sentença, que é:

$$Avg\text{-}TF\text{-}ISF(s) = \sum_{i=1}^{W(s)} TF\text{-}ISF(i,s) / W(s)$$

sendo,

W(s): número de palavras na sentença s.

- Uma vez que o valor do *Avg-TF-ISF(s)* é computado para cada sentença s, o passo final é selecionar as sentenças mais relevantes (as que apresentam maior valor de *Avg-TF-ISF(s)*). Isto é feito da seguinte maneira:

- O sistema seleciona a sentença com o maior valor de *Avg-TF-ISF(s)*, chamada de *Max-Avg-TF-ISF*. O usuário especifica um limiar percentual deste valor, denotado por *percentage-threshold*. Este valor é utilizado para computar o *Threshold-TF-ISF*, da seguinte maneira:

  *Threshold-TF-ISF = percentage-threshold * Max-Avg-TF-ISF*

- A saída do algoritmo de sumarização consiste de todas as sentenças nas quais o valor de *Avg-TF-ISF(s)* é maior ou igual a *Threshold-TF-ISF*.

A tela do sistema pode ser vista a seguir:

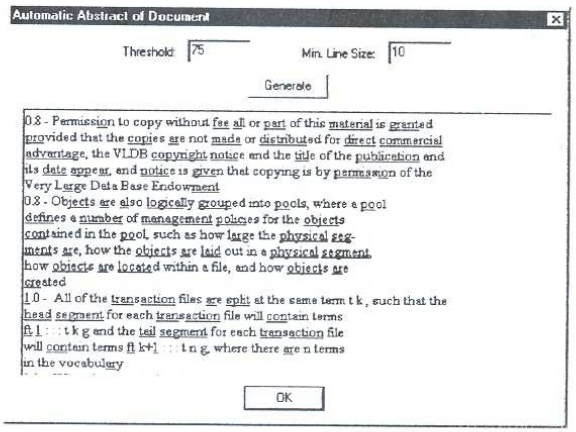


Figura 11 - Tela do sistema de sumarização de textos *TF-ISF*

Note que, da forma descrita acima, o sistema pode selecionar um número bastante variável de sentenças, dependendo das diferenças entre os valores *TF-ISF* para cada sentença. Porém, pode-se facilmente fazer o sistema selecionar um número máximo de sentenças, uma vez que as sentenças estejam em ordem decrescente de *Avg-TF-ISF*, selecionando-se as n primeiras sentenças, aonde n é um parâmetro definindo o tamanho do sumário desejado.

Sistema 2: Sumarização de textos baseada na importância relativa dos tópicos

O sistema funciona da seguinte maneira [Larocca 00b]:

Fase 1: Pré-processamento

O documento é convertido para a representação *bag-of-words*, e todos os pré-processamentos tradicionais são aplicados: case-folding, remoção de *stopwords* e *stemming* [Witten 94]. Para remoção de *stopwords*, utilizou-se uma lista de 524 palavras, obtidas do código fonte da biblioteca *BOW*, desenvolvida pela Universidade de *Carnegie Mellon*. Para efetuar o processo de *stemming*, utilizou-se o conhecido algoritmo de Porter [Porter 80]. Para textos em outras línguas ou multilíngues, optou-se pela representação de n-grams [Cavnar 94], mais especificamente, *quad-grams*, devido à aparente ausência de algoritmos de *stemming* para línguas como o português. Para representar o valor associado a cada dimensão do vetor, utilizou-se a métrica *TF-ISF*(*term frequency – inverse sentence frequency*) [Larocca 00a].

Fase 2: Segmentação em Tópicos

Para segmentar o documento em tópicos, utilizou-se uma versão modificada do algoritmo *TextTiling*. O algoritmo *TextTiling* [Hearst 93] propõe um método para particionar documentos em unidades coerentes. Na versão modificada, a métrica *TF-IBF*(*term frequency – inverse block frequency*) é utilizada para comparação de blocos adjacentes. Além disso, a versão desenvolvida suporta a representação de n-grams, que não necessita dos pré-processamentos tradicionais e é portanto independente de língua, permitindo a utilização do algoritmo em textos escritos em português, por exemplo. O algoritmo funciona em 2 passos:

No primeiro passo, todos os pares de blocos adjacentes no texto (geralmente de 3 a 5 sentenças) são segmentados, e uma representação vetorial contendo pesos *TF-IBF* é produzida de forma similar ao cálculo do *TF-ISF*, substituindo-se o conceito de sentença por bloco. Os vetores adjacentes são então comparados e associados a um valor de similaridade. A similaridade entre os blocos é calculada através da conhecida similaridade dos cosenos, que calcula o ângulo formado entre os dois vetores correspondentes aos blocos. Na similaridade dos cosenos, o valor 1.00 indica total semelhança entre os blocos e valores próximos de 0.00 indicam nenhuma semelhança.

No segundo passo, os valores de similaridade são suavizados, utilizando-se um filtro de medianas, similar ao utilizado em algoritmos de processamento de imagens, com

janela de tamanho 3, para eliminar pequenos mínimos locais. O gráfico é então analisado em relação a picos e vales. Picos indicam coerência e vales indicam potenciais quebras entre segmentos coerentes do texto (tópicos). As divisões de segmentos são localizadas através das porções mais baixas do gráfico. Os valores reais da similaridade não são levados em conta, e sim as diferenças relativas.

Um parâmetro ajustável é o tamanho dos blocos utilizados para comparação, que pode ser heurísticamente determinado utilizando-se o tamanho médio dos parágrafos do texto. Outro parâmetro determina exatamente quando uma queda de similaridade no gráfico é significativa o bastante para ser considerada uma quebra de tópico.

Utilizou-se um algoritmo simples que detecta uma mudança de tópico toda vez que uma queda relativa na similaridade é maior que um limiar pré-estabelecido. Valores típicos para o limiar são 0,3 ou 0,2.

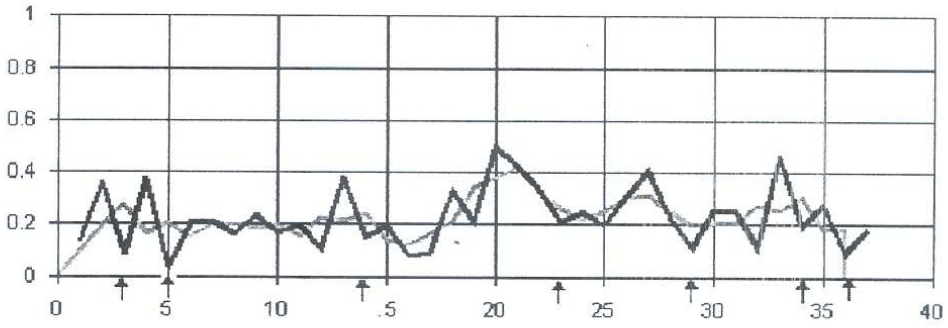


Figura 12 - Aplicação do algoritmo TextTiling em um documento

Na figura acima pode-se ver o resultado da aplicação do algoritmo *TextTiling* em um documento em português contendo aproximadamente 110 linhas, com um bloco de 3 linhas e utilizando *quad-grams*. A linha em preto representa os valores da similaridade *TF-IBF* entre blocos. A linha em cinza representa os mesmos valores após a aplicação do filtro de medianas com janela de tamanho 3. As setas representam vales descobertos pelo algoritmo descrito utilizando diretamente os valores da similaridade *TF-IBF*, que empiricamente apresentaram os melhores resultados.

As fases 3 e 4, descritas a seguir, utilizaram técnicas de autoria própria, sendo as maiores contribuições do autor neste sistema.

Biblioteca Central - PUC/PR

Fase 3: Determinação da força dos tópicos

Após a segmentação dos tópicos, o sistema deve obter um fator que represente a importância do tópico no documento, de forma a estabelecer tópicos prioritários e secundários. Tal fato é de extrema importância para evitar que sumários pouco representativos sejam gerados. Supondo que desejássemos sumarizar um artigo técnico, tópicos como "*Related Work*" e "*References*" claramente não devem figurar no sumário, e portanto, devem apresentar um coeficiente de importância baixo.

A solução utilizada para determinação da importância dos tópicos foi calcular a relevância total das frases existentes em cada tópico t contra a relevância total de todos os tópicos, ou seja:

$$\text{RelTopImp(t)} = \frac{Top\,\text{Im}\,p(t)}{\sum_{i=1}^{|T|} Top\,\text{Im}\,p(i)}$$

onde |T| é o número de tópicos encontrados na fase 2

O resultado da fórmula RelTopImp(t) fornece um valor entre 0 e 1 que corresponde à contribuição do tópico no documento.

Para calcular a relevância total das frases existentes no tópico, *TotImp(t)*, necessita-se estabelecer antes uma métrica para cálculo da relevância de uma frase. Foi utilizada a métrica *Avg-TF-ISF*, introduzida em [Larocca 00a], e descrita acima no Sistema 1.

A determinação da relevância total das frases existentes no tópico é dada pela soma dos valores de *Avg-TF-ISF* para todas as sentenças existentes no tópico, ou seja:

$$Top\,\text{Im}\,p(t) = \sum_{x \in T^A} Avg\text{–}TF - ISF(x)$$

onde T^A é o conjunto de sentenças do tópico A.

Fase 4: Seleção de sentenças relevantes nos tópicos

Esta etapa seleciona as n sentenças mais relevantes de cada tópico, sendo que n é obtido através da força associada ao tópico.

Dentro de cada tópico, as sentenças são selecionadas de acordo com a sua similaridade *TF-ISF* em relação ao centróide do tópico. O centróide do tópico representa o vetor médio de todas as sentenças contidas no tópico. As sentenças são então selecionadas em ordem decrescente de valores similaridade *TF-ISF* com relação ao centróide, fazendo com que as frases mais representativas para um determinado tópico sejam selecionadas antes.

Uma visão geral do processamento deste sistema, executando seqüencialmente as quatro fases acima, é mostrada na Figura 13.

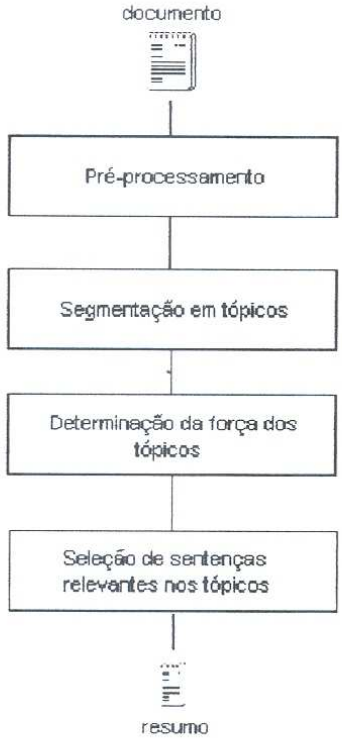


Figura 13 - Arquitetura do sistema de sumarização baseado na importância relativa dos tópicos

Sistema 3: Sumarização Treinável de Documentos Utilizando Características Posicionais, Temáticas e de Coesão

O sistema funciona da seguinte maneira [Larocca 00c]:

- Inicialmente são produzidos sumários extrativos "ideais" com 10% das sentenças do texto a partir do sumário fornecido pelo autor, em uma técnica similar a [Mani98a]. Esta técnica permite a criação de sumários ideais de forma automática, sem a necessidade de um juiz humano, e baseia-se em uma consulta do sumário provido pelo autor contra sentenças contidas no documento.
- São extraídas 7 características dos documentos da base *CMP-LG*, todas de natureza contínua. As características utilizadas para detecção da presença de sentenças no sumário podem ser separadas em 3 grupos: posicionais, temáticas e baseadas em coesão.

  As características posicionais e relativas ao tamanho são:

  - **Posição da Sentença**: posição da sentença no texto, similar a [Nevill-Manning 99]. Indica a posição da palavra no documento. O valor é normalizado na escala de 0 a 1.
  - **Tamanho da Sentença**: tamanho da sentença, normalizado pelo tamanho da maior sentença do documento.

As características temáticas são:

- **TFISF Médio**: *TF-ISF* médio das palavras de cada sentença. Os pesos *TF-ISF* [Larocca 00a] representam o valor de cada palavra na representação vetorial de documentos, combinando as estatísticas *TF(i, d)* e *SF(i)*, e fornecem uma indicação da importância das palavras no documento.
- **Semelhança com o Título**: resultado da aplicação do título do documento como consulta a todas as sentenças. Inicialmente tanto o título como cada sentença do documento são convertidos para representação vetorial, sendo que a similaridade entre os dois vetores é obtida através da similaridade dos cosenos [Salton 88].
- **Semelhança com Palavras Temáticas**: é processado de forma similar à característica "Semelhança com o Título", só que ao invés do título do documento, utiliza-se de 15 palavras-chaves do mesmo. As 15 palavras-chaves foram extraídas utilizando a *API* de programação do *software Extractor*, um conhecido programa de extração de palavras-chaves de documentos, proposto em [Turney 00].

As características baseadas em coesão são:

- **Conectividade da Sentença**: é processada da seguinte maneira:
  - Para cada sentença do documento, efetuar uma consulta utilizando a sentença selecionada contra todas as demais sentenças.
  - Somar os valores de similaridade.
  - Após processar os somatórios para todas as sentenças, normalizar o valor obtido pela maior soma.

  A idéia intuitiva desta métrica é que sentenças com valores próximos de 1.00 para esta característica são as mais coesas do documento. A idéia proposta é similar ao caminho *Global Bushy Path* proposto em [Mitra 97].

- **Semelhança com o Centróide**: é processada da seguinte maneira:
  - Calcular o centróide do documento (vetor médio de todas as sentenças do documento, incluindo título).
  - Aplicá-lo como consulta a todas as sentenças do documento, obtendo a similaridade entre cada sentença e o centróide.
  - Normalizar as similaridades pela maior obtida.

  A idéia intuitiva desta métrica é que sentenças com valores próximos de 1 para esta característica representam as idéias gerais do documento.

Em seguida as características são discretizadas em intervalos com largura constante. Atualmente o sistema utiliza somente 3 intervalos de valores possíveis para todas as características: alto, médio e baixo. Foram efetuados testes com métodos mais sofisticados de discretização, como a discretização baseada em entropia, fornecida pelo utilitário *Discretize* da biblioteca MLC++ [Koravi 96]. Entretanto, os resultados ficaram abaixo dos obtidos por uma discretização com largura constante.

Utiliza-se o classificador Naive-Bayes para predição das linhas que devem compor o sumário. Optou-se pelo classificador Naive-Bayes devido a sua simplicidade e o fato de

já ter sido utilizado em diversos trabalhos na área de sumarização de dados [Kupiec 95] [Teufel 99] com bons resultados.

As telas do sistema podem ser vistas nas figuras 14 e 15.

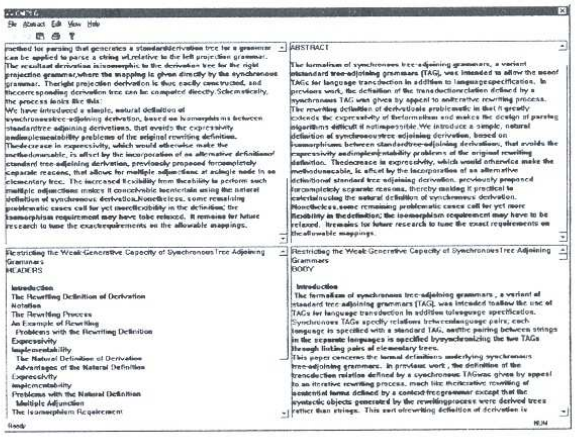


Figura 14 - Sistema Extrator de Características

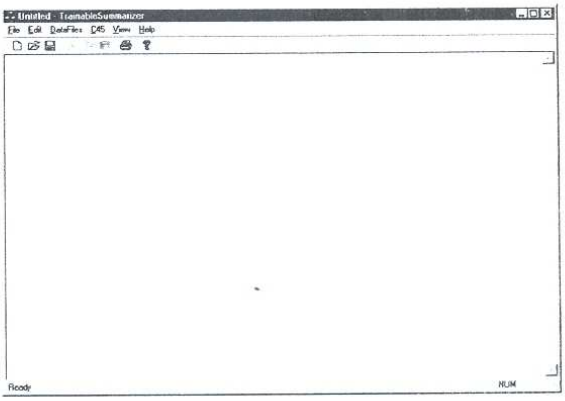


Figura 15 - Sistema Gerador de Arquivos para Classificação

Sistema 4: Sumarização Treinável de Documentos Utilizando a Estrutura Argumentativa do Texto

Neste sistema, o objetivo geral é desenvolver um sistema treinável para sumarização de notícia e obtenção aproximada da estrutura argumentativa do texto [Larocca 00d].

A obtenção aproximada da estrutura argumentativa do texto é efetuada da seguinte maneira:

- O texto é processado pelo algoritmo de *clustering* aglomerativo, como proposto por [Yaari 97].
- Cada sentença presente na saída do *clustering* aglomerativo é classificada como relevante (captura as idéias principais do documento) ou de fundo (*background*) (contém informação adicional, não essencial).
- A detecção de sentenças de fundo é baseada nos seguintes critérios:
    - A sentença contém pouco ou nenhum conceito principal do texto.
    - A sentença apresenta baixa profundidade na árvore do clustering aglomerativo, pois apresenta pouca coesão com as demais sentenças. A profundidade é considerada baixa se for menor ou igual a metade da profundidade geral da árvore.
    - A sentença contém anáforas ou referências externas.
    - Existem marcadores de discurso no início da sentença.
- Após detectar as sentenças de fundo, as demais sentenças são consideradas como relevantes.

O sistema treinável de sumarização funciona da seguinte maneira:

- Inicialmente são produzidos sumários extrativos com 10% e 20% das sentenças do texto a partir do sumário fornecido pelo autor, em uma técnica similar a [Mani98a].
- São extraídas 7 características dos documentos da base de notícias da ZIFF-Davis, algumas baseadas na estrutura argumentativa aproximada do texto. As características são:
    - Indicador de conceitos principais: Indica se a sentença captura ou não os conceitos principais do documento. Partindo do pressuposto que a maioria das palavras relevantes são substantivos, a identificação é realizada da seguinte maneira:

- Todos os substantivos da sentença são selecionados, utilizando um software de *part-of-speech* [Brill 92].
- Substantivos repetidos são removidos.
- Para cada substantivo, é calculado o número de sentenças na qual o termo aparece pelo menos uma vez.
- Os 15 termos com maior número de ocorrência nas sentenças são selecionados.

- Ocorrência de nomes próprios: a ocorrência de nomes próprios de pessoas e locais representam dicas de relevância positiva da sentença no sumário, especialmente para textos de notícias. Os nomes próprios foram identificados por um software de *part-of-speech* [Brill 92].
- Ocorrência de anáforas: a ocorrência de anáforas geralmente indica a presença de informação adicional, não essencial para o texto. A detecção de anáforas foi feita de forma similar a [Strzalkowski 98]: verifica-se pela existência de certas palavras no início da sentença (primeiras 6 palavras).
- Ocorrência de marcadores de discurso no início da sentença: Alguns marcadores de discurso, como "*because*", "*furthermore*" e "*additionaly*" são considerados indicadores de presença de informação adicional, não essencial ao sumário.
- Conectividade das sentenças: Parte da evidência que sentenças não essenciais ao sumário apresentam baixa coesão. É processada de forma similar a descrita no sistema 3.
- Profundidade da sentença na árvore: Indica a profundidade da sentença na árvore produzida pelo algoritmo de *Clustering* Aglomerativo.
- Posição na árvore: considera o caminho da raiz da árvore do *Clustering* Aglomerativo até a sentença selecionada, considerando somente profundidades de até 4 níveis. Como a referida árvore é binária, os valores possíveis são:

"*left*" , "*right*" e "*none*" (a sentença está em uma profundidade menor do que 4).

- Após extrair as características, o sistema foi treinado em dois classificadores: C4.5 [Quinlan93] e Naive-Bayes.

4 Experimentos e Análise de Resultados

Este capítulo está organizado da seguinte maneira: em "Formato e métricas de avaliação", os procedimentos e *softwares* utilizados na validação dos sumários produzidos automaticamente são detalhados. A seção "Sistemas de sumarização utilizados" descreve os sistemas de sumarização que participaram da avaliação, bem como os parâmetros de configuração dos mesmos. A seção "Descrição das bases de documentos utilizadas" descreve as 3 bases de documentos selecionadas e finalmente a seção "Resultados dos experimentos" contém os resultados obtidos nos 8 experimentos realizados.

4.1 Formato e métricas de Avaliação

Para análise de resultados, utilizou-se o modelo do "sumário ideal", uma tarefa intrínseca, que avalia a qualidade do sumário baseada na análise do mesmo. A justificativa para tal procedimento é que mesmo que o "sumário ideal" seja difícil de ser estabelecido (pois pode existir um grande número de sumários de boa qualidade gerados a partir de um documento), existem estudos que afirmam que embora os juizes não concordem em todas as sentenças a serem incluídas no sumário, eles concordam nas sentenças mais importantes a serem incluídas. A vantagem da utilização de uma tarefa intrínseca com relação a tarefas extrínsecas é que avaliações extrínsecas podem ter seus resultados influenciados por fatores externos, além de serem mais difíceis de serem realizadas. Desta forma, pode-se afirmar que a avaliação conduzida é rigorosa, no sentido que "pune" sumários que contenham sentenças relevantes mas que não façam parte do sumário ideal. Entretanto, como todos os métodos foram avaliados com o mesmo rigor, pode-se afirmar que os resultados comparativos entre os sistemas são válidos.

Os documentos analisados foram escritos na língua inglesa, e as taxas de compressão de 10% e 20% das sentenças do texto foram utilizadas. A limitação da língua é imposta pela maioria dos sistemas, que envolvem pré-processamentos disponíveis somente para a língua inglesa. Com relação às taxas de compressão, adotou-se as duas mais comumente utilizadas em experimentos na literatura.

Os "sumários ideais" foram gerados de duas formas:

- Manual

Neste caso, um juiz humano seleciona sentenças de alta relevância. O número de sentenças selecionadas depende da taxa de compressão utilizada. Nestes experimentos, foram utilizadas as taxas de 10% e 20%, o que significa que o juiz seleciona 10% ou 20% do número total de sentenças do documento. Para construção dos sumários manuais, contratou-se uma professora de inglês, que embora não tenha o inglês como idioma nativo, é graduada em Lingüística e leciona a diversos anos.

- Automático

Sistemas de sumarização dificilmente são avaliados em bases de treinamento de tamanho consideravelmente grande (acima de 100 documentos), visto que tal fato envolve grande trabalho manual para a criação de sumários extrativos. [Mani 98a] propôs uma técnica inovadora para a obtenção de sumários extrativos: utilizar o sumário provido pelo autor do documento (não-extrativo) como consulta a cada uma das sentenças do documentos. Neste contexto, sentenças que apresentarem maior relevância a consulta são ordenadas e sumários com tamanho fixo (tipicamente 10% ou 20% das sentenças) são obtidos. Desta forma, qualquer base de dados textual que apresente sumários fornecidos pelo autor pode ser utilizada no treinamento do sistema. Esta técnica tem a vantagem de ser muito mais simples de ser obtida, pois não envolve interação humana, mas os sumários gerados não são garantidos de terem a mesma coerência do sumário provido pelo autor. Um dos experimentos realizados visa comparar os sumários manuais com os sumários automáticos, tentando avaliar o grau de similaridade entre os mesmos.

As métricas de avaliação utilizadas foram *precision* e *recall*, utilizadas na grande maioria dos trabalhos da área.

As fórmulas para cálculo de *precision* e *recall* foram definidas na seção 2.3.

Deve-se notar que em casos aonde o sistema seleciona o mesmo número de sentenças que o "sumário ideal", os valores de *precision* e *recall* são idênticos, e os sistemas utilizam simplesmente o termo taxa de acerto. Nesta avaliação, alguns sistemas

apresentavam *precision* e *recall* idênticos mas outros não, de forma que preferiu-se adotar as duas métricas.

Além disso, para análise estatística, o desvio padrão da amostra para os valores de *precision* e *recall* foi calculado, utilizando-se a seguinte fórmula:

$$\sigma = \sqrt{\frac{\sum_{i=1}^{n}(P_i - \bar{x})^2}{\frac{n-1}{n}}}$$

sendo,

n: número de documentos

P_i: valor na amostra da métrica analisada (*precision* ou *recall*)

$\bar{x}$: média na amostra da métrica analisada (*precision* ou *recall*)

Para realização dos experimentos, desenvolveram-se 4 *softwares*, como indicado na figura 16 e descritos a seguir.

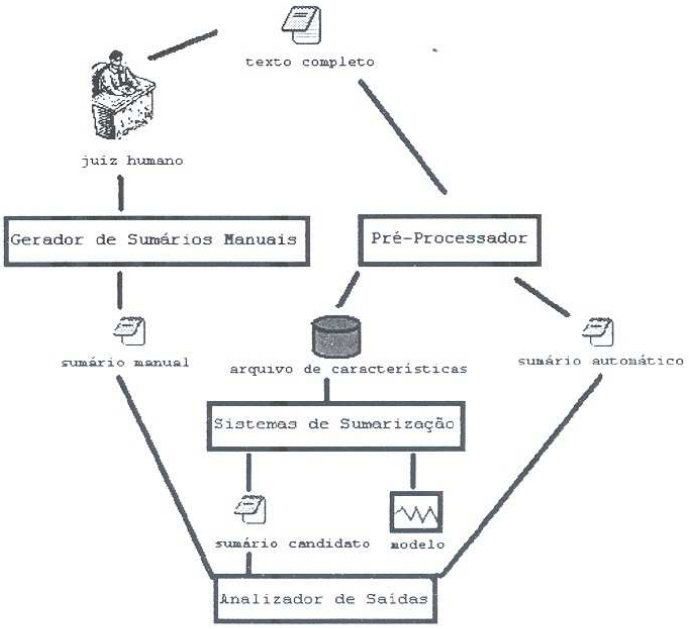


Figura 16 – Softwares utilizados nos experimentos

1) Pré-processador (*DocumentBase Preprocessor*)

Tem como objetivo pré-processar os documentos de entrada (fornecidos no formato *XML*) e extrair diversas características, contidas nos seguintes arquivos:

- Orig: contém o texto original.
- Topics: contém os termos de indexação do documento, quando disponível.
- Dat: arquivo de características utilizado pelos sistemas de sumarização baseados em treinamento.
- Abs: texto do sumário provido pelo autor.
- Qabs: texto do sumário extrativo automático obtido a partir do sumário do autor.
- Vabs: sumário extrativo automático em formato binário, para comparação posterior.

2) Sistemas de Sumarização (*Summarization Algorithms*)

Tem como objetivo executar as diversas técnicas de sumarização nos documentos especificados. Efetua os processos de treinamento e de validação.

No processo de treinamento, são oferecidos os algoritmos C4.5 e Naïve-Bayes. Os algoritmos produzem como saída um modelo. No caso do C4.5, uma árvore de decisão é gerada; no caso do Naïve-Bayes, um arquivo contendo as probabilidades de cada atributo.

No processo de validação, as técnicas disponíveis são utilizadas na validação de documentos. Para cada documento de entrada, é produzido um arquivo contendo o sumário extrativo no formato binário. Os algoritmos oferecidos são:

- C4.5 (sob um conjunto de características selecionadas)
- Naïve-Bayes (sob um conjunto de características selecionadas)
- Sumários produzidos com as primeiras sentenças do texto
- Sumários produzidos com sentenças aleatórias do texto
- Word Summarizer

- TF-ISF
- TF-ISF baseado em tópicos

3) Analisador de Saídas (*Output Analyser*)

Analisa e gera arquivos binários de sumários. Apresenta os seguintes recursos:

- Visualização do sumário produzido para um documento, a partir dos arquivos orig (texto original) e vabs (sumário em formato binário).
- Comparação dos resultados de sumários produzidos pelas técnicas com sumários ideais, fornecendo os valores de *precision*, *recall*, *mean precision*, *mean recall* e seus respectivos desvios padrões.
- Permite criar arquivos binários de sumários manualmente. Este recurso foi utilizado para entrada dos dados de sumários ideais proveniente do juiz humano, para posterior comparação com as técnicas de sumarização automática.
- Efetua o processo de *clustering* aglomerativo de sumários, descrito mais adiante nos experimentos, produzindo uma árvore binária como saída.

4) Gerador de sumários manuais (*Manual Summaries*)

Analisa o texto de um documento (no formato *XML*), efetua os pré-processamentos mínimos para separação de linhas, informa quantas linhas devem ser selecionadas para sumários de 10% e 20% das sentenças do texto e indexa as linhas, para facilitar a entrada posterior no sistema 3.

4.2 Sistemas de Sumarização Utilizados

Os 4 sistemas de sumarização propostos na seção 3.2 foram utilizados nos experimentos. Em dois destes sistemas (o sistema 3 e o 4), experimentos foram feitos utilizando dois classificadores diferentes (C4.5 e Naive Bayes). Foi também realizado um experimento com um sistema híbrido 3/4, novamente com dois classificadores. Portanto, ao todo 8 variantes dos sistemas de sumarização propostos neste trabalho foram avaliados.

Estas 8 variantes foram comparadas com 3 outros sistemas de sumarização. A seguir é descrito resumidamente cada um dos 11 sistemas utilizados nos experimentos.

- Sistema 1 – *TF-ISF*

Seleciona sentenças baseado nos valores da métrica *TF-ISF*. Este algoritmo apresenta uma heurística fixa e portanto não utiliza os dados de treinamento.

- Sistema 2 – *Topical TF-ISF*

Seleciona sentenças particionando o documento em tópicos, fornecendo valores de relevâncias aos mesmos e extraindo sentenças de cada tópico de forma proporcional a relevância. Este algoritmo apresenta uma heurística fixa e portanto não utiliza os dados de treinamento.

- Sistema 3 com classificador C4.5

Utiliza as características do sistema 3, combinado com o classificador C4.5. Para evitar o problema de desbalanceamento de classes (visto que somente dez ou vinte porcento das sentenças são nomeadas com a classe positiva), utilizou-se somente 30% dos exemplos negativos disponíveis. As características utilizadas foram: Posição da Sentença, Tamanho da Sentença, TFISF Médio, Semelhança com o Título, Semelhança com Palavras Temáticas, Conectividade da Sentença e Semelhança com o Centróide.

- Sistema 3 com classificador Naïve Bayes

Utiliza as características do sistema 3, combinado com o classificador Naïve Bayes. As características utilizadas foram: Posição da Sentença, Tamanho da Sentença, TFISF Médio, Semelhança com o Título, Semelhança com Palavras Temáticas, Conectividade da Sentença e Semelhança com o Centróide.

- Sistema 4 com classificador C4.5

Utiliza as características do sistema 4, combinado com o classificador C4.5. Para evitar o problema de desbalanceamento de classes (visto que somente dez ou vinte porcento das sentenças são nomeadas com a classe positiva), utilizou-se somente 30% dos exemplos negativos disponíveis. As características utilizadas foram: Indicador de conceitos

principais, Ocorrência de nomes próprios, Ocorrência de anáforas, Ocorrência de marcadores de discurso no início da sentença, Conectividade das sentenças, Profundidade da sentença na árvore e Posição na árvore (4 atributos, representando os 4 níveis de profundidade utilizados).

- Sistema 4 com classificador Naïve Bayes

Utiliza as características do sistema 4, combinado com o classificador Naïve Bayes. As características utilizadas foram: Indicador de conceitos principais, Ocorrência de nomes próprios, Ocorrência de anáforas, Ocorrência de marcadores de discurso no início da sentença, Conectividade das sentenças, Profundidade da sentença na árvore e Posição na árvore (4 atributos, representando os 4 níveis de profundidade utilizados).

- Sistema 3/4 com classificador C4.5

Utiliza as características do sistema 3 somadas com as características do sistema 4, combinado com o classificador C4.5. Para evitar o problema de desbalanceamento de classes (visto que somente dez ou vinte porcento das sentenças são nomeadas com a classe positiva), utilizou-se somente 30% dos exemplos negativos disponíveis. As características utilizadas foram: Posição da Sentença, Tamanho da Sentença, TFISF Médio, Semelhança com o Título, Semelhança com Palavras Temáticas, Conectividade da Sentença, Semelhança com o Centróide, Indicador de conceitos principais, Ocorrência de nomes próprios, Ocorrência de anáforas, Ocorrência de marcadores de discurso no início da sentença, Profundidade da sentença na árvore e Posição na árvore (4 atributos, representando os 4 níveis de profundidade utilizados).

- Sistema 3/4 com classificador Naïve Bayes

Utiliza as características do sistema 3 somadas com as características do sistema 4, combinado com o classificador Naïve Bayes. As características utilizadas foram: Posição da Sentença, Tamanho da Sentença, TFISF Médio, Semelhança com o Título, Semelhança com Palavras Temáticas, Conectividade da Sentença, Semelhança com o Centróide, Indicador de conceitos principais, Ocorrência de nomes próprios, Ocorrência de anáforas,

Ocorrência de marcadores de discurso no início da sentença, Profundidade da sentença na árvore e Posição na árvore (4 atributos, representando os 4 níveis de profundidade utilizados).

- FirstSentences

Seleciona as n primeiras sentenças do documento, sendo n estimado a partir da taxa de compressão desejada. A motivação para este sistema é o fato constatado em [Brandow 94], no qual sumários produzidos com as n-primeiras sentenças do texto obtiveram resultados similares a métodos mais elaborados.

- Random Sentences

Seleciona n sentenças aleatórias do documento, sendo n estimado a partir da taxa de compressão desejada. A motivação para este sistema é fornecer um limite inferior para a performance de qualquer sistema de sumarização automática, pois todos devem apresentar resultados acima do aleatório.

- Word Summarizer

O Microsoft Word Summarizer é o sumarizador de textos integrante do processador de textos Microsoft Word, e foi utilizado para comparação de resultados em diversos trabalhos realizados na literatura [Barzilay 97] [Marcu 99] [Chuang 00]. Para os experimentos, utilizou-se a versão contida no pacote Microsoft Office 97. O sistema utiliza técnicas não documentadas para sumarizar um documento de forma extrativa, com um grau de compressão especificado pelo usuário. Entretanto, ele apresenta algumas particularidades dos demais sistemas, a saber:

1) O grau de compressão especificado refere-se ao número de caracteres presentes no sumário, e não o número de linhas.

2) Algumas sentenças são modificadas pelo sumarizador durante a execução, visando diminuir o número de caracteres sem modificar a compreensão do texto. Por exemplo, se a sentença abaixo for selecionada para inclusão no sumário, o pronome de tratamento "Mr." no início poderia ser removido, gerando sentenças que não são exatamente iguais as do documento original.

Mr. Darden's case is part of a broad federal crackdown on * computer hackers that has led to more than 30 raids in cities across the country.

Tendo em vista a particularidade 1, os sumários gerados pelo Microsoft Word Summarizer podem conter mais ou menos sentenças que as selecionadas pelos demais algoritmos. Neste caso, os valores de *precision* e *recall* serão diferentes. Tendo em vista a particularidade 2, como algumas sentenças são modificadas durante a sumarização, nem sempre é possível determinar a correspondência exata com sentenças do documento original. Neste caso, estas sentenças são ignoradas. Entretanto, elas não correspondem a um número significativo do total de sentenças selecionadas, evitando desta forma comprometer os resultados.

4.3 Descrição das bases de documentos utilizadas

Para a realização dos experimentos, foram necessárias 3 bases de documentos, obtidas dos textos da editora Ziff-Davis, na base TIPSTER [Harman94]. A base consiste de textos de revistas sobre computadores, hardware, software, etc. Dentre os textos disponíveis, 33.658 contém sumários providos pelo autor. Os textos apresentam tamanhos variados: de 2 Kbytes a mais de 64 Kbytes. As 3 bases de documentos produzidas foram:

- Base de treinamento

  Contém os documentos utilizados para treinamento. A base é composta de cerca de 100 documentos selecionados aleatoriamente da base TIPSTER, com a seguinte distribuição:

  - 25 documentos de 11 Kbytes
  - 25 documentos de 12 Kbytes
  - 25 documentos de 16 Kbytes
  - 25 documentos de 31 Kbytes

O tamanho médio dos documentos é de 129,5 sentenças, produzindo um total de 12.950 sentenças, sendo que, para sumários com taxa de compressão de 10%, 1.251 sentenças eram exemplos positivos de inclusão no sumário e 11.699 eram exemplos

negativos. Para sumários com taxa de compressão de 20%, 2.548 sentenças eram exemplos positivos de inclusão no sumário e 10.402 eram exemplos negativos.

Esta base foi utilizada para treinamento dos sistemas nos experimentos descritos nas seções 4.4.1, 4.4.2, 4.4.3 e 4.4.4. Os sumários extrativos dos documentos desta base foram gerados conforme a técnica proposta por [Mani 98a], descrita na seção 2.5.

- Base de validação com sumários automáticos

  Contém os documentos utilizados na validação dos sistemas utilizando sumários extrativos automáticos. Estes sumários foram gerados conforme a técnica proposta por [Mani 98a], descrita na seção 2.5. A base é composta de cerca de 100 documentos selecionados aleatoriamente da base TIPSTER, com a seguinte distribuição:

  - 25 documentos de 10 Kbytes
  - 25 documentos de 13 Kbytes
  - 25 documentos de 15 Kbytes
  - 25 documentos de 28 Kbytes

  O tamanho médio dos documentos é de 118,6 sentenças, produzindo um total de 11.860 sentenças, sendo que, para sumários com taxa de compressão de 10%, 1.142 sentenças eram exemplos positivos de inclusão no sumário e 10.718 eram exemplos negativos. Para sumários com taxa de compressão de 20%, 2.333 sentenças eram exemplos positivos de inclusão no sumário e 9.527 eram exemplos negativos.

  Esta base foi utilizada para validação dos resultados dos sistemas nos experimentos descritos nas seções 4.4.1, 4.4.2 e 4.4.6.

- Base de validação com sumários manuais

  Contém os documentos utilizados na validação dos sistemas utilizando sumários extrativos manuais, elaborados por um juiz humano contratado especificamente para este trabalho. É composta de cerca de 30 documentos selecionados aleatoriamente da base TIPSTER. Todos os documentos apresentam tamanho

aproximando de 10 Kbytes, que fornece textos com aproximadamente 3 folhas. Este tamanho foi escolhido devido a restrição de que um juiz humano deveria selecionar sentenças para cada documento nesta base, de forma que textos muitos longos fariam a análise humana inviável. O processo de construção de sumários manuais para os 30 documentos, contendo 10% e 20% das sentenças do texto consumiu cerca de 15 horas.

O tamanho médio dos documentos é de 65,67 sentenças, produzindo um total de 1.970 sentenças, sendo que, para sumários com taxa de compressão de 10%, 183 sentenças eram exemplos positivos de inclusão no sumário e 1.787 eram exemplos negativos. Para sumários com taxa de compressão de 20%, 382 sentenças eram exemplos positivos de inclusão no sumário e 1.588 eram exemplos negativos.

Esta base foi utilizada para validação dos resultados dos sistemas nos experimentos descritos nas seções 4.4.3, 4.4.4 e 4.4.5. Adicionalmente, nos experimentos 4.4.7 e 4.4.8 a base foi particionada em 2 conjuntos: treinamento e validação. Neste caso, a base de treinamento apresentava 20 documentos e um tamanho médio de 65,75 sentenças, produzindo um total de 1.315 sentenças, sendo que, para sumários com taxa de compressão de 10%, 123 sentenças eram exemplos positivos de inclusão no sumário e 1.192 eram exemplos negativos. Para sumários com taxa de compressão de 20%, 255 sentenças eram exemplos positivos de inclusão no sumário e 1.060 eram exemplos negativos. A base de validação apresentava 10 documentos e um tamanho médio de 65,5 sentenças, produzindo um total de 655 sentenças, sendo que, para sumários com taxa de compressão de 10%, 60 sentenças eram exemplos positivos de inclusão no sumário e 595 eram exemplos negativos. Para sumários com taxa de compressão de 20%, 127 sentenças eram exemplos positivos de inclusão no sumário e 528 eram exemplos negativos.

4.4 Resultados dos Experimentos

Foram realizados 8 experimentos, sendo que seis experimentos visam determinar a taxa de acerto dos sistemas contra sumários ideais produzidos de forma automática e manual com taxas de compressão de 10% e 20%, um experimento visa comparar sumários

ideais automáticos e manuais, e um experimento consiste em agrupar os sumários produzidos através do uso de *clustering* aglomerativo.

4.4.1 Análise da taxa de acerto dos 11 sistemas contra sumários ideais automáticos com 10% das sentenças do documento

Nesta análise, todos os sistemas foram avaliados na base de validação com sumários automáticos com taxa de compressão de 10%. Para sistemas que necessitam de treinamento, a base de treinamento foi utilizada.

Abaixo segue os resultados obtidos no experimento:

Sistema	Precision Médio	Desvio padrão do Precision	Recall Médio	Desvio padrão do Recall
Sistema 1(TF-ISF)	0.0512617	0.0100513	0.0512617	0.0100513
Sistema 2 (Topical TF-ISF)	0.232195	0.0120885	0.232195	0.0120885
Sistema 3 C4.5	0.363993	0.018968	0.336457	0.0181738
Sistema 3 Naïve-Bayes	0.385551	0.0197567	0.385551	0.0197567
Sistema ¾ C4.5	0.223642	0.0148359	0.223642	0.0148359
Sistema ¾ Naïve-Bayes	0.404757	0.0199062	0.404757	0.0199062
Sistema 4 C4.5	0.219257	0.014398	0.219257	0.014398
Sistema 4 Naïve-Bayes	**0.405007**	**0.0193252**	**0.405007**	**0.0193252**
Primeiras sentenças	0.239475	0.0160156	0.239475	0.0160156
Sentenças aleatórias	0.0891089	0.0094255	0.0891089	0.0094255
Word Summarizer	0.261302	0.0120983	0.344455	0.0157514

Tabela 3 - Taxa de acerto dos 11 sistemas contra sumários ideais automáticos com 10% das sentenças do documento

O sistema com maior valor médio de *precision* e *recall* foi o sistema 4 com o classificador Naïve-Bayes. Algumas análises que podem ser retiradas dos resultados são:

- O sistema 1 apresentou uma performance comparável à seleção de sentenças aleatórias do texto. Mesmo considerando-se que este sistema utiliza heurística fixa e que este tipo de avaliação (sumários ideais) é extremamente restrita, os resultados foram muito abaixo do esperado. Para tentar descobrir se a idéia básica de selecionar sentenças com alto valor de *TF-ISF* não produz bons resultados para sumarização, desenvolveu-se um novo sistema de sumarização. O novo sistema também é baseado no uso do *TF-ISF*, mas o critério de decisão

para selecionar uma sentença no sumário não é fixo, sendo induzido de uma base de treinamento através do classificador C4.5. Neste caso, a árvore de decisão encontrada pelo sistema, considerando a mesma base de treinamento que os demais experimentos, foi:

```
Decision Tree:
MeanTFISFCategoric = high: true (10.0/2.0)
MeanTFISFCategoric = medium: false (1530.0/507.0)
MeanTFISFCategoric = low: false (1899.0/397.0)
```

A árvore de decisão encontrada confirma a heurística fixa, no sentido que associa valores altos de *TF-ISF* a sentenças que devem ser selecionadas no sumário. Tal regra também está de acordo com a encontrada em [Mani 98a], na qual a métrica foi utilizada em conjunto com outras características.

Entretanto, aplicando-se o sistema gerado na base de validação com sumários automáticos, os resultados obtidos foram:

Sistema	Precision Médio	Desvio padrão do Precision	Recall Médio	Desvio padrão do Recall
Sistema 5: TF-ISF C4.5	0.262963	0.0429098	0.0321794	0.0060920

Tabela 4 -Resultados do sistema 5 contra sumários ideais automáticos com 10% das sentenças do documento

O sistema apresentou um valor de *precision* médio razoável, mas um péssimo *recall* médio. Tal fato se deve ao sistema ter selecionado muito menos sentenças que as necessárias para a taxa de compressão especificada. Ou seja, haviam poucas sentenças no documento com o valor "*high*" para a métrica *TF-ISF*, de forma que pode-se concluir que esta heurística, se utilizada isoladamente, não é o suficiente para realização da tarefa desejada. Este novo sistema será também utilizado nos experimentos seguintes.

- O sistema 2 apresentou resultados próximos da seleção das primeiras sentenças do documento.
- No caso do sistema 3, o classificador Naive-Bayes produziu resultados ligeiramente melhores que o C4.5. No caso do sistema 4 e o híbrido 3/4, o algoritmo Naive-Bayes produziu resultados muito melhores que o C4.5. Por

fim, a união das características dos sistemas 3 e 4 não resultou em ganhos na taxa de acerto, sendo que o melhor resultado foi obtido pelo sistema 4 isoladamente.

4.4.2 Análise da taxa de acerto dos 12 sistemas contra sumários ideais automáticos com 20% das sentenças do documento

Nesta análise, todos os sistemas foram avaliados na base de validação com sumários automáticos com taxa de compressão de 20%. Para sistemas que necessitam de treinamento, a base de treinamento foi utilizada.

Abaixo segue os resultados obtidos no experimento:

Sistema	Precision Médio	Desvio padrão do Precision	Recall Médio	Desvio padrão do Recall
Sistema 1(TF-ISF)	0.111797	0.0076706	0.111797	0.0076706
Sistema 2 (Topical TF-ISF)	0.347815	0.0111973	0.347815	0.0111973
Sistema 3 C4.5	0.513498	0.0141437	0.510668	0.0140515
Sistema 3 Naïve-Bayes	**0.519822**	**0.0144226**	**0.519822**	**0.0144226**
Sistema 3/4 C4.5	0.346825	0.0101486	0.346256	0.0102561
Sistema ¾ Naïve-Bayes	0.514266	0.0147384	0.514266	0.0147384
Sistema 4 C4.5	0.344739	0.00994729	0.344739	0.0099472
Sistema 4 Naïve-Bayes	0.492938	0.0139879	0.492938	0.0139879
Sistema 5 (TFISF C4.5)	0.392264	0.0224239	0.261961	0.0152669
Primeiras Sentenças	0.320348	0.0135512	0.320348	0.0135512
Sentenças Aleatórias	0.195716	0.0087902	0.195716	0.00879022
Word Summarizer	0.388028	0.0113877	0.436667	0.0129552

Tabela 5 - Taxa de acerto dos 12 sistemas contra sumários ideais automáticos com 20% das sentenças do documento

O sistema com maior valor médio de *precision* e *recall* foi o sistema 3 com o classificador de Naïve-Bayes. Algumas análises que podem ser retiradas dos resultados são:

- O melhor resultado obtido neste experimento foi consideravelmente melhor do que os obtidos no experimento anterior, que tratava de sumários com 10% das sentenças do documento. Além disso, todos os sistemas, inclusive sistemas muito simples como a seleção das primeiras sentenças do texto também obtiveram resultados consideravelmente melhores. Esta melhora geral nos resultados era esperada, já que o problema de produzir bons sumários com 20%

do texto é mais fácil do que o problema de produzir bons sumários com 10% do texto.

- O sistema 1 apresentou uma performance comparável à seleção de sentenças aleatórias do texto. Já a sua variação (o sistema 5) apresentou resultados melhores. Este fato pode ser confirmado inspecionando-se a árvore de decisão produzida na base de treinamento para sumários de 20% do documento:

```
Decision Tree:
MeanTFISFCategoric = high: true (4.0)
MeanTFISFCategoric = medium: true (968.0/460.5)
MeanTFISFCategoric = low: false (1727.0/709.5)
```

Neste caso, o classificador associou os valores "*high*" e "*medium*" como envidência positiva de sentenças no sumário, diferentemente do experimento anterior, aonde somente valores "*high*" eram considerados.

- O sistema 2 apresentou resultados ligeiramente superiores à seleção das primeiras sentenças do documento.
- Os sistema 3 apresentou resultados muito similares para os classificadores Naïve-Bayes e C4.5.
- Os sistemas 4 e a combinação dos sistemas 3 e 4 apresentaram resultados muito melhores utilizando o classificador Naïve-Bayes do que o C4.5. Por fim, a união das características dos sistemas 3 e 4 não resultou em ganhos na taxa de acerto, mas os resultados obtidos foram muito próximos do melhor resultado.

4.4.3 Análise da taxa de acerto dos 12 sistemas treinados com sumários automáticos contra sumários ideais manuais com 10% das sentenças do documento

Nesta análise, todos os sistemas foram avaliados na base de validação com sumários manuais com taxa de compressão de 10%. Para sistemas que necessitam de treinamento, a base de treinamento foi utilizada. Deve-se notar que a referida base contém valores de classe baseados em sumários ideais automáticos, e não

manuais, de forma que pode-se dizer que os sistemas foram treinados com sumários automáticos e validados em sumários manuais.

Os resultados obtidos foram:

Sistema	Precision Médio	Desvio padrão do Precision	Recall Médio	Desvio padrão do Recall
Sistema 1(TF-ISF)	0.0717857	0.0238868	0.0717857	0.0238868
Sistema 2 (Topical TF-ISF)	0.195278	0.0297532	0.195278	0.0297532
Sistema 3 C4.5	0.25504	0.0332067	0.25504	0.0332067
Sistema 3 Naïve-Bayes	0.25246	0.0327294	0.25246	0.0327294
Sistema 3/4 C4.5	0.243889	0.0283654	0.243889	0.0283654
Sistema 3/4 Naïve-Bayes	**0.261389**	**0.0332726**	**0.261389**	**0.0332726**
Sistema 4 C4.5	0.164802	0.0251246	0.164802	0.0251246
Sistema 4 Naïve-Bayes	0.238532	0.0342992	0.238532	0.0342992
Sistema 5 (TFISF C4.5)	0.0666667	0.0463206	0.0111111	0.0077200
Primeiras Sentenças	0.187857	0.0254693	0.187857	0.0254693
Sentenças Aleatórias	0.0972222	0.0177375	0.0972222	0.0177375
Word Summarizer	0.14229	0.0216963	0.172421	0.0256079

Tabela 6 - Taxa de acerto dos 12 sistema treinados com sumários automáticos contra sumários ideais manuais com 10% das sentenças do documento

O sistema com maior valor médio de *precision* e *recall* foi a combinação dos sistemas 3 e 4, utilizando o classificador Naïve-Bayes. Deve-se notar que os resultados estão muito abaixo dos experimentos 1 e 2, relatados nas duas seções anteriores, que utilizavam sumários gerados automaticamente. Algumas análises que podem ser retiradas dos resultados são:

- O sistema 1 mais uma vez apresentou resultados comparáveis à seleção aleatória de sentenças.
- O sistema 2 apresentou resultados praticamente idênticos à seleção das primeiras sentenças do texto.
- Os sistema 3 apresentou resultados muito similares para os classificadores Naïve-Bayes e C4.5.
- Os sistemas 4 e a combinação dos sistemas 3 e 4 apresentaram melhores resultados utilizando o classificador Naïve-Bayes do que o C4.5. Por fim, a

união das características dos sistemas 3 e 4 resultou na melhor taxa de acerto do experimento.

4.4.4 Análise da taxa de acerto dos 12 sistemas treinados com sumários automáticos contra sumários ideais manuais com 20% das sentenças do documento

Nesta análise, todos os sistemas foram avaliados na base de validação com sumários manuais com taxa de compressão de 20%. Para sistemas que necessitam de treinamento, a base de treinamento foi utilizada. Deve-se notar que a referida base de treinamento contém valores de classe baseados em sumários ideais automáticos, e não manuais, de forma que pode-se afirmar que os sistema foram treinados com sumários automáticos e validados em sumários manuais.

Os resultados obtidos foram:

Sistema	Precision Médio	Desvio padrão do Precision	Recall Médio	Desvio padrão do Recall
Sistema 1(TF-ISF)	0.177032	0.0177679	0.177032	0.0177679
Sistema 2 (Topical TF-ISF)	0.33682	0.0206268	0.33682	0.0206268
Sistema 3 C4.5	0.380042	0.0241849	0.380042	0.0241849
Sistema 3 Naïve-Bayes	**0.387705**	**0.0240577**	**0.387705**	**0.0240577**
Sistema ¾ C4.5	0.31729	0.0240873	0.31729	0.0240873
Sistema ¾ Naïve-Bayes	0.375006	0.0229808	0.375006	0.0229808
Sistema 4 C4.5	0.274021	0.0229665	0.274021	0.0229665
Sistema 4 Naïve-Bayes	0.355325	0.0220771	0.355325	0.0220771
Sistema 5 (TFISF C4.5)	0.158424	0.0317221	0.112355	0.0221937
Primeiras Sentenças	0.280085	0.0207736	0.280085	0.0207736
Sentenças Aleatórias	0.194918	0.0185397	0.194918	0.0185397
Word Summarizer	0.247948	0.0222047	0.275595	0.024078

Tabela 7 - Taxa de acerto dos 12 sistemas treinados com sumários automáticos contra sumários ideais manuais com 20% das sentenças do documento

O sistema com maior valor médio de *precision* e *recall* foi o sistema 3, utilizando o classificador Naïve-Bayes. Pode-se notar que com sumários de 20% de compressão, os resultados foram consideravelmente melhores do que utilizando 10% de compressão. Isto era esperado, conforme explicado na seção 4.4.2. Algumas análises que podem ser retiradas dos resultados são:

- O sistema 1 mais uma vez apresentou resultados comparáveis à seleção aleatória de sentenças.
- O sistema 2 apresentou resultados melhores que a seleção das primeiras sentenças do texto.
- Os sistema 3 apresentou resultados muito similares para os classificadores Naïve-Bayes e C4.5.
- Os sistemas 4 e a combinação dos sistemas 3 e 4 apresentaram melhores resultados utilizando o classificador Naïve-Bayes do que o C4.5. Por fim, a união das características dos sistemas 3 e 4 não resultou em ganhos na taxa de acerto, mas os resultados obtidos foram muito próximos do melhor resultado.

4.4.5 Comparação de sumários ideais automáticos contra sumários ideais manuais

O objetivo desta análise é medir o grau de concordância entre sumários extrativos produzidos automaticamente com sumários produzidos por juizes humanos. Tal comparação é de extrema valia, visto que [Mani 98a] propôs a geração de sumários automáticos, mas não efetuou nenhuma análise de aceitação nos mesmos.

Para realizar a avaliação, foram criados sumários extrativos automáticos para a base de validação com sumários manuais, utilizando as taxas de compressão de 10% e 20%. Os resultados foram estimados utilizando-se as métricas de *precision* e *recall* médio, considerando os 30 documentos componentes da base.

Os resultados obtidos foram:

Tamanho do sumário	Precision Médio	Desvio padrão do Precision	Recall Médio	Desvio padrão do Recall
10%	0.307937	0.0395799	0.307937	0.0395799
20%	0.429865	0.0242379	0.429865	0.0242379

Tabela 8 - Comparação entre sumários automáticos e manuais

Os resultados demonstram que sumários automáticos diferem consideravelmente de sumários manuais. Entretanto, mesmo juizes humanos não apresentam grau de concordância muito maior: em [Mitra 97], dois juizes humanos selecionaram as mesmas

sentenças em somente 46% das vezes. Desta forma, pode-se dizer que a discordância existente entre sumários automáticos e manuais é praticamente similar a existentes entre sumários manuais produzidos por pessoas diferentes. Entretanto, tal resultado explica parcialmente a baixa taxa de acerto relatada nos experimentos das seções 4.4.3 e 4.4.4, visto que os sistemas foram treinados em sumários automáticos e validados em sumários manuais. Devido a este procedimento, os limites superiores em termos de *precision* médio (e analogamente ao *recall* médio) para os experimentos das seções 4.4.3 e 4.4.4 são de 0.307937 e 0.429865, respectivamente.

4.4.6 Agrupamento dos sumários produzidos pelos 11 sistemas através do uso de *Clustering* Aglomerativo

Em [Mochizuki 00], diversas técnicas diferentes de sumarização foram comparadas entre si a partir da saída produzida, visando determinar grupos de algoritmos/heurísticas que produzem resultados similares. Utilizando-se um procedimento similar, o objetivo desta análise é agrupar os sistemas apresentados, a partir da saída produzida e do uso de *clustering* aglomerativo, para taxas de compressão de 10% e 20%. A base utilizada foi a base de validação com sumários automáticos.

O processo foi realizado da seguinte maneira:

1) Cada sumário produzido pelos sistemas é representado como um vetor. Para cada dimensão do vetor, o valor de 1.00 significa a seleção da respectiva sentença no sumário e o valor 0.00 em caso contrário.

2) Para cada técnica de sumarização é calculado um "vetor médio", que apresenta tamanho igual ao número de sentenças do maior documento da base e aonde os valores de cada dimensão são expressos como a média dos valores desta dimensão em todos os documentos. Caso a dimensão não exista no documento em questão, o valor é considerado 0.00.

3) Os vetores médios são agrupados consecutivamente dois a dois por similaridade através do uso de *clustering* aglomerativo, até que se obtenha um único elemento. Para métrica de distância entre os vetores, utilizou-se a distância euclidiana, expressa por:

$$d = \sqrt{\sum_{i=1}^{n} (s_i - t_i)^2}$$

sendo,

s, t: vetores a serem comparados

n: número de dimensões dos vetores (ambos devem ter o mesmo número de dimensões)

s_i: valor da dimensão i no vetor s

t_i: valor da dimensão i no vetor t

4) A árvore binária produzida no passo 3 permite identificar técnicas que apresentam saídas similares.

Para sumários com taxa de compressão de 10%, a árvore binária produzida foi:

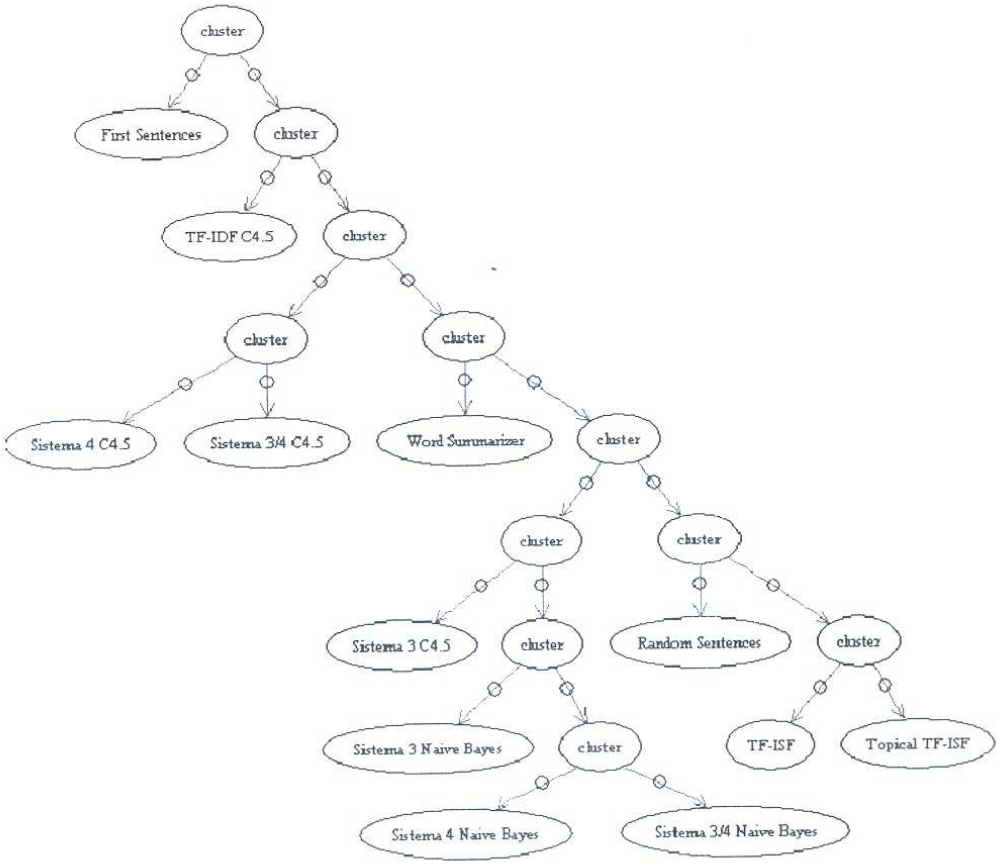


Figura 17 - Saída do *Clustering* Aglomerativo entre as técnicas para sumários de 10%

Para sumários com taxa de compressão de 20%, a árvore binária produzida foi:

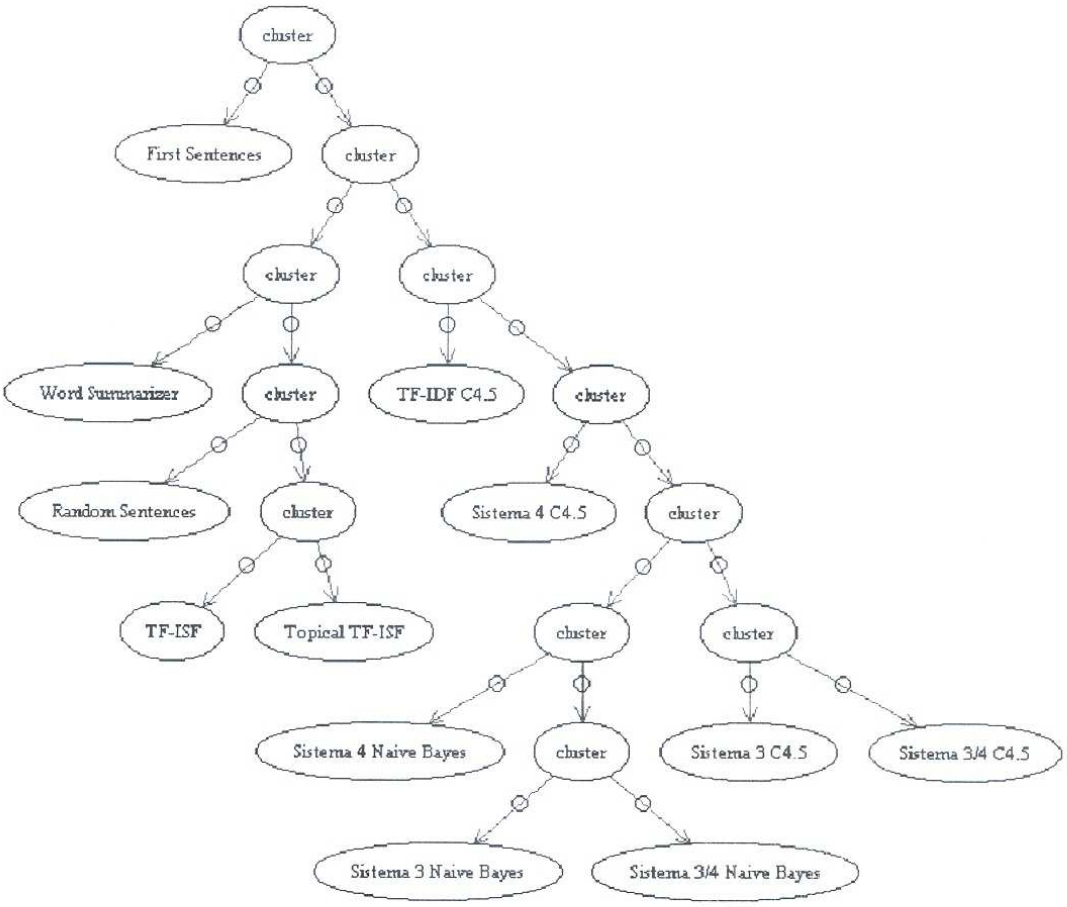


Figura 18 - Saída do *Clustering* Aglomerativo entre as técnicas para sumários de 20%

Algumas conclusões interessantes que podem ser encontradas nas árvores produzidas são:

- Os sistemas TF-ISF e Topical TF-ISF foram em ambos os casos agrupados no mesmo *cluster*, o que demonstra a similaridade conceitual dos algoritmos. Ambos foram também agrupados com o algoritmo de sentenças aleatórias, o que sugere que ambos apresentam um comportamento bastante distinto dos demais sistemas.
- O algoritmo First Sentences é muito diferente dos demais, como era de esperar, sendo a folha de nível mais alto da hierarquia gerada, embora seja uma base de comparação forte.

- De um modo geral, os sistemas treináveis utilizando o classificador C4.5 foram diretamente agrupados com outros sistemas que utilizavam o mesmo algoritmo, sendo que o mesmo ocorreu para sistemas que utilizavam o classificador Naive Bayes.
- Uma constatação importante é que os mesmos sistemas utilizando classificadores diferentes aparecem em pontos diferentes da hierarquia, o que é surpreendente, visto que os algoritmos utilizaram os mesmos atributos nos mesmos dados de treinamento. Eles estão em *clusters* bem distintos, o que sugere que eles selecionaram sentenças relativamente diferentes entre si. Esta é uma conclusão importante, pois levanta a hipótese da criação de sistemas híbridos, que utilizem dois ou mais sumarizadores combinados para produzir uma única saída.

4.4.7 Análise da taxa de acerto dos 12 sistemas treinados com sumários manuais contra sumários ideais manuais com 10% das sentenças do documento

Nesta análise, a base de sumários manuais foi particionada em conjuntos de treinamento e validação, e os sistemas foram avaliados na geração de sumários com taxa de compressão de 10%. Desta forma, este experimento não apresenta as desvantagens apresentadas nos experimentos das seções 4.4.3 e 4.4.4, na qual sistemas de sumarização treináveis utilizaram uma base de treinamento com sumários ideais automáticos. Entretanto, a base de treinamento com sumários manuais apresenta somente 20 documentos, contra 100 documentos na base de treinamento com sumários automáticos.

Os resultados obtidos foram:

Sistema	Precision Médio	Desvio padrão do Precision	Recall Médio	Desvio padrão do Recall
Sistema 1(TF-ISF)	0.0366667	0.0245704	0.0366667	0.0245704
Sistema 2 (Topical TS-ISF)	0.175238	0.0394967	0.175238	0.0394967
Sistema 3 C4.5	0.191905	0.0405247	0.191905	0.0405247
Sistema 3 Naïve-Bayes	0.287143	0.0367079	0.287143	0.0367079
Sistema 3/4 C4.5	0.13619	0.0381956	0.13619	0.0381956
Sistema 3/4 Naïve-Bayes	0.272857	0.0448286	0.272857	0.0448286
Sistema 4 C4.5	0.182857	0.0512686	0.182857	0.0512686

Sistema 4 Naïve-Bayes	**0.301429**	**0.0393321**	**0.301429**	**0.0393321**
Sistema 5 (TFISF C4.5)	0.151905	0.0394498	0.151905	0.0394498
Primeiras Sentenças	0.115238	0.035697	0.115238	0.035697
Sentenças Aleatórias	0.101905	0.0411672	0.101905	0.0411672
Word Summarizer	0.12619	0.0540674	0.110952	0.042828

Tabela 9 - Taxa de acerto dos 12 sistemas treinados com sumários manuais contra sumários ideais manuais com 10% das sentenças do documento

O sistema com maior valor médio de *precision* e *recall* foi o sistema 4, utilizando o classificador Naïve-Bayes. Pode-se notar que os resultados obtidos foram baixos, embora superiores ao experimento descrito na seção 4.4.3, que utiliza a base de treinamento com sumários automáticos. Os baixos resultados obtidos podem ser parcialmente justificados pelo pequeno número de exemplos de treinamento e ao fato de somente 10 documentos terem sido utilizados na obtenção da taxa de acerto de cada técnica, um número muito menor que os demais experimentos. Algumas análises que podem ser retiradas dos resultados são:

- O sistema 1 mais uma vez apresentou resultados comparáveis à seleção aleatória de sentenças.
- O sistema 2 apresentou resultados consideravelmente melhores que a seleção das primeiras sentenças do texto, embora inferior aos sistemas baseados em treinamento.
- Os sistemas 3, 4 e o híbrido 3/4 apresentaram resultados muito melhores utilizando o classificador Naïve-Bayes do que o C4.5.
- Os sistemas 3, 4 e o híbrido 3/4 apresentaram resultados muito próximos.
- O sistema 5 superou todas as bases de comparação, apresentando resultados próximos ao sistema 2.

4.4.8 Análise da taxa de acerto dos 12 sistemas treinados com sumários manuais contra sumários ideais manuais com 20% das sentenças do documento

Nesta análise, a base de sumários manuais foi particionada em conjuntos de treinamento e validação, e os sistemas foram avaliados na geração de sumários com taxa de compressão de 20%. Desta forma, este experimento não apresenta as desvantagens apresentadas nos experimentos das seções 4.4.3 e 4.4.4, na qual sistemas baseados em treinamento utilizaram uma base de treinamento com sumários ideais automáticos. Entretanto, a base de treinamento com sumários manuais apresenta somente 20 documentos, contra 100 documentos na base de treinamento com sumários automáticos.

Os resultados obtidos foram:

Sistema	Precision Médio	Desvio padrão do Precision	Recall Médio	Desvio padrão do Recall
Sistema 1(TF-ISF)	0.140618	0.0223044	0.140618	0.0223044
Sistema 2 (Topical TS-ISF)	0.278392	0.0299183	0.278392	0.0299183
Sistema 3 C4.5	0.300326	0.0474011	0.29366	0.0418161
Sistema 3 Naïve-Bayes	**0.345828**	**0.0544746**	**0.345828**	**0.0544746**
Sistema 3/4 C4.5	0.316573	0.0269164	0.316573	0.0269164
Sistema 3/4 Naïve-Bayes	0.338135	0.053187	0.338135	0.053187
Sistema 4 C4.5	0.28662	0.0338967	0.28662	0.0338967
Sistema 4 Naïve-Bayes	0.338135	0.053187	0.338135	0.053187
Sistema 5 (TFISF C4.5)	0.243275	0.0294303	0.243275	0.0294303
Primeiras Sentenças	0.238007	0.0254382	0.238007	0.0254382
Sentenças Aleatórias	0.134324	0.017107	0.134324	0.017107
Word Summarizer	0.205689	0.0564869	0.204336	0.0561999

Tabela 10 - Taxa de acerto dos 12 sistemas treinados com sumários manuais contra sumários ideais manuais com 20% das sentenças do documento

O sistema com maior valor médio de *precision* e *recall* foi o sistema 3, utilizando o classificador Naïve-Bayes. Pode-se notar que os resultados obtidos foram baixos, e inferiores aos descritos na seção 4.4.4, que utiliza a base de treinamento com sumários automáticos. Surpreendentemente, os resultados foram apenas ligeiramente superiores aos obtidos em sumários com taxa de compressão de 10%, ao contrário dos experimentos 4.4.4 e 4.4.2, aonde sumários de 20% do texto apresentaram resultados consideravelmente melhores do que os sumários de 10%. Algumas análises que podem ser retiradas dos resultados são:

- O sistema 1 mais uma vez apresentou resultados comparáveis à seleção aleatória de sentenças.
- O sistema 2 apresentou resultados consideravelmente melhores que a seleção das primeiras sentenças do texto, e relativamente próximo dos sistemas baseados em treinamento.
- Os sistemas 3, 4 e o híbrido 3/4 apresentaram resultados muito melhores utilizando o classificador Naïve-Bayes do que o C4.5.
- Os sistemas 3, 4 e o híbrido 3/4 apresentaram resultados praticamente idênticos.
- O sistema 5 superou todas as bases de comparação, apresentando resultados próximos ao sistema 2.

5 Conclusões e direções futuras

Como resultados finais do trabalho, pode-se salientar:

- O sistema 1 (TF-ISF) apresenta performance similar à seleção aleatória de sentenças, invalidando a técnica como método de sumarização automática de textos.
- O sistema 2 (Topical TF-ISF) apresentou resultados similares ou ligeiramente superiores à seleção das primeiras sentenças do texto, uma base de comparação considerada forte em experimentos realizados na literatura [Mitra 97] [Brandow 94]. Logo, o sistema 2 não é muito vantajoso, pois resultados de qualidade semelhante podem ser obtidos com uma heurística muito mais simples e de rápida execução.
- Os sistemas 3, 4 e a combinação dos mesmos foram responsáveis pelos melhores resultados em todos os experimentos, superando com boa margem os sistemas utilizados como base de comparação (seleção aleatória de sentenças, seleção das primeiras sentenças do texto e Microsoft Word Summarizer).
- O classificador Naive-Bayes apresentou performance consistentemente superior ao classificador C4.5 em sistemas de sumarização treináveis. Os melhores resultados do classificador NaiveBayes sinalizam a existência de interação entre os atributos utilizados na tarefa de sumarização, visto que este classificador se comporta melhor nesta situação do que o classificador C4.5. Tal fato pode também ser confirmado devido a característica TF-ISF apresentar baixos resultados isoladamente (no sistema 1 e 5), e bons resultados quando combinada com outras características (nos sistemas 3 e 3/4).
- Sumários automáticos apresentam um grau de discordância com sumários manuais similar ao grau de discordância entre sumários produzidos por dois juizes humanos.
- Os resultados obtidos utilizando-se sumários ideais manuais foram consideravelmente inferiores aos obtidos utilizando-se sumários automáticos. Entretanto, deve-se lembrar que em alguns experimentos os sistemas foram treinados sob sumários automáticos e validados em sumários manuais, o que limitou a avaliação. Em outros experimentos, os sistemas foram treinados e validados com sumários manuais, mas neste caso o número de documentos para treinamento foi relativamente pequeno.

- Sistemas que apresentaram os melhores resultados não necessariamente selecionam as mesmas sentenças do texto, e podem ser utilizados em sistemas híbridos de sumarização, que utilizem dois ou mais sumarizadores combinados para produzir uma única saída.

Este trabalho apresenta diversas possibilidades de expansão e direções futuras de pesquisa, a saber:

- A sumarização baseada em treinamento parece ser o caminho mais natural de pesquisa futura, considerando-se os resultados obtidos na literatura e no presente trabalho e a capacidade de adaptação para diferentes tipos e gêneros de texto. Entretanto, o uso de técnicas de aprendizado de máquina ainda está limitado. Neste contexto, mais experimentos devem ser efetuados em técnicas do paradigma simbólico [Quinlan 93], conexionista [Zell et al 95] e aprendizado baseado em instância [Dasarathy 91].
- Ainda na sumarização baseada em treinamento, deve-se explorar o uso de características de treinamento mais sofisticadas, envolvendo técnicas de processamento em linguagem natural como a análise da contribuição retórica e morfológica de cada sentença, uso de bases lexicográficas como a WordNet, e métricas de recuperação de informações como MMR e pesos TF-TL. Uma outra possibilidade relacionada a esta direção de pesquisa consiste em usar métodos de construção indutiva para construir novas características, a partir das características individuais propostas neste trabalho.
- Avaliações baseadas em sumários ideais devem ser substituídas por análises de aceitabilidade e captura das idéias principais do texto ou em tarefas extrínsecas (determinadas cuidadosamente para não influenciarem os resultados da sumarização), evitando restringir a avaliação excessivamente, baseado em críticas realizadas à aceitação de um "sumário ideal" para um documento e na alta discordância entre juizes humanos.

6 Referências Bibliográficas

[Baldwin 98] Baldwin, B. ; Morton, T. Coreference-Based Summarization. In *Proceedings of TIPSTER Text Phase III Workshop*. 1998.

[Banko 99] Banko, M. ; Mittal, V. ; Kantrowitz, M. ; Goldstein, J. Generating Extraction Based Summaries from Handwritten Summaries by Aligning Text Spans. In *Proceedings of PacLing-99*. 1999.

[Barzilay 97] Barzilay, R. ; Elahad, M. Using Lexical Chains for Text Summarization. In Mani, I. e Maybury, M. T., eds., In *Proceedings of the ACL/EACL-97 Workshop on Intelligent Scalable Text Summarization*. Association of Computional Linguistics. 1997.

[Brandow 94] Brandow, R. ; Mitze, K. , Rau, L. Automatic condensation of eletronic publications by sentence selection. *Information Processing and Management*, 31, 5 , 675-685. 1994.

[Brill 92] Brill, E. A simple rule-based part-of-speech tagger. In *Proccedings of the Third Conference on Applied Computational Linguistics*.Association for Computational Linguistics. 1992.

[Carbonell 98] Carbonell, J. G.; Goldstein, J. the use of MMR, diversity-based reranking for reordering documents and producing summaries. In *Proccedings of SIGIR-98*. 1998.

[Carbonell 99] Carbonell, J. G.; Goldstein, J. Kantrowitz, M. ; Mittal, V. O. Summarizing Text Documents: Sentence Selection and Evaluation Metrics. In *Proccedings of SIGIR 1999*: 121-128. 1999.

[Cavnar 94] Cavnar, W. B. Using An N-Gram-Based Document Representation With a Vector Processing Retrieval Model. In *Proceedings Of TREC-3 (Third Text Retrieval Conference)*. Gaithersburg, Maryland, USA. 1994.

[Chuang 00] Chuang, W. ; Yang, J. Estracting Sentence Segments for Text Summarization: A Machine Learning Approach. *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'00)*, Athens, Greece. 2000.

[Dasarathy 91] Dasarathy BV. (Ed.) *Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques*. IEEE Computer Society Press, 1991.

[Edmundson 69] Edmundson, H. P. new methods in automatic extracting. *Journal of the Association for Computing Machinery* 16(2):2644-285. 1969.

[Harabagiu 99] Harabagiu, S. From Lexical Cohesion To Textual Coherence: A Data Driven Perspective. *Journal of Pattern Recognition and Artificial Intelligence, Vol 13, No. 2*. 1999.

[Harman 94] Harman, D. Data Preparation. In R. Merchant, editor, *The Proceedings of the TIPSTER Text Program Phase I*. Morgan Kaufmann Publishing Co. 1994.

[Hearst 93] Hearst, Marti A. TextTiling: A Quantitative Approach to Discourse Segmentation. *Technical Report 93/24*. University of California, Berkeley. 1993.

[Joachims 96] Joachims, T. A Probabilistic Analysis of Rocchio Algorithm with TFIDF for Text Categorization. *Technical Report CMU-CS-96-118*. Departament of Computer Science, Carnegie Melow University. 1996.

[Kohavi 96] Kohavi, R. ; Sommerfield, D. ; Dougherty, J. Data Mining using MLC++: A Machine Learning Library in C. In *Tools with AI 1996*. 1996.

[Kupiec 95] Kupiec, J. ; Pedersen, J. O.; Chen, F. A trainable document summarizer. In *Proceedings of the 18th ACM-SIGIR Conference, Association of Computing Machinery, Special Interest Group Information Retrieval*, 68-73. 1995.

[Larocca 00a] Larocca Neto, Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. . Document clustering and text summarization. *Proceedings of 4th Int. Conf. Practical Applications of Knowledge Discovery and Data Mining (PADD-2000)*, 41-55. London: The Practical Application Company.

2000.

[Larocca 00b] Larocca Neto, Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. . Generating Text Summaries through the Relative Importance of Topics. *Proceedings of IBERAMIA-2000 (7th Ibero-American Conf. on Artif. Intel.) & SBIA-2000 (15th Brazilian Symp. on Artif. Intel.)*. Sao Paulo, SP, Brazil. Nov. 2000.

[Larocca 00c] Larocca Neto, Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. ; Nievola, Júlio C. . Sumarização como um problema de classificação. *Technical Report PPGIA*. Pontifícia Universidade Católica. Jul. 2000.

[Larocca 00d] Larocca Neto, Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. ; Nievola, Júlio C. . A trainable algorithm for summarizing news stories. *Proceedings of PKDD-2000 Workshop on Machine Learning and Textual Information Access*. Lyon, France. Sep. 2000.

[Lin 95] Lin, C. Y. Knowledge-based automatic topic identification. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL-95)*, pp 308-310. 1995.

[Luhn 58] Luhn, H. The automatic Creation of Literature Abstracts. *IBM Journal of Research and Development* 2(92):159-165. 1958.

[Mani 98a] Mani, I.; Bloedorn, E. Machine Learning of Generic and User-Focused Summarization. In *Proceedings of the Fifteenth National Conference on AI (AAAI-98)*, 821-826, 1998

[Mani 98b] Mani, I.; Bloedorn, E ; Gates, B. Using Cohesion and Coherence Models for Text Summarization. 1998 *AAAI Symposium Technical Report SS-989-06*. AAAI Press. 1998.

[Mani 98c] Mani, I.; House, D.; Klein, G.; Hirschman, L.; Obrsl, L.; Firmin, T.; Chrzanowski, M.; Sundheim, B. *The TIPSTER SUMMAC Text Summarization Evaluation.* MITRE Technical Report MTR 98W0000138. The MITRE Corporation, Oct. 1998.

[Marcu 99] Marcu, D. Discourse trees are good indicators of importance in text. In I. Mani and M. Maybury editors, *Advances in Automatic Text Summarization*, pages 123-136, The MIT Press. 1999.

[Miller 90] Miller, G.; Beckwith, R. ; Fellbaum, C. ; Gross, D.; Miller, K. Five papers on WordNet. *Technical Report CSL Report 43*, Cognitive Science Laboratory, Princeton University.

[Mitra 97] Mitra, M. ; Singhal, A. ; Buckley, C. Automatic Text Summarization by Paragraph Extraction. In *Proceedings of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*. Madrid, Spain. 1997.

[Mochizuki 00] Mochizuki, H. ; Okumura, M. A Comparison of Summarization Methods Based on Task-Based Evaluation. *Proceedings of LREC2000.* 2000.

[Morris 92] Morris, A. H.; Kasper, G. M. ; Adams, D. A. The effects and limitations of Automated Text Condensing on Reading Comprehension Performance. *Information Systems Research* 3:1 pages 17-35. 1992.

[Nevill-Manning 99] Nevill-Manning, C. G. ; Witten, I. H. Paynter, G. W. et al KEA: Practical Automatic Keyphrase Extraction. *ACM DL 1999*: 254-255, 1999.

[Porter 80] Porter, M.F. An algorithm for suffix stripping. *Program 14*, 130-137. 1980. Reprinted in: Sparck Jones, K. and Willet, P. (Eds.) *Readings in Information Retrieval*, 313-316. Morgan Kaufmann, 1997.

[Quinlan 93] Quinlan, J. *C4.5: Programs for Machine Learning*. Morgan Kaufmann, Sao Mateo, CA. 1992.

[Radev 00] Radev, D. R. ; Jing, H. ; Budzikowska, M. . Summarization of multiple documents: clustering, sentence extraction, and evaluation. In *ANLP/NAACL Workshop on Summarization*. 2000.

[Rath 61] Rath, G. J. ; Resnick A. ; Savvage R. The formation of abstracts by the selection of sentences: Part 1: sentence selection by man and machines. *American Documentation* 12 (2) pp 139-141. 1961.

[Sanderson 98] Sanderson, M. Accurate user directed summarization from existing tools. In the *Proceedings of the 7th International Conference on Information and Knowledge Management (CIKM 98)*. 1998.

[Salton 88] Salton, G. and Buckley, C. Term-weighting approaches in automatic text retrieval. *Information Processing and Management 24,* 513-523. 1988. Reprinted in: Sparck Jones, K. and Willet, P. (Eds.) *Readings in Information Retrieval*, 323-328. Morgan Kaufmann, 1997.

[Spark Jones 99] Sparck Jones, K. Automatic Summarizing: factors and directions. In Mani, I. Maybury, M., *Advances in automatic Text Summarization*, pages 1-12. The MIT Press. 1999.

[Strzalkowski 99] Strzalkowski , T. ; Stein, G. ; Wang, J. ; Wise, B. A Robust Practical Text Summarizer. In I. Mani and M. Maybury editors, *Advances in Automatic Text Summarization.* The MIT Press. 1999.

[Teufel 99] Teufel, S. ; Moens, M. Argumentative classification of extracted sentences as a first step towards flexible abstracting. In: I. Mani, M. Maybury (eds.), *Advances in automatic Text Summarization*, MIT Press, 1999

[Turney 00] Turney, P.D. Learning algorithms for keyphrase extraction. *Information Retrieval*, 2 (4). 2000.

[Witten 94] Witten, Ian H.; Moffat, Alistair; Bell, Timothy C. Managing Gigabytes. *Van Nostrand Reinhold.* New York. 1994.

[Yaari 97] Yaari, Y. Segmentation of Expository Texts by Hierarchical Agglomerative Clustering. *Technical Report*, Bar-Ilan University, Israel. 1997.

[Zechner 95] Zechner, K. Automatic text abstracting by selecting relevant passages. *Master's thesis*, Centre for Cognitive Science, University of Edinburgh.

[Zell et al 95] Zell, A et al. Stuttgart Neural Network Simulator(SNNS) User Manual, Version 4.1 *Report No. 6/95* University of Stuttgart.

APÊNDICE: EXEMPLO COMPLETO DE SUMARIZAÇÃO

Esta seção contém um exemplo completo de sumarização em um texto da base TIPSTER [Harman 94]. O texto completo, seguido dos sumários providos pelo autor, gerado automaticamente e a saída de cada um dos 12 sistemas são apresentados a seguir.

Número do Documento: ZF32-153-237

Publicação: Datamation May 15 1991 v37 n10 p93(4) * Full Text COPYRIGHT Cahners Publishing Co. 1991.

Título: Relational power spans platforms. (relational database management systems move from mainframes to microcomputers) (buyers guide)

Autor: Graham, Carol.

Texto Completo

[1] Advances in database technology have significantly increased the power, versatility and performance of relational database management systems (RDBMSS), enabling them to distribute data to virtually every corner of today's business environment.
[2] From decision support to office automation, RDBMSs integrate the necessary features to enhance productivity and enrich the process of information exchange.
[3] The proliferation of high-powered RDBMS products across ail hardware platforms and operating environments gives midrange and microcomputer users access to functions that were once only accessible through mainframes.
[4] Most existing databases use either hierarchical or relational technology.
[5] Although hierarchical DBMSs have dominated the market, the relational DBMS (or some variation thereco is now the database of choice.
[6] Relational DBMSs allow users to store and maintain information according to the relationships among the various data elements.
[7] Hierarchical database design, in contrast, produces a more structured database with information stored in specific file locations.
[8] Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.
[9] Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.
[10] Despite its advantages, relational technology still requires a cautious evaluation from two perspectives: the differing requirements
[11] each hardware platform and the degree to which a particular product is relational.
[12] Yet despite these potential obstacles, many users say the features associated with relational technology are exactly what they want and need to organize and process information.
[13] As a result of market demand, more and more vendors are offering RDBMS products that permit separate databases to coexist as a single entity.

[14] Vendors are also providing users with a variety of alternatives for accessing disparate databases, including gateways, single-database strategies and data propagators.
[15] Several RDBMS vendors have introduced gateway products that enable programmers to access existing databases with the latest Structured Query Language SQL)-based access tools, such as fourth-generation languages (4GLs), graphically oriented query facilities and computer-aided software engineering (CASE) products.
[16] The gateway software ensures that the data retrieval process is transparent to the user or programmer, eliminating or at least minimizing) concern for the specific query dialect, error handling and dictionary conventions, or communications protocol.
[17] Disadvantages of such gateways include degradation of system performance and price tags ranging from $1,000 to more than $100,000 per system
[18] Database Unity
[19] A number of major vendors, including Cincom Systems Inc. and Computer Associates International Inc. (CA), advocate a single-database strategy. CA's emulator product can, for example, physically migrate nonrelations database information form other DBMSs, such as Cincom's Total or IBM's IMS, DL/1 or VSAM into current versions of CA's CA-IDMS or CADATACOM/DB, enabling users to run existing applications without modification.
[20] Cincom Systems's SUPRA helps users of IBM IMS, DL/1 and VSAM databases avoid database migration altogether.
[21] It permits users to retain the existing data structures while using SUPRA's SQL and 4GL facilities to develop relational-based applications.
[22] The single database a approach adopted by CA and Cincom differs from the database concept used by IBM and Digital Equipment Corp. Both offer a class of low-level hooks between relational and nonrelational databases (known as data propagators or distributors) that permit old and new databases to be merged into a single entity.
[23] Analysts predict that, by 1993, facilities will be widely available from virtually all of the major vendors to assist users in migrating and converting existing databases-whether they use hierarchical, flat file or hybrid designs.
[24] Future Trends
[25] Database systems are poised to move well beyond the functional limitations of today's relational products.
[26] Although the majority of new DBMS products will continue to use relational technology, they are also likely to add interfaces to such technologies as optical storage, expert systems and object orientation.
[27] The information retained in future databases is likely to incorporate much more than the rows and columns that form the basis of relational databases.
[28] In addition to changes in information technology, DBMS product development is strongly influenced by an ongoing tug of war between vendors and users. Today, users rather than vendors are driving product development.
[29] This trend is evidenced by the increasing emphasis on industry standards such as SQL, the use of open architectures to ensure wide spread product compatibility and th availability of links to existing DBMS prod ucts.
[30] IBM's development efforts and associated product introductions particularly in the areas of information management and systems development-ar likely to shape the future of the DBM market as competitors attempt to emu late or surpass IBM's offerings.
[31] Similarly, the ongoing trend toward cooperative processing (such as client/ server architectures and heterogeneous computing) is likely to force software vendors to develop DBMS products that can be easily shared across network lines.
[32] In the not-too-distant future, then, relational and nonrelational databases residing on disparate hardware platforms will be able to talk to one another and exchange information with minimal user or programmer intervention.

Sumário provido pelo autor

The database management system (DBMS) market has been dominated by hierarchical DBMSs, but now relational DBMSs are the systems of choice. With relational DBMSs, users can store and maintain

information according to the relations among the data elements. Hierarchical DBMSs are more structured. Relational DBMSs allow greater flexibility in defining, storing and retrieving data. Relational DBMS vendors are providing a variety of means for accessing databases, including data propagators, gateways and single-database strategies. Gateway software provides users and programmers a transparent data retrieval process; drawbacks include performance degradation and cost. Some major vendors of DBMSs recommend a single-database strategy and provide tools for integrating disparate databases. A list of some relational DBMSs is provided, including product name, price, vendor, environment and features.

Sumário extrativo automático

Most existing databases use either hierarchical or relational technology.
Although hierarchical DBMSs have dominated the market, the relational DBMS (or some variation thereco is now the database of choice.
Relational DBMSs allow users to store and maintain information according to the relationships among the various data elements.
Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.
Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.
Vendors are also providing users with a variety of alternatives for accessing disparate databases, including gateways, single-database strategies and data propagators.

Sumário Sistema 3 (C4.5)

Advances in database technology have significantly increased the power, versatility and performance of relational database management systems (RDBMSS), enabling them to distribute data to virtually every corner of today's business environment.
Most existing databases use either hierarchical or relational technology.
Hierarchical database design, in contrast, produces a more structured database with information stored in specific file locations.
Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.
Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.
Vendors are also providing users with a variety of alternatives for accessing disparate databases, including gateways, single-database strategies and data propagators.

Sumário Sistema 4 (C4.5)

Advances in database technology have significantly increased the power, versatility and performance of relational database management systems (RDBMSS), enabling them to distribute data to virtually every corner of today's business environment.
From decision support to office automation, RDBMSs integrate the necessary features to enhance productivity and enrich the process of information exchange.
The proliferation of high-powered RDBMS products across ail hardware platforms and operating environments gives midrange and microcomputer users access to functions that were once only accessible through mainframes.
Most existing databases use either hierarchical or relational technology.
Hierarchical database design, in contrast, produces a more structured database with information stored in specific file locations.

Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.

Sumário Sistema 3/4 (C4.5)

Advances in database technology have significantly increased the power, versatility and performance of relational database management systems (RDBMSS), enabling them to distribute data to virtually every corner of today's business environment.
From decision support to office automation, RDBMSs integrate the necessary features to enhance productivity and enrich the process of information exchange.
Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.
Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.
Yet despite these potential obstacles, many users say the features associated with relational technology are exactly what they want and need to organize and process information.
The gateway software ensures that the data retrieval process is transparent to the user or programmer, eliminating or at least minimizing) concern for the specific query dialect, error handling and dictionary conventions, or communications protocol.

Sumário Sistema 3 (Naive Bayes)

Advances in database technology have significantly increased the power, versatility and performance of relational database management systems (RDBMSS), enabling them to distribute data to virtually every corner of today's business environment.
Most existing databases use either hierarchical or relational technology.
Hierarchical database design, in contrast, produces a more structured database with information stored in specific file locations.
Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.
Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.
The single database a approach adopted by CA and Cincom differs from the database concept used by IBM and Digital Equipment Corp. Both offer a class of low-level hooks between relational and nonrelational databases (known as data propagators or distributors) that permit old and new databases to be merged into a single entity.

Sumário Sistema 4 (Naive Bayes)

Most existing databases use either hierarchical or relational technology.
Relational DBMSs allow users to store and maintain information according to the relationships among the various data elements.
Hierarchical database design, in contrast, produces a more structured database with information stored in specific file locations.
Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.

Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.
The single database a approach adopted by CA and Cincom differs from the database concept used by IBM and Digital Equipment Corp. Both offer a class of low-level hooks between relational and nonrelational databases (known as data propagators or distributors) that permit old and new databases to be merged into a single entity.

Sumário Sistema 3/4 (Naive Bayes)

Most existing databases use either hierarchical or relational technology.
Hierarchical database design, in contrast, produces a more structured database with information stored in specific file locations.
Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.
Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.
Several RDBMS vendors have introduced gateway products that enable programmers to access existing databases with the latest Structured Query Language SQL)-based access tools, such as fourth-generation languages (4GLs), graphically oriented query facilities and computer-aided software engineering (CASE) products.
The single database a approach adopted by CA and Cincom differs from the database concept used by IBM and Digital Equipment Corp. Both offer a class of low-level hooks between relational and nonrelational databases (known as data propagators or distributors) that permit old and new databases to be merged into a single entity.

Sumário Sistema TF-IDF (C4.5)

Advances in database technology have significantly increased the power, versatility and performance of relational database management systems (RDBMSS), enabling them to distribute data to virtually every corner of today's business environment.
Most existing databases use either hierarchical or relational technology.

Sumário Sistema TF-ISF

Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.
Despite its advantages, relational technology still requires a cautious evaluation from two perspectives: the differing requirements
Several RDBMS vendors have introduced gateway products that enable programmers to access existing databases with the latest Structured Query Language SQL)-based access tools, such as fourth-generation languages (4GLs), graphically oriented query facilities and computer-aided software engineering (CASE) products.
Disadvantages of such gateways include degradation of system performance and price tags ranging from $1,000 to more than $100,000 per system
The single database a approach adopted by CA and Cincom differs from the database concept used by IBM and Digital Equipment Corp. Both offer a class of low-level hooks between relational and nonrelational databases (known as data propagators or distributors) that permit old and new databases to be merged into a single entity.

In addition to changes in information technology, DBMS product development is strongly influenced by an ongoing tug of war between vendors and users. Today, users rather than vendors are driving product development.

Sumário Sistema Topical TF-ISF

Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.

Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.

Several RDBMS vendors have introduced gateway products that enable programmers to access existing databases with the latest Structured Query Language SQL)-based access tools, such as fourth-generation languages (4GLs), graphically oriented query facilities and computer-aided software engineering (CASE) products.

A number of major vendors, including Cincom Systems Inc. and Computer Associates International Inc. (CA), advocate a single-database strategy. CA's emulator product can, for example, physically migrate nonrelations database information form other DBMSs, such as Cincom's Total or IBM's IMS, DL/1 or VSAM into current versions of CA's CA-IDMS or CADATACOM/DB, enabling users to run existing applications without modification.

The single database a approach adopted by CA and Cincom differs from the database concept used by IBM and Digital Equipment Corp. Both offer a class of low-level hooks between relational and nonrelational databases (known as data propagators or distributors) that permit old and new databases to be merged into a single entity.

In addition to changes in information technology, DBMS product development is strongly influenced by an ongoing tug of war between vendors and users. Today, users rather than vendors are driving product development.

Sistema Primeira Sentenças

Advances in database technology have significantly increased the power, versatility and performance of relational database management systems (RDBMSS), enabling them to distribute data to virtually every corner of today's business environment.

From decision support to office automation, RDBMSs integrate the necessary features to enhance productivity and enrich the process of information exchange.

The proliferation of high-powered RDBMS products across ail hardware platforms and operating environments gives midrange and microcomputer users access to functions that were once only accessible through mainframes.

Most existing databases use either hierarchical or relational technology.

Although hierarchical DBMSs have dominated the market, the relational DBMS (or some variation thereco is now the database of choice.

Relational DBMSs allow users to store and maintain information according to the relationships among the various data elements.

Sistema Sentenças Aleatórias

The proliferation of high-powered RDBMS products across ail hardware platforms and operating environments gives midrange and microcomputer users access to functions that were once only accessible through mainframes.

Relational DBMSs allow users to store and maintain information according to the relationships among the various data elements.

Despite its advantages, relational technology still requires a cautious evaluation from two perspectives: the differing requirements

As a result of market demand, more and more vendors are offering RDBMS products that permit separate databases to coexist as a single entity.

Vendors are also providing users with a variety of alternatives for accessing disparate databases, including gateways, single-database strategies and data propagators.

Several RDBMS vendors have introduced gateway products that enable programmers to access existing databases with the latest Structured Query Language SQL)-based access tools, such as fourth-generation languages (4GLs), graphically oriented query facilities and computer-aided software engineering (CASE) products.

Sistema Word Summarizer

Most existing databases use either hierarchical or relational technology.

Hierarchical database design, in contrast, produces a more structured database with information stored in specific file locations.

Although relational database users do not need to know the physical file structure in order to access data, users of hierarchical databases must know some details of the database design in order to store and retrieve information.

Both relational and hierarchical databases use unique data fields called "keys" to identify specific portions of data, but relational databases typically offer a greater degree of flexibility in' information storage and retrieval-and they're particularly well suited for processing large amounts of data.

Vendors are also providing users with a variety of alternatives for accessing disparate databases, including gateways, single-database strategies and data propagators.

Future Trends