**ALINE RODRIGUES FERREIRA**

# CONTRIBUIÇÃO AO ESTUDO DA SUMARIZAÇÃO AUTOMÁTICA DE TEXTOS: RELAÇÕES SEMÂNTICAS ENTRE ELEMENTOS TEXTUAIS

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática Aplicada.

**CURITIBA**

**2004**

**ALINE RODRIGUES FERREIRA**

# CONTRIBUIÇÃO AO ESTUDO DA SUMARIZAÇÃO AUTOMÁTICA DE TEXTOS: RELAÇÕES SEMÂNTICAS ENTRE ELEMENTOS TEXTUAIS

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática Aplicada.

Área de Concentração: Sistemas Inteligentes

Orientador: Prof. Dr. Celso Antônio Alves Kaestner

**CURITIBA**

**2004**

Ferreira, Aline Rodrigues

Contribuição ao Estudo da Sumarização Automática de Textos: Relações Semânticas entre Elementos Textuais.

Curitiba, 2004. 65p.

Dissertação (Mestrado) – Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Informática Aplicada.

1. Sumarização de Textos 2. Grafos de Relacionamento entre substantivos 3. Wordnet 4. Aprendizagem de Máquina. Pontifícia Universidade Católica do Paraná. Centro de Ciências Exatas e de Tecnologia. Programa de Pós-Graduação em Informática Aplicada.

# TERMO DE APROVAÇÃO

# Agradecimentos

**“Elevo os meus olhos para o monte: de onde virá o meu socorro?
O meu socorro vem do Senhor, que fez o céu e a terra”.
Salmos 121:1-2**

À Deus, por ter me guiado e concedido discernimento para fazer escolhas tão certas quanto as que venho fazendo.

À minha mãe, por toda paciência, apoio e compreensão nesses últimos anos.

Aos meus queridos amigos Gisele, Cristiane, Fernanda, Fernando, Carlos, Daniella, David, Evandro, Otávio e Díbio, pelos momentos de estudo e descontração.

À Capes, pelo suporte financeiro concedido para que essa pesquisa fosse realizada.

Ao meu orientador Professor Celso Kaestner pela paciência, orientação e longas discussões sobre os rumos deste trabalho. E aos professores Alex Freitas e Julio César Nievola pela atenção e apoio para com o desenvolvimento do trabalho.

E a todos aqueles que de alguma maneira contribuíram para que esse trabalho fosse realizado.

# SUMÁRIO

# LISTA DE FIGURAS

# LISTA DE TABELAS

# RESUMO

A quantidade de informações disponíveis em forma textual está continuamente crescendo, e os usuários dispõem de cada vez menos tempo para acessar todas estas informações, fazendo com que mecanismos para a sumarização automática de textos se tornem ferramentas indispensáveis.

Este trabalho propõe e implementa um sistema para sumarização de texto, utilizando como ferramenta a criação de um grafo que indica relacionamentos semânticos entre os elementos de um texto, a partir de relações semânticas - tais como hipônimos e hiperônimos - extraídas do sistema de referências léxicas Wordnet.

O grafo é então utilizado para a extração de diversas características que são empregadas por um sumarizador de textos baseado em aprendizagem de máquina. O sumarizador emprega fundamentalmente o algoritmo Naïve-Bayes e os mecanismos usuais para treinamento e classificação. O sistema é aplicado a coleções de documentos extraídas da base TIPSTER, sendo apresentados os resultados obtidos.

**Palavras Chave: 1. Sumarização de textos, 2. Aprendizagem de máquina, 3. Recuperação de Informações, 4. Semântica Textual, 5. Wordnet**.

# ABSTRACT

Automatically text summarization is a crucial task in the modern world, where the amount of available text information grows exponentially, while the time users dedicate to analyse these information reduces potentially.

This work proposes and implements a text summarization system, using as a tool a graph creation that points semantic relationships among elements in a text. The considered semantic relations, such as hyponyms and hipernyms, were extract from a thesaurus named WordNet.

In a second step, the created graph is used to extract a set of features that will be employed by a text summarizer based in machine learning. The summarizer utilizes the Naïve-Bayes algorithm, and the usual methods for training and classifying. The system was applied to document collections extracted from the TIPSTER database.

**Key-words: 1. Texts Summarization, 2. Machine Learning, 3. Information Retrieval, 4. Textual Semantic, 5. Wordnet**.

# 1. INTRODUÇÃO

A quantidade de informações disponíveis de forma textual está crescendo cada vez mais, e há cada vez menos tempo para ler todas estas informações e tomar decisões baseadas em seu conteúdo. Dessa forma, a sumarização automática de textos parece ser uma ferramenta indispensável para auxiliar a solução desse problema.

Segundo [Mani 01], um *sumarizador* é um sistema cujo objetivo é produzir uma representação condensada do conteúdo de um texto para consumo humano. Neste contexto, a principal característica que difere a sumarização de outras tarefas é que a condensação da informação contida no documento tem por objetivo atender a realização de uma tarefa específica e beneficiar ao leitor.

Os sumários podem ser produzidos a partir de diversos tipos de entrada, como: imagens, sons, textos ou a combinação de todos eles. Mas em sumarização automática a principal entrada é a textual.

Existem basicamente três 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, e expressa a opinião do revisor em relação ao texto original.

Pelo fato da tecnologia necessária para produzir os sumários estar além da atual, e poucas aplicações utilizarem os sumários críticos, a pesquisa em sumarização até o momento restringe-se ao desenvolvimento de sumários informativos.

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

- Análise: constitui-se da interpretação do texto para criar uma representação abstrata do mesmo;

- Transformação: corresponde à passagem da representação do texto origem para uma representação de sumário;

- Síntese: Geração do sumário a partir da representação de sumário gerada no passo anterior.

A sumarização automática denominada *extrativa* utiliza técnicas estatísticas e empíricas para identificar as partes mais importantes do texto, e utiliza elementos extraídos diretamente do texto, tais como sentenças, para formar o sumário final.

Os métodos básicos de sumarização, em termos do espaço lingüístico, podem seguir duas abordagens:

- Abordagem Superficial: Nesta abordagem os diferentes elementos do texto são utilizados em um nível sintático, usualmente com a aplicação de técnicas estatísticas. Esta abordagem normalmente produz sumários cuja principal vantagem é a robustez.

- Abordagem Profunda: Esta abordagem assume pelo menos um nível de representação semântico para as sentenças, envolvendo análise e geração de linguagem natural ou representação nível de discurso.

Em sumarização, quanto às técnicas baseadas em estruturas do discurso, dois elementos são de fundamental importância: a coerência e a coesão do texto [Mani 98b].

- A coesão envolve relações entre palavras e seu sentido, anáforas, elipses, conjunções e relações léxicas como sinônimos, hiperônimos

e hipônimos, sendo representadas em termos de ligações entre os elementos do texto, onde são expressas as relações semânticas existentes entre estes elementos.

- A coerência envolve a descoberta de relações de argumentação entre as sentenças e as cláusulas do texto, como as palavras “Apesar” e “Por exemplo”, que indicam determinados tipos de relações entre as cláusulas envolvidas. Estas relações determinam a estrutura argumentativa do texto, que é responsável por tornar o texto coerente.

Neste contexto, o caminho mais natural para a representação da coesão de um texto, considerando os propósitos computacionais, é o de representar um texto como um grafo [Mani 01]. Neste grafo os nós são os elementos textuais – geralmente os substantivos, adjetivos e verbos presentes no textos – e os arcos são as ligações entre os elementos, representando relações semânticas entre os mesmos. A idéia básica de representação do texto em termos de um grafo é que a topologia do grafo revela algo de interessante sobre a estrutura da informação presente no texto.

Para verificar o tipo de relação existente entre os elementos do texto, normalmente utiliza-se o *Wordnet*, que é um dicionário semântico para a língua inglesa contendo elementos tais como substantivos, verbos, adjetivos e advérbios. Os substantivos são representados na forma de uma cadeia de conceitos. No *Wordnet* dado um substantivo como entrada, podem ser obtidos outros substantivos ligados àquele da entrada pelas relações de hiperônimo (generalização), hipônimo (especialização) e outras, como será visto adiante (ver seção 2.6).

O objetivo do trabalho é melhorar a sumarização utilizando como ferramenta um grafo de relacionamento entre os substantivos existentes em um texto, com o auxílio do dicionário semântico *Wordnet*, de forma que seja possível extrair as relações semânticas – tais como hipônimos e hiperônimos – entre os mesmos. A partir do grafo de relacionamentos gerado o mesmo será utilizado

para a obtenção de diversas características semânticas entre os elementos textuais. Em seguida realizar-se-á, como aplicação, o uso das características obtidas num sistema para a sumarização automática de textos fundamentado em um algoritmo de aprendizagem de máquina.

Nesta aplicação e sumarização automática, estende-se o trabalho de [Larocca 02]. Os experimentos realizados neste trabalho utilizam as características empregadas nos dois sistemas de [Larocca 02], acrescentando-se outras duas características extraídas dos grafos gerados a partir do relacionamento de hipônimos e hiperônimos entre os substantivos componentes das sentenças, extraído do *WordNet*.

O restante deste trabalho está organizado em 4 capítulos. O Capítulo 2 apresenta uma revisão bibliográfica dos trabalhos relacionados à tarefa de sumarização automática de textos. O Capítulo 3 descreve a geração do grafo de relacionamento entre os elementos textuais, a extração das características do mesmo, e a abordagem proposta para a aplicação no problema de sumarização automática de textos. Já o Capítulo 4 apresenta os experimentos realizados, bem como os resultados obtidos. Por último são apresentadas as conclusões e perspectivas do trabalho no Capítulo 5.

# 2. REVISÃO BIBLIOGRÁFICA

Neste capítulo são apresentados os seguintes conceitos:

- A tarefa clássica de recuperação de informações, com ênfase na apresentação dos métodos utilizados para o pré-processamento dos textos, e no modelo de representação vetorial empregado para os textos;

- A tarefa da sumarização automática de textos, com a apresentação dos principais métodos utilizados, desde o mais clássico até os mais sofisticados, que utilizam técnicas de aprendizagem de máquina.

## 2.1. Recuperação de Informações

Nos últimos 20 anos a área de pesquisa em recuperação de informações vem crescendo vertiginosamente, com o objetivo de efetuar a indexação de texto e a busca por documentos úteis em uma coleção [Baeza-Yates 99]. Atualmente, a pesquisa inclui os seguintes tópicos: modelos para a representação de textos, classificação e categorização de documentos, arquitetura de sistemas, interface com o usuário, visualização de dados, filtragem, linguagens, etc.

Vários fatores, incluindo o surgimento dos softwares para o processamento de textos que gerou a expansão de textos de forma eletrônica, motivaram o surgimento de técnicas de busca de informações em textos complexos. De uma forma geral os sistemas de recuperação de informação devem de algum modo “interpretar” os conteúdos de informação que aparecem em uma coleção de documentos e classificá-los por ordem de relevância, a partir de uma consulta do usuário. Esta “interpretação” do conteúdo de um documento envolve a extração de informações sintáticas e semânticas do texto. Pode-se dizer que a maior dificuldade do sistema de recuperação de informação é não somente extrair desta informação em si, mas a decisão sobre a relevância do documento em

relação à consulta do usuário. Portanto, a noção de relevância é o centro dos sistemas de recuperação de informação [Baeza-Yates 99].

### 2.1.1 O Modelo Vetorial

Como modelo formal para a representação de textos utiliza-se freqüentemente a representação vetorial, proposta inicialmente por [Salton 88]: os documentos são considerados como vetores multi-dimensionais, onde cada dimensão do vetor representa um radical (*stem*) ou termo, e seu valor é a freqüência de ocorrência de um termo no documento.

No modelo vetorial a avaliação da medida de similaridade entre um documento *dj* e uma consulta *q* é feita pela correlação entre os vetores que os representam, quantificada pelo coseno do ângulo formado por *dj* e *q*. Esta métrica é conhecida como medida de similaridade do coseno. De forma grosseira, quanto menor o ângulo entre os dois vetores mais similares são os documentos. Se X e Y são dois vetores n-dimensionais, o ângulo entre os dois satisfaz:

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

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

$$\cos\theta = \frac{X \cdot Y}{|X||Y|} = \frac{\sum_{i=1}^{n} x_i y_i}{\left( \sum_{i=1}^{n} x_i^2 \right)^{1/2} \left( \sum_{i=1}^{n} y_1^2 \right)^{1/2}}$$

Os valores de cos θ variam de 1 para documentos com máxima similaridade até 0 para documentos sem nenhuma similaridade.

Esta métrica é muito utilizada em recuperação de informação onde a consulta é comparada com cada um dos documentos pertencentes a uma base, e os resultados são ordenados de acordo com a medida de similaridade do cosseno.

Para saber quais termos do documento são mais relevantes ou menos relevantes, Salton propôs várias técnicas para calcular seus pesos [Salton 88]. No modelo vetorial os pesos mais utilizados para relacionar um termo i em um documento d são o TF(i,d) e o TF-IDF(i,d).

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

Para o cálculo do TF-IDF(i,d) são necessários outros elementos: o DF(i) (*document frequency*) é o número de documentos no qual o termo i aparece ao menos uma vez; o IDF(i) (*inverse document frequency*) pode ser calculado a partir do 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 [Larocca 02].

Finalmente o valor do TF-IDF(i,d), que corresponde à dimensão i do vetor d é então calculado através da seguinte fórmula:

TF-IDF (i,d) = TF(i,d) * IDF(i)

Portanto, um termo que ocorre freqüentemente em um documento é considerado importante (TF alto), e um termo que é muito freqüente na coleção de documentos é considerada pouco importante (IDF baixo).

Em [Larocca 00a] é apresentada uma proposta para a utilização de uma medida similar ao TF-IDF na tarefa de sumarização de documentos. Na tarefa de sumarização cada sentença é representada como um vetor de pesos, e os valores destes são calculados pela métrica TS-ISF (*term frequency – inverse sentence frequency*). A computação do TF-ISF para cada palavra é similar à computação do TF-IDF para documentos [Salton88]. A diferença é que a noção de "documento" do TF-IDF é substituída pela noção de sentença no TS-ISF, e analogamente o "número de documentos" é substituído pelo número de sentenças no documento.

### 2.1.2 Pré-processamento

Existem várias técnicas na área de recuperação de informações que são utilizadas para realizar pré-processamento e transformar um documento em uma representação vetorial [Baeza-Yates 99]. Como um documento apresenta um grande número de palavras únicas, são aplicados métodos para reduzir a dimensionalidade.

Entre eles destacam-se os seguintes procedimentos:

- *Case Folding*: é a substituição dos caracteres para o mesmo formato, ou seja, as palavras que estejam escritas em caixa alta, caixa baixa e somente a primeira maiúscula, ficarem padronizadas no mesmo formato.

- *Stopwords*: é a eliminação de palavras como artigos, preposições e conjunções, podendo também ser incluídos no conjunto de *stopwords* alguns verbos, advérbios, adjetivos e outras palavras que não devem ser consideradas como de conteúdo semântico.

- *Stemming*: é a eliminação dos prefixos e sufixos das palavras ficando somente o radical. Isto permite que elementos

textuais de semântica similar, tais como “correr”, “correndo”, “corri” e “corrida” sejam reduzidos ao mesmo radical (*stem*) comum “corr”. Para tal tarefa o algoritmo mais utilizado é o algoritmo de Porter [Porter 80], que requer conhecimento detalhado de lingüística da língua em que o texto foi escrito.

A avaliação de sistema de recuperação de informações geralmente utiliza duas métricas como unidades de medida de taxa de acerto [Barzilay 97], [Marcu 99]:

- *Precisão*: é a proporção de sentenças corretas que serão selecionadas pelo sistema;

- *Cobertura*: é a proporção das respostas corretas que o sistema selecionou com relação a todas as sentenças que deveriam ser consideradas corretas.

## 2.2. Aprendizagem de Máquina e Sumarização de Textos

O Aprendizado de Máquina (AM) é uma subárea da Inteligência Artificial que pesquisa métodos computacionais relacionados à aquisição de novos conhecimentos, novas habilidades e novas formas de organizar o conhecimento já existente [Mitchell 97]. Mitchell define a AM como “qualquer programa de computador que aumenta sua performance de uma tarefa através da experiência”.

Técnicas de AM têm sido muito usadas em todos os ramos da computação, por exemplo, reconhecimento de imagens, sistemas baseados em conhecimento, roteamento de redes e processamento de textos, conseguindo resultados satisfatórios e, às vezes, até melhores do que o esperado.

As técnicas de AM são classicamente divididas em técnicas de aprendizado supervisionado e não supervisionado. No aprendizado supervisionado, o conjunto de dados do qual se pretende extrair conhecimento já

vem todo rotulado, isto é, a cada instância está associada sua classe, a que o algoritmo de AM deve aprender a definir. No aprendizado não supervisionado, o conjunto de dados não vem rotulado, sendo o algoritmo de AM incumbido de tentar agrupar os dados de acordo com suas características da melhor maneira possível, formando o que se chama de *clustering*.

As técnicas de AM podem ainda ser classificadas de acordo com o paradigma que seguem, que pode ser simbólico, estatístico, neural ou genético. O aprendizado simbólico se caracteriza por extrair conhecimento que seja acessível e interpretável por seres humanos; o aprendizado estatístico trabalha com fórmulas estatísticas e probabilidades; o aprendizado neural consiste, principalmente, no uso de redes neurais para classificação; o aprendizado genético, por fim, engloba os algoritmos genéticos e suas aplicações.

O processo de sumarização baseado em aprendizagem de máquina envolve alguns conceitos que serão esclarecidos logo a seguir.

### 2.2.1 O Problema de Classificação

Um sistema de classificação é utilizado para prever a classe de um objeto baseado em seus atributos, se enquadrando como um procedimento de AM supervisionado.

Os dados utilizados para resolução desse tipo de tarefa consistem em um conjunto de atributos denominados previsores e um atributo denominado meta, que define a classe a que esse registro pertence. O objetivo dessa tarefa é descobrir um relacionamento entre os atributos previsores e o atributo meta, usando registros cuja classe é conhecida, para que posteriormente esses atributos previsores possam ser utilizados para prever a classe de um registro cuja classe é desconhecida [Hand 97].

Quando se trabalha na avaliação de um classificador, os exemplos disponíveis para criação de um modelo de classificação são divididos em dois conjuntos mutuamente exclusivos: um conjunto de treinamento e um conjunto de teste. O conjunto de treinamento fica disponível para o classificador, que analisa as relações entre os atributos previsores e o atributo meta. Os relacionamentos descobertos, a partir desses exemplos, são então utilizados para prever a classe dos registros presentes no conjunto de teste. Para o classificador, o atributo meta do conjunto de teste fica indisponível. Após prever a classe dos exemplos do conjunto de teste, as classes previstas são então comparadas com as classes reais dos exemplos, definidas pelo atributo meta. Se a classe prevista for igual a real, a previsão foi correta, caso contrário, a previsão foi incorreta.

Um dos principais objetivos na tarefa de classificação é maximizar a taxa de classificações corretas nos dados de teste, que corresponde à razão entre o número de exemplos corretamente classificados e o número total de exemplos disponíveis no conjunto de testes.

O conhecimento descoberto pelo classificador, através dos exemplos de treinamento, pode ser representado de várias formas. Neste trabalho, o interesse está voltado para o conhecimento representado através do algoritmo Naive-Bayes [Mitchell 97].

### 2.2.2 Aprendizagem de Máquina e o Naive-Bayes

O algoritmo de aprendizagem de máquina é um classificador que informa a um dado conjunto a qual classe pertence. No caso do algoritmo de aprendizagem de máquina Naive-Bayes, uma abordagem probabilística de inferência é utilizada.

O algoritmo *Naive Bayes* é baseado na abordagem Bayesiana, projetando um classificador com base nas probabilidades incondicionais do atributo-meta a partir do conjunto de treinamento. A entrada desse algoritmo consiste de um conjunto de dados no formato atributo/valor [Mitchell 97]. O classificador *Naive*

*Bayes* se baseia na suposição simplificada de que os vários atributos dos exemplos de entrada são condicionalmente independentes, dado o valor final da função de saída.

Assim, esse classificador considera que a probabilidade de ocorrência de uma conjunção de atributos em um dado exemplo é igual ao produto das probabilidades de ocorrência de cada atributo isoladamente. Assumir a independência é claramente incorreto e produz uma probabilidade incorreta dos membros. Mesmo sabendo que ao assumir essa independência o *Naive Bayes* produz uma estimativa de probabilidade imprecisa, é ainda possível classificar exemplos de teste usando *Naive Bayes* com uma alta precisão.

### 2.2.3 Sumarização como classificação

A sumarização pode ser vista como um problema de classificação, onde uma sentença do texto pode pertencer a uma de 2 classes: *pertencente* e *não-pertencente* ao sumário. O algoritmo de aprendizagem de máquina deve definir quais sentenças irão pertencer a cada uma das classes. Para tal tarefa são seguidas as seguintes etapas:

- Identificação das sentenças do texto original;
- Associação de cada sentença a um vetor de características previsores, cujos valores são obtidos diretamente do conteúdo da própria sentença;
- Para o conjunto de treinamento associação de cada sentença a cada uma das seguintes classes: *pertencente* ao sumário ou *não-pertencente* ao sumário.

Como é comum na tarefa da classificação, o objetivo do algoritmo é descobrir a partir dos dados, qual o relacionamento que prevê corretamente o

valor de cada classe baseado nos valores das características previsoras daquela sentença.

## 2.3. Sistemas para a sumarização automática de textos

O primeiro trabalho publicado sobre sumarização automática de textos foi o de Luhn [Luhn 58] que descreve uma técnica estatística simples, utilizando a freqüência das palavras contidas no texto e sua posição na sentença como elementos para formar o sumário.

O algoritmo de Luhn primeiro faz um pré-processamento no texto, filtrando os termos no documento usando uma lista de *stopwords*. Em seguida, se faz o cálculo de similaridade entre as palavras, baseado no número de letras diferentes entre elas. Caso o número de letras fosse menor do que 6, as palavras eram consideradas iguais. Em seguida são procurados conjuntos que continham palavras significantes para cada sentença, sendo que cada sentença era dividida em segmentos de não mais do que 4 palavras, e cada segmento era contado considerando-se o quadrado do número de palavras significantes do agrupamento dividido pelo número total de palavras agrupadas. As sentenças eram classificadas pelos maiores valores de importância e selecionadas de acordo com um ponto de corte de relevância.

Luhn descreve várias possíveis extensões do algoritmo básico, variando o comprimento do resumo e dando um valor às palavras de uma lista de domínio específico. Ele também menciona a possibilidade de aplicar o algoritmo para outras línguas e sugere o uso dessas técnicas para gerar termos de indexação para recuperação de informação.

Como desvantagem, esta técnica não considera a semântica do texto e várias de suas soluções foram substituídas pelo uso de *stemming* e o uso de freqüência de palavras em vários documentos.

Edmundson (1969) criou programas para pesos das sentenças baseados em 4 métodos [Edmundson 69]:

- *Cue Phrase*: a relevância da sentença é baseada na presença de palavras indicadoras de relevância, como “significante”, “impossível” e “difícil”.

- *Keyword*: palavras relevantes com alta freqüência são úteis para determinar a relevância da sentença do sumário.

- *Location*: sentenças que ocorrem em certas seções do documento ou que não ocorrem no começo ou no fim do documento ou do parágrafo poder ser mais ou menos relevantes para constituir o sumário.

- *Title*: a relevância de uma sentença está baseada na presença de palavras do título ou nome de seções do documento.

Edmundson avaliou cada um dos programas ajustando os pesos manualmente, dividindo seu conjunto de artigos em conjunto de treinamento e teste. Na fase de treinamento, foi usado *feedback* de avaliações para reajustar os pesos usados por cada um dos programas, que foi então testado e avaliado nos dados de teste. Como resultado foram encontradas três características, denominadas “freqüência de medida das palavras”, que foram utilizadas na criação do melhor resumo.

Em [Barzilay 97] a técnica de sumarização utilizada foi a do uso das cadeias léxicas, que são, por definição, as seqüências de palavras relacionadas que indicam tópicos conectados no texto, ou seja, um tipo de coesão.

O trabalho de Barzilay propôs a utilização do *Wordnet*, uma rede semântica de representação do conhecimento contendo relações de sinônimos, hipônimos e hiperônimos entre outras, além de conter mais de 118.000 formas de palavras

diferentes. Neste contexto, cada conjunto de palavras relacionadas semanticamente através de relações de sinônimos é denominado *Synset*.

Antes da criação da cadeia léxica, é necessário que o texto seja segmentado; para tanto aplicou-se o algoritmo *TextTiling* [Hearst 93], que permite a divisão de um texto nos vários segmentos que o compõem. Também é feita a extração dos substantivos simples e compostos através do algoritmo *Part-of-Speech* (rotulador sintático) [Brill 92] . As cadeias léxicas são criadas através de relações divididas nas seguintes categorias:

- extra-forte: repetições da mesma palavra;
- forte: 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 uma distância máxima para que a palavra seja considerada pertencente à mesma cadeia:

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

Em [Barzilay 97] se apresenta o seguinte exemplo para representação da cadeia léxica:

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

A primeira palavra é *Mr*., que segundo o WordNet apresenta somente um sentido. 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 no WordNet:

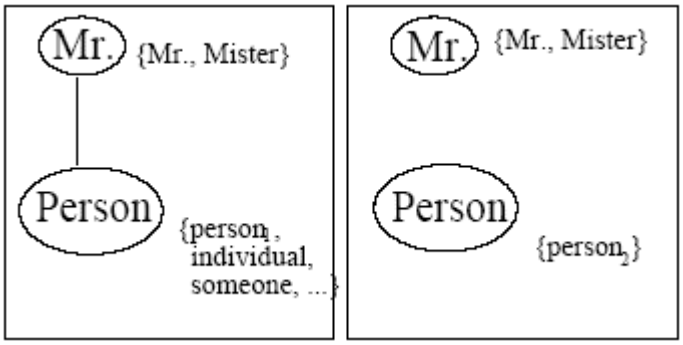


Figura 1: Cadeia Léxica 1 / Passo 1

A próxima palavra é *machine*, que tem 5 sentidos, sendo que o primeiro sentido "*an efficient person*", é relacionado aos sentidos de *person* e *Mr*, embora possa não ser o sentido correto para a sentença.

Para continuar o processo, são inseridas as palavras "*micro-computer*", "*device*" and "*pump*", e o número de alternativas aumenta. As interpretações mais fortes são indicadas na Figura 2.

Considerando o princípio de que o texto é coeso, define-se que a melhor interpretação é a que apresenta maior número de conexões. Neste caso, a segunda interpretação é selecionada, pois determina o sentido correto para a palavra *machine*. Neste caso, o valor de uma cadeia é determinado pelo número e peso das relações entre os 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 eliminadas.

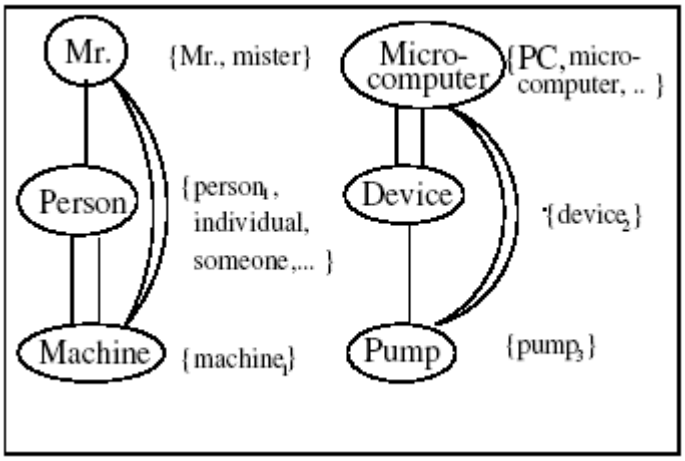


Figura 2: Cadeia Léxica Passo 2/ Interpretação 1

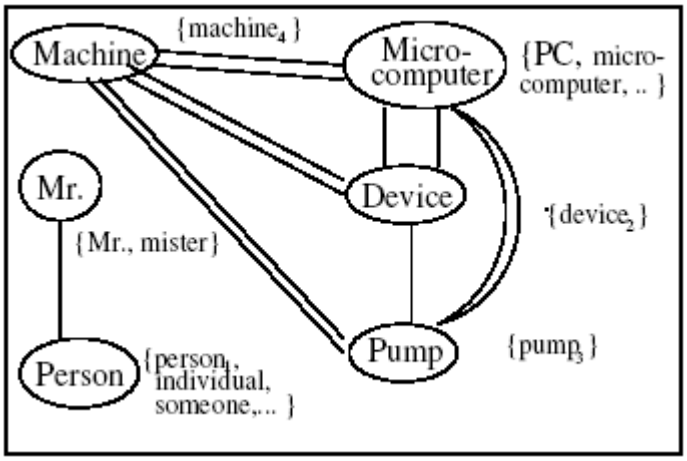


Figura 3: Cadeia Léxica Passo 2/ Interpretação 2

As cadeias são computadas em separado, depois podem ser misturadas dentro de um mesmo segmento; isto ocorre 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 a geração dos sumários através das cadeias léxicas é utilizado um algoritmo em 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:

- o Tamanho: número de ocorrência de membros da cadeia.
- o Índice de Homogeneidade: número de ocorrências distintas dos membros da cadeia dividido pelo tamanho da cadeia.

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

Força (cadeia) = tamanho x índice de homogeneidade

As cadeias mais fortes são:

Força (cadeia) > Média (Força) + 2 * Desvio Padrão (Força)

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

- ✓ Passo 2: Extrair sentenças mais significantes:

Foram propostas 3 heurísticas:

- o Heurística 1: Para cada cadeia forte, escolher a sentença que aparece primeiro em um membro da cadeia do texto.
- o Heurística 2: Para cada cadeia forte, escolher a sentença que aparece primeiro em um membro representativo da cadeia do texto. Membros representativos da cadeia são palavras que apresentam freqüência na cadeia acima da média.
- o Heurística 3: Para cada cadeia forte, encontrar a unidade do texto onde a cadeia está mais concentrada e extrair a sentença com a primeira aparição da cadeia na unidade. A concentração da cadeia na unidade 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 heurística 3 apesar de mais elaborada, apresentou resultados piores.

Barzilay obteve bons resultados comparando o algoritmo de cadeias léxicas com a ferramenta de sumarização do Microsoft Office 97 – Microsoft Word Summarizer. Foi utilizada uma base com 40 documentos, sendo que para cada documento foram extraídos 10 sumários feitos por 5 juízes humanos, onde cada juiz produziu 2 sumários com 10% e 20% do número de sentenças.

Os resultados medidos em termos de precisão e cobertura foram:

- Sumários 10%: Word Summarizer com precisão igual a 33% e cobertura 37% e para as Cadeias Léxicas a precisão igual a 61% e a cobertura igual a 67%.

- Sumários 20%: Word Summarizer com precisão igual a 32% e Cobertura 39% e para as Cadeias Léxicas a precisão igual a 47% e a cobertura igual a 64%.

Uma proposta para sumarização utilizando coesão foi a de [Mitra 97], em que ele apresenta a sumarização baseada em parágrafos como a unidade de extração, ou seja, o parágrafo possui mais contexto do que as sentenças.

A relação entre os parágrafos é determinada através de um mapa de relacionamentos do texto, sendo que parágrafos são associados aos nós de um grafo e relacionados por arcos, que se baseiam na similaridade numérica entre cada par de parágrafos. Este 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. Também o mapa pode ser utilizado na geração de sumários, através da identificação de parágrafos importantes.

No trabalho de Mitra são sugeridos quatro caminhos no mapa que selecionam frases para o sumário:

- *Global Bushy path*: que seleciona os *n* parágrafos mais conectados no mapa, onde *n* é o número de parágrafos desejados no sumário.

- *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. Este caminho pode minimizar o problema que o *Global Bushy path* possui, onde os parágrafos selecionados são altamente conectados a outros parágrafos, mas não necessariamente entre si o que pode gerar sumários incoerentes e com má legibilidade.

- *Segmented Bushy Path*: este caminho constrói *Global Bushy paths* separados para cada segmento do texto e concatena os parágrafos selecionados na ordem do texto. No mínimo um parágrafo é selecionado para cada segmento, o restante do resumo é formado selecionando-se os nós mais altamente conectados de cada segmento na proporção do seu tamanho.

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

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 com o *Global Bushy Path*.

As propostas apresentadas a seguir são técnicas utilizadas para computar a coerência no texto. A primeira delas é a de [Marcu 99], em que ele propõe um sistema que utiliza uma árvore da estrutura retórica do texto, isto é, uma árvore

binária onde cada folha é um núcleo (expressa o que é essencial na argumentação do texto) ou satélites (informações detalhadas, que visam convencer o leitor de uma afirmação). Os melhores resultados obtidos pelo sistema foram precisão igual 65,51 % e *cobertura* igual a 67,85 %.

Os experimentos realizados por Marcu confirmaram 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.

[Teufel 99] propôs uma técnica para fazer a sumarização de textos longos, como artigos de revistas, com 20 ou mais páginas. Para essa tarefa também foi utilizada 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.

Ou seja, o objetivo foi o de extrair sentenças para composição do sumário, tentando separar as sentenças que capturam regras retóricas das sentenças irrelevantes, que são a maior parte do texto, gerando um sumário intermediário e identificando a regra retórica correta de cada sentença candidata em uma das sete unidades argumentativas propostas: *Background, Topic/Aboutness, Related Work, Purpose/Problem, Solution/Method, Result, Conclusion/Claim*. Todos os artigos técnicos utilizados para a avaliação do sistema seguem a estrutura argumentativa citada anteriormente.

Para o treinamento do sistema, foram utilizadas características similares ao de [Kupiec 95]:

- *Indicator Quality*: indica meta-comentários do texto.
- *Indicator Rhetorics*: modela a contribuição retórica das frases.

- *Header Type*: representa a divisão retórica da sentença, especificando a divisão na qual a sentença aparece no texto (“Introdução”, “Conclusão”, etc).

O método foi testado com uma base de sumários extrativos. Analisando os resultados na geração de um sumário intermediário, a melhor característica testada individualmente, a qual obteve 54,4% de taxa de acerto foi o *Indicator Quality.* O melhor resultado no geral foi a combinação de todas as características, excluindo o *Indicator Rhetorics*, com taxa de acerto de 66%. Já na identificação da regra retórica correta de cada sentença candidata, o melhor resultado foi a combinação do *Indicator Rethorics, Location* e *Title* com 64,2% de taxa de acerto. A base de comparação utilizada foi a seleção da regra retórica com maior ocorrência em todas as sentenças, onde a taxa de acerto foi de 40%.

Em [Kupiec 95], é apresentada uma abordagem para sumarização como um problema estatístico de classificação. Dado um conjunto de treinamento de documentos, com documentos selecionados manualmente, o sistema obtém uma função de classificação que estima a probabilidade de uma dada sentença ser incluída no resumo.

Neste caso, foram utilizadas sete características para obter a função de classificação:

- *Sentence Length Cut-off*: sentenças curtas tendem a não ser incluídas no sumário. Para um dado limiar, 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”, “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*: é uma característica verdadeira para sentenças presentes nos 10 primeiros parágrafos ou nos 5 últimos parágrafos do documento. Sentenças em um parágrafo são distinguidas de acordo com sua ocorrência 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 é selecionado e cada sentença é classificada em função da freqüência das referidas palavras. A característica é binária, sendo verdadeira para as sentenças que apresentam um maior número de palavras temáticas.

- *Uppercase Word*: Pressupõe que palavras em maiúscula 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 para cada sentença e abreviaturas comuns (Kg, F,...). A característica é binária, sendo verdadeira para as sentenças que apresentam maior número de palavras em maiúscula.

Para tal tarefa de classificação, foi utilizado o classificador Naive-Bayes, que faz o cálculo da probabilidade de uma sentença ser incluída no sumário. O melhor resultado foi encontrado, com uma taxa de acerto de 42 % para as sentenças selecionadas pelo sistema, utilizando-se uma combinação das características *Paragraph, Fixed-Phrased* e *Sentence Length Cut-off*.

Em [Mani 98a] é proposto um método de sumarização baseado em aprendizagem de máquina para um conjunto de documentos contendo resumos fornecidos pelos autores. Foram utilizadas 3 classes de características:

- Locacionais: que exploram a estrutura do texto
  - 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 um valor variando de 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 nomes próprios.
- Coesão: envolvem relações entre palavras, indicando o quão conectado é o texto.

- 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.

Para tal tarefa foram usados os algoritmos de treinamento:

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

- C4.5: que produz regras a partir da árvore de decisão produzida pelo C4.5 [Quinlan 93].

- AQ15c: indutor de regras que otimiza as regras de acordo não apenas com sua precisão preditiva, mas também de acordo com a simplicidade (número de condições) das regras.

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

Em [Larocca 02] são propostos 2 sistemas de sumarização utilizando aprendizagem de máquina.

No primeiro sistema baseado na técnica de [Mani 98a] são produzidos sumários “ideais” com 10% das sentenças do texto a partir do sumário fornecido pelo autor. Foram extraídas 7 características, sendo elas:

- Posição da Sentença: utiliza técnica similar à de [Nevill-Manning 99], indica a posição em que a sentença está no texto, onde o valor é normalizado na escala de 0 a 1.

- Tamanho da Sentença: valor que é normalizado pelo tamanho da maior sentença do texto.

- TFISF Médio: onde o TF-ISF representa o valor de cada palavra na representação vetorial dos documentos, indicando a importância das palavras no documento [Larocca 00a].

- Semelhança com o Título: as sentenças do texto são comparadas com o título, onde tanto as sentenças como também o título são transformados para representação vetorial, utilizando para tal comparação a similaridade dos co-senos [Salton 88].

- Semelhança com Palavras Temáticas: em [Turney 00] é proposto um programa para a extração de palavras-chaves do texto. Foram extraídas 15 palavras-chaves utilizando a *API* de programação do software *Extractor*. Esta característica é empregada no sistema.

- Conectividade da Sentença: Para cada sentença do texto é realizada uma consulta contra todas as outras sentenças do texto, depois são somados os valores de similaridade para todas as sentenças e normalizar pelo valor da maior soma [Mitra 97].

- Semelhança com o Centróide: é calculado o valor do centróide do texto pelo vetor médio de todas as sentenças incluindo o título; em seguida se calcula a similaridade entre este centróide e todas as sentenças do texto, fazendo-se uma normalização para o intervalo [0, 1].

As características são discretizadas em 3 intervalos de largura constante: alto, médio e baixo. Para se fazer a classificação, emprega-se o classificador Naive-Bayes, que realiza o cálculo de probabilidade das sentenças fazerem parte do sumário [Kupiec 95] e [Teufel 99].

No segundo sistema utilizando treinamento proposto em [Larocca 02] foram utilizadas outras 7 características baseadas no algoritmo de *clustering* aglomerativo [Yarri 97]. Neste processo, o texto é processado pelo algoritmo de *clustering* aglomerativo, onde cada sentença de saída é classificada como relevante (possui as idéias principais do texto) ou de fundo (que possui informação não essencial):

- Indicador de conceitos principais: indica se a sentença possui ou não os conceitos principais do texto. Considerando que os substantivos são as palavras que possuem maior relevância, os mesmos são extraídos do texto utilizando o software *part-of-speech* [Brill 92], removendo os substantivos repetidos. Para cada substantivo é calculado o número de sentenças em que o termo aparece, sendo que os 15 termos mais freqüentes são selecionados.

- Ocorrência de nomes próprios: os nomes próprios são identificados pelo software *part of-speech* [Brill 92], representam dicas importantes especialmente em texto de notícia.

- Ocorrência de anáforas: são detectadas de forma similar a [Strzalkowski 98] onde são identificadas no início da sentença, as 6 primeiras palavras. Indica uma informação adicional ao texto e não essencial.

- Ocorrência de marcadores de discurso no início da sentença: é verificada a existência de marcadores de discurso como "*because*", "*furthermore*" e "*additionaly*" e como as anáforas são informação adicional e não essencial ao texto.

- Conectividade das sentenças: é uma característica utilizada no primeiro sistema citado acima, onde as sentenças que não são essenciais ao sumário possuem baixa coesão.

- Profundidade da sentença na árvore: representa a profundidade da sentença na árvore gerada pelo algoritmo de *clustering* aglomerativo.

- Posição na árvore: considera o caminho da raiz de árvore produzida pelo algoritmo de *clustering* aglomerativo até a sentença selecionada, onde são consideradas as profundidades de até 4 níveis.

O sistema é treinado com dois classificadores: C4.5 [Quinlan 93] e o Naive-Bayes.

Nos experimentos de [Larroca 01] foram utilizados “sumários ideais” de duas maneiras:

- Sumários ideais automáticos

Foram obtidos a partir da proposta de [Mani 98a]. Mani sugere usar o sumário provido pelo autor do documento, ou seja, uma sumário não extrativo, como consulta a cada uma das sentenças do texto, por meio do cálculo da similaridade do co-seno. As sentenças que apresentarem maior relevância (os maiores valores de similaridade com o sumário do autor) são ordenadas e aquelas de maior similaridade são utilizadas para formar os sumários, atendendo ao tamanho de 10% e 20% do texto original.

- Sumários ideais manuais

Para a obtenção de sumários manuais, utilizaram-se os serviços de uma professora de inglês que é graduada em Lingüística e leciona há vários anos. Neste caso, a professora selecionou as sentenças com alta relevância para inclusão no sumário, de forma a atender os percentuais de 10% e 20% para a compressão.

O maior valor médio de *precisão* e *cobertura* foi obtido pelo sistema com o classificador Naive-Bayes, onde para os sumários ideais automáticos com 10%

das sentenças do texto apresentou *precisão* e *cobertura* igual a 40%. Já para os sumários ideais automáticos com 20% das sentenças do texto o sistema apresentou *precisão* e *cobertura* igual a 51%. Para os sumários manuais automáticos com 10% das sentenças do texto os valores de *precisão* e *cobertura* foram 26% e os sumários manuais automáticos com 20% das sentenças do texto obtiveram *precisão* e *cobertura* igual a 38%.

## 2.4. O WordNet

O *Wordnet* é um dicionário contendo substantivos, verbos, adjetivos e advérbios para a língua inglesa, sendo que os substantivos são representados como uma rede semântica de conceitos.

Segundo [Miller 90], o *WordNet* contém aproximadamente 80.000 substantivos organizados em 60.000 conceitos léxicos. Com isso, o *WordNet* não é um dicionário convencional, pois tenta fazer relações entre sentido das palavras mais explícito e mais fácil de usar.

A relação semântica básica no *WordNet* é o sinônimo e um conjunto de sinônimos é chamado de *synset* . A maior parte dos *synsets* é acompanhada por um tipo de observação descritiva como a fornecida por dicionários convencionais. Mas um *synset* não é equivalente a uma entrada do dicionário, por ter palavras polissêmicas (palavras que possuem mais de um significado), tem várias notas diferentes e o *synset* só tem uma nota simples. Portanto um dicionário pode conter informação semântica que no *WordNet* seria distribuído em vários *synsets* diferentes.

No *Wordnet* um sinônimo é uma relação de “igualdade” entre formas de palavras, e é a relação semântica mais importante para organizar substantivos em uma relação de conceitos léxicos.

Outras relações importantes entre sentidos particulares de palavras são: (1) a generalização, que obtém as palavras chamadas de hiperônimos, e (2) a especialização que obtém as palavras chamadas de hipônimos.

Segue um exemplo explicativo destes conceitos. Seja o relacionamento:

Computer, data processor, eletronic computer, information processing system

- machine
  - device
    - instrumentality, instrumentation
      - artifact, artefact
        - object, physical object
          - entity, something

Uma busca por hipônimos (especializações) da palavra “computer” têm como resultado:

Computer, data processor, eletronic computer, information processing system

- analog computer, analogue computer
- digital computer
- node, client, guest
- number cruncher
- pari-mutuel machine, totaliser, totalizator, totalisator

server, host

A hierarquia vai dos termos mais específicos até os mais genéricos do topo da árvore hierárquica do *Wordnet*.

## 2.5. Os Sumarizadores

A seguir, na Tabela 1 segue um comparativo entre alguns sistemas de sumarização existente. Nota-se uma diferença significativa entre os resultados apresentados, ressaltando que conforme a base de documentos utilizada ocorre uma melhora nos resultados, pois há casos que são usados artigos muito específicos, além do tamanho da base.

| | | |
|---|---|---|
| Barzilay 97 | Cadeias Léxicas, utilizando uma base com 40 documentos e para cada documento foram gerados 10 sumários. | Sumários 10%:<br>Precisão = 61%<br>Cobertura = 67%<br><br>Sumários 20%:<br>Precisão = 47%<br>Cobertura = 64% |
| Mitra 97 | Geração de um mapa de relacionamentos entre os parágrafos, sendo a base utilizada foi a seção de artigos da TREC. | 20% dos parágrafos utilizados no mapa. |
| Marcu 99 | Árvore de Estrutura Retórica – Árvore Binária, cada folha é um núcleo essencial no texto, usando 5 artigos curtos da revista “Scientific American” com tamanhos de 161 a 725 palavras. | Precisão = 65,51%<br>Cobertura = 67,85% |
| Mani 98 | Sumarização baseada em Aprendizagem de Máquina, utilizando a mesma base de Marcu 99. | C4.5 Rules = 69% |
| Kupiec 95 | Sumarização como um problema estatístico – Naive Bayes, sendo a base utilizada contendo 948 sentenças. | 42% das sentenças selecionadas a pertencer ao sumário |
| Larocca 01 | 2 sistemas de sumarização utilizando aprendizagem de máquina. Base com 100 documentos para treinamento e 100 documentos para validação, 30 documentos para os sumários manuais extrativos selecionados aleatoriamente da base TIPSTER. | Precisão = 40%<br>Cobertura = 40% |

Tabela 1: Comparativo entre os sumarizadores

## 2.6. Conclusões

Neste capítulo foram apresentados diversos elementos básicos da área de Recuperação de Informação: o modelo vetorial, que considera os documentos como vetores multi-dimensionas e o pré-processamento que é a aplicação de métodos para a redução de dimensionalidade do texto.

Em seguida, foram relatadas: a Aprendizagem de Máquina e a Sumarização de Textos. A aprendizagem de máquina e o problema de classificação cujo objetivo da tarefa é descobrir a relação entre atributos previsores e atributos meta usando registros cuja classe é conhecida. No contexto, a Aprendizagem da Máquina e o Naive Bayes que utiliza abordagem probabilística de inferência e a sumarização como classificação onde considera-se que uma sentença do texto pode pertencer a 2 classes: Pertence e Não Pertence ao sumário.

Também foram apresentados sistemas de sumarização existentes, salientando os que utilizam abordagem de aprendizado de máquina e técnicas de coesão do texto assim como [Barzilay 97], bem como as características usadas em [Larocca01] que são híbridas (coesão e coerência do texto). Finalmente, o uso do Wordnet para a extração de relações semânticas existentes entre as palavras.

# 3. A ABORDAGEM PROPOSTA

De modo geral, os sumários são produzidos pela seleção das sentenças que indicam relevância em seu conteúdo. O modelo mais comum de extrair as sentenças é associar um escore a cada sentença de acordo com algumas características que possam indicar a sua possível relevância, e em seguida selecionar aquelas com maior escore para compor o sumário.

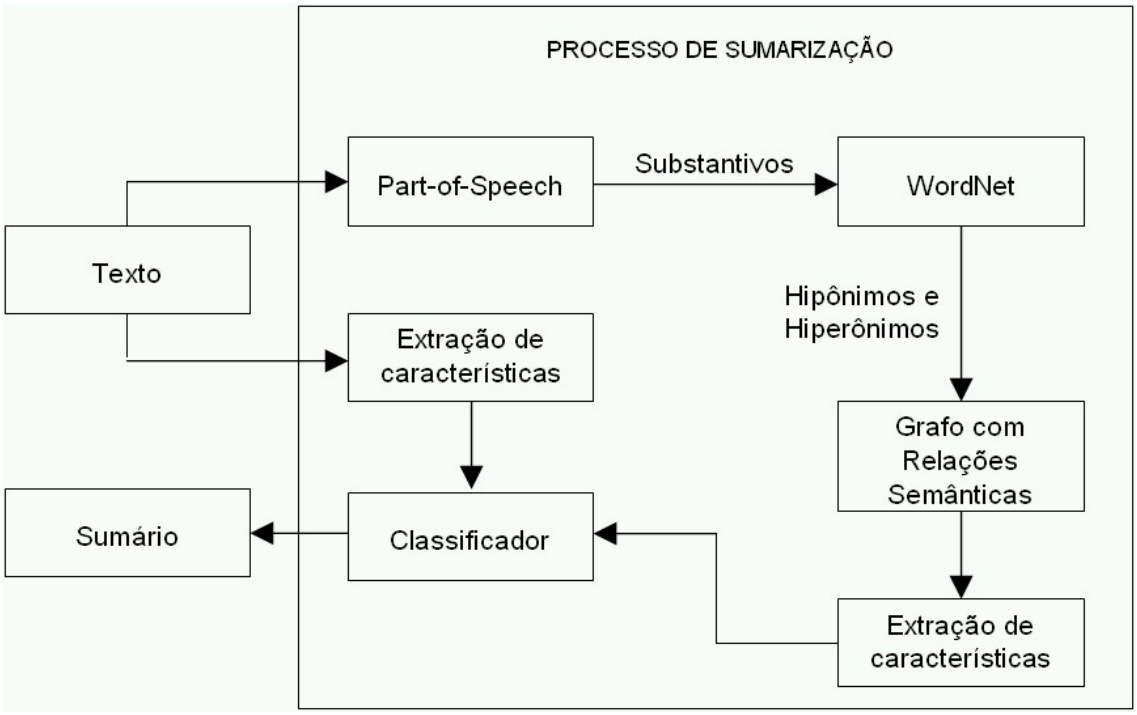


Figura 4: Visão geral do processo

Neste trabalho o escore associado a cada sentença é obtido a partir de características extraídas de um grafo semântico de relacionamento entre os substantivos que compõem o texto. Sua construção está baseada nas relações de hipônimos e hiperônimos obtidas no dicionário semântico *WordNet*.

Neste capítulo serão apresentadas os mecanismos utilizados para o pré-processamento dos textos, a geração de um grafo de relacionamentos entre os substantivos encontrados no mesmo, com base no *Wordnet,* e a extração das características deste grafo juntamente com o algoritmo de aprendizagem de máquina utilizado conforme ilustra a Figura 4 acima.

## 3.1. Pré – processamento

Nesta etapa, os textos originais são processados pelo algoritmo *part-of-speech* [Brill 92], onde as palavras são consideradas individualmente, ou seja, o algoritmo insere um marcador em cada uma das palavras do texto indicando a sua classe gramatical.

A partir desse processamento, são extraídos somente os substantivos simples e próprios do texto, pois assume-se que são as palavras que possuem maior nível significativo dentre as demais. Também são retirados os substantivos repetidos no texto.

Para as características que foram utilizadas a partir de [Larocca01], o pré-processamento foi realizado aplicando as técnicas de *stopwords*, que é a eliminação de palavras como artigos, preposições e conjunções e *stemming* que é a eliminação dos prefixos e sufixos das palavras ficando somente o radical, como estão descritas na Seção 2.1.2.

## 3.2. Geração do Grafo

Após realizado o pré-processamento, utilizando uma *API* de programação do *WordNet*, os substantivos extraídos são utilizados para uma consulta no dicionário semântico, para extração dos substantivos associados às relações de sinônimos, hipônimos (especialização), e hiperônimos (generalização).

O processo acontece da seguinte forma: é pesquisada a relação de cada substantivo com todos os outros que aparecem no texto, por meio do *WordNet*; esta pesquisa mostra se existe alguma relação de sinônimo, hipônimo ou hiperônimo entre estes elementos. Caso exista, se exibe a distância entre os substantivos em questão, considerando a hierarquia presente no dicionário semântico. Este procedimento é aplicado sucessivamente a todos os substantivos presentes no texto. A partir das relações entre os substantivos extraídos do *WordNet*, são gerados grafos de relacionamento semântico entre os substantivos do texto.

Através de matrizes de adjacências, são formados os grafos: são duas matrizes que relacionam substantivos a substantivos, sendo uma para as relações de hipônimos e outra para as relações de hiperônimos. As células das matrizes são preenchidas com o valor da distância entre os substantivos.

Em seguida apresenta-se um exemplo, onde os substantivos *Dog* e *Cat* são submetidos ao *WordNet*.

O relacionamento de hiperônimo é:

*dog -> canine -> carnivore -> feline -> cat*

Portanto, o valor da distância que existe entre as palavras é:

Dist (*dog,cat*) $= 1 - \left(\frac{2}{4}\right) = 0{,}5$, pois a palavra *carnivore* é hiperônimo de *dog* e *cat.* Neste caso não haveria hipônimo entre as 2 palavras, mas subentende-se que o hipônimo de *carnivore* são as palavras *canine* e *feline*, e que o hipônimo de *canine* é *dog* e de *feline* é *cat*, respectivamente.

De acordo com a Figura 5 de um exemplo de um texto mostrada a seguir, são extraídos os seguintes substantivos na etapa de pré-processamento: *house*, *animal*, *dog* e *Charlie*.

Título: Animals in my house

Palavra-chave: Animal

In my house there is a animal.
The one I like the most is my dog.
It's name is Charlie.
Charlie is the best dog ever.

Figura 5: Exemplo de texto

Os substantivos são submetidos ao *Wordnet*, onde são extraídas as distâncias existentes entre as palavras nas relações de hiperônimos e hipônimos, como indicado às Tabelas 2 e 3. Desta forma são obtidas as matrizes de adjacências que representam o grafo semântico de relacionamento entre os elementos textuais como mostra a Figura 6.

| | house | animal | dog | Charlie |
|---|---|---|---|---|
| house | - | 0.25 | 0.6 | 0.0 |
| animal | 0.75 | - | 1.0 | 0.0 |
| dog | 0.39 | 0.0 | - | 0.0 |
| Charlie | 0.0 | 0.0 | 0.0 | - |

Tabela 2: Distância de Hiperônimos entre substantivos

| | House | animal | dog | Charlie |
|---|---|---|---|---|
| house | - | 0.0 | 0.0 | 0.0 |
| animal | 0.0 | - | 0.0 | 0.0 |
| dog | 0.0 | 1.0 | - | 0.0 |
| Charlie | 0.0 | 0.0 | 0.0 | - |

Tabela 3: Distância de Hipônimo entre substantivos

Em paralelo ao cálculo do valor da distância dos substantivos no *WordNet*, foi calculada a freqüência com que cada substantivo aparece em cada sentença do texto, como mostra a Tabela 4.

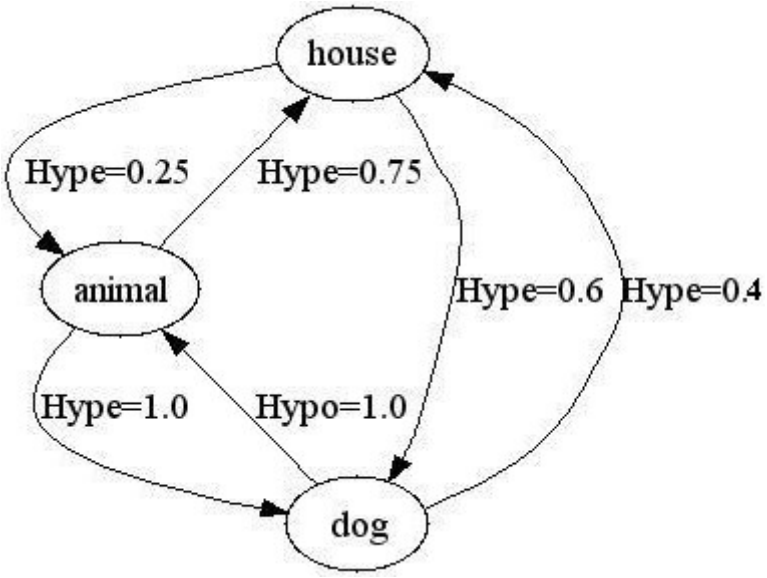


Figura 6: Grafo com relações de hipônimos e hiperônimos

Finalmente são combinadas com as distâncias existentes entre as palavras do grafo para a extração das características.

| Sentença | house | animal | Dog | Charlie |
|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 |
| 2 | 0 | 0 | 1 | 0 |
| 3 | 0 | 0 | 0 | 1 |
| 4 | 0 | 0 | 1 | 1 |

Tabela 4: Freqüência dos substantivos em cada sentença

## 3.3. Extração de Características

Para o cálculo das características extraídas por sentença, é feita uma relação entre a freqüência em que o substantivo aparece na mesma sentença com o valor da distância do substantivo no *WordNet*. Ou seja, para toda sentença e para cada substantivo encontrado na sentença multiplica-se o valor da sua freqüência pelo somatório do valor das distâncias que este substantivo têm para com os outros substantivos a ele relacionados.

A Figura 6 apresenta o pseudocódigo do procedimento que realiza a extração das características.

O cálculo da freqüência dos substantivos é realizado a partir dos dois grafos gerados (hipônimos e hiperônimos), de forma que são obtidas duas novas características por sentença do texto.

Juntamente com as características das relações semânticas extraídas, também foram combinadas as características utilizadas em [Larocca 02].

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, Conectividade das sentenças, Profundidade da sentença na árvore, Posição na árvore.

Todas estas características foram obtidas conforme anteriormente detalhado neste texto, e foram adicionadas ao sistema de classificação que implementa o sumarizador.

O sumarizador é um classificador Naive Bayes que como está descrito na Seção 2.2.2, calcula a probabilidade de uma sentença pertencer ou não pertencer ao sumário a partir das características extraídas do texto.

```
// Freqüência do substantivo na sentença (sentenças X substantivos)
// Contagem de palavras nas sentenças
VAR frequencia[][];

// Distância entre hiperônimos e hipônimos (substantivos X substantivos)
// Valor obtido em consulta no WordNet
VAR profundidadeHiperonimo[][];
VAR profundidadeHiponimo[][];

// Valor da característica do hiperônimo (sentenças X valor)
// Resultado que será calculado abaixo
VAR sentencaHiperonimo[];
VAR sentencaHiponimo[];

PARA i DE 0 A QUANTIDADE(sentenca) - 1 FAÇA

    sentencaHiperonimo[i] = 0;
    sentencaHiponimo[i] = 0;

    PARA j DE 0 A QUANTIDADE(frequencia) - 1 FAÇA

        SE frequencia[i][j] > 0 ENTÃO

            VAR somaHiper = 0;
            VAR somaHipo = 0;

            PARA k DE 0 A QUANTIDADE(substantivo) FAÇA
                somaHiper = somaHiper + profundidadeHiperonimo[j][k];
                somaHipo = somaHipo + profundidadeHiponimo[j][k];
            FIM PARA

            sentencaHiperonimo[i] += somaHiper * frequencia[i][j];
            sentencaHiponimo[i] += somaHipo * frequencia[i][j];

        FIM SE

    FIM PARA

FIM PARA
```

Figura 7: Pseudo código da extração de características do grafo

## 3.4. Conclusões

Este capítulo apresentou a abordagem utilizada no trabalho: o pré-processamento dos textos, a geração do grafo de relacionamento entre os substantivos e a extração das características combinadas às utilizadas em [Larocca 02].

No pré-processamento foi utilizado o algoritmo *part-of-speech* que idetifica a classe gramatical das palavras, neste trabalho foram utilizados os substantivos simples e os substantivos próprios. Em seguida tem-se a etapa de geração do grafo, onde os substantivos são submetidos ao wordnet para a extração das relações (hipônimos e hiperônimos) existente entre eles.

Também foi apresentada a etapa de extração das características do grafo, onde é realizado um cálculo envolvendo a distância entre as palavras no grafo e a frequência em que a palavra aparece na sentença. Estes valores foram combinados com as características utilizadas em [Larocca 02] para serem posteriormente submetidas ao classificador.

# 4. EXPERIMENTOS REALIZADOS

Neste capítulo são apresentados os experimentos realizados com o sistema de sumarização obtido.

No sumarizador foram utilizadas todas as características apresentadas no trabalho de [Larocca 02], sendo adicionadas novas características extraídas do grafo de relacionamento semântico entre substantivos gerado a partir do *Wordnet*.

São apresentadas a seguir as métricas utilizadas para se fazer a avaliação dos resultados, bem como a descrição das bases utilizadas e os experimentos realizados e os resultados obtidos.

## 4.1. Características Utilizadas

Nesta aplicação e sumarização automática, estende-se o trabalho de [Larocca 02]. Os experimentos realizados neste trabalho utilizam as características empregadas nos dois sistemas de [Larocca 02], acrescentando-se outras duas características extraídas dos grafos gerados a partir do relacionamento de hipônimos e hiperônimos entre os substantivos componentes das sentenças, extraído do *WordNet*.

Foram utilizados 2 conjuntos de características, sendo elas sintáticas (oriundas de estatísticas) e semânticas (dependem de lingüística).

Características Sintáticas:

- 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
- Conectividade das sentenças;

Características Semânticas:

- Profundidade da sentença na árvore;
- Posição na árvore;
- Relações de Hiperônimo: característica extraída do grafo conforme descrito na Seção 3.3;
- Relações de Hipônimo: característica extraída do grafo conforme descrito na Seção 3.3.

No total foram utilizadas 16 características. O algoritmo de aprendizagem de máquina utilizado foi o Naive-Bayes, pois este foi o classificador que obteve melhores resultados na literatura e no trabalho de [Larocca 02].

## 4.2. Bases de Documentos Textuais

Para a realização dos experimentos foram utilizadas as mesmas bases de documentos usadas por [Larocca 02]. Foram obtidas 3 bases dos textos da editora Ziff-Davis, da base TIPSTER [Harman94]. A base consiste de textos de revistas sobre computadores, hardwares, softwares, etc. Dentre os textos disponíveis, 33.658 contêm sumários providos pelo autor.

Como base de treinamento, são utilizados 100 documentos selecionados aleatoriamente; o tamanho médio dos documentos é de 129.5 sentenças, num total de 12.950 sentenças. Para a base de teste são utilizados sumários extrativos automáticos, gerados conforme a técnica proposta por [Mani 98a], onde o tamanho médio dos documentos é de 118.6 sentenças, num total de 11.860 sentenças.

Os documentos estão em inglês; desta forma o pré-processamento utilizado emprega ferramentas que são disponíveis somente para este idioma, como o *part-of-speech* de Brill e o *Wordnet*. Os textos analisados possuem taxas de compressão de 10% e 20% das sentenças dos textos, pois são as mais comuns usadas em experimentos na literatura.

Para a realização dos experimentos e análise dos resultados, foram utilizados os “sumários ideais” de duas maneiras: automático e manual. Desta forma os procedimentos metodológicos adotados foram os mesmos do trabalho de Larroca [Larroca 01].

No total foram analisados 230 documentos (usando taxas de compressão de 10% e 20% do texto) divididos em:

- 100 documentos para treinamento onde foram extraídos “sumários ideais” automáticos;
- 100 documentos para teste onde foram extraídos “sumários ideais” automáticos;

- 30 documentos manuais para teste onde foram extraídos “sumários ideais” manualmente por um juiz humano.

## 4.3. Avaliação dos resultados

O sistema proposto foi avaliado de acordo com as métricas *Precisão* e *Cobertura*, utilizadas na grande maioria dos trabalhos da área [Barzilay 97], [Marcu 99].

Para os sistemas de sumarização os valores de *precisão* e *cobertura* são dados por:

- *Precisão* = PV/(PV+PF), o número de sentenças que o sistema selecionou para o sumário e de fato pertencem ao sumário dividido pelo número total de sentenças que o sistema selecionou para o sumário.

- *Cobertura* = PV/(PV+NF), o número de sentenças que o sistema selecionou para o sumário e de fato pertencem ao sumário dividido pelo número total de sentenças pertencentes ao sumário.

onde,

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

PF = positivos falsos (o número de sentenças não incluídas no sumário ideal que foram incorretamente selecionados pelo sistema).

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

Para a tarefa se sumarização, tem-se que *precisão* = *cobertura*, pois o número de exemplos que o sistema seleciona como sendo “positivos” é igual ao número de exemplos que são de fato “positivos”.

## 4.4. Resultados dos Experimentos

Os resultados obtidos nos experimentos realizados, de acordo com as características e o classificador utilizado, são os que aparecem na Tabela 5, onde são mostrados os resultados para sumários automáticos com 10% e 20% das sentenças do texto. Pode-se notar que, para sumários automáticos, o melhor resultado foi com os resumos com 20% das sentenças do texto.

| Sumários | Precisão e Cobertura Média | Desvio Padrão da Precisão e Cobertura |
|---|---|---|
| **Sumários Automáticos 10%** | 31,70 | 1,87 |
| **Sumários Automáticos 20%** | 48,29 | 1,54 |

Tabela 5: Taxa de acerto dos sumários ideais automáticos

A Tabela 6 mostra os resultados para os sumários manuais, com 10% e 20% das sentenças do texto. O melhor resultado foi com o texto com 20% das sentenças do texto.

| Sumários | Precisão e Cobertura Média | Desvio Padrão da Precisão e Cobertura |
|---|---|---|
| **Sumários Manuais 10%** | 23,56 | 2,82 |
| **Sumários Manuais 20%** | 37,12 | 2,23 |

Tabela 6: Taxa de acerto dos sumários ideais manuais

O melhor resultado obtido para todos os experimentos foi o com a utilização de sumário automático, com 20% das sentenças do texto.

Em comparação com os resultados obtidos no trabalho de [Larocca 02], houve valores bastante similares, principalmente nos sumários manuais.

Entretanto pode-se notar que não houve melhora significativa dos resultados. Isto pode ser originado pelo reduzido número de características incorporadas ou pelo fato de ter ocorrido um conflito na combinação das características semânticas utilizadas.

Desta forma, acredita-se que a inclusão de mais características, extraídas diretamente do grafo de relacionamento semântico entre os elementos textuais e a análise das características mais relevantes deve aumentar a taxa de acerto da sumarização.

# 5. CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho propôs uma nova abordagem para a avaliação de um texto por meio da extração de relacionamentos semânticos entre substantivos presentes no texto.

A proposta está fundamentada na construção de um grafo que contém os relacionamentos semânticos entre os substantivos presentes no texto. Estes relacionamentos são do tipo hipônimo, hiperônimo, e foram obtidos com o auxílio do dicionário *Wordnet*.

Resumidamente, a partir de um texto-fonte, realiza-se a extração dos substantivos, são extraídas as relações semânticas entre estes elementos utilizando o dicionário semântico *WordNet* e é gerado um grafo de relacionamentos entre estes elementos, com a indicação das relações de hipônimos e hiperônimos.

A partir deste grafo são extraídas características que são empregadas em um sistema de sumarização, fundamentado no algoritmo de aprendizagem de máquina Naive Bayes. O sumarizador classifica as sentenças com maior probabilidade de pertencer ao sumário, a partir de probabilidades calculadas a partir de uma base de treinamento. Também são empregadas outras características para realizar a sumarização, extraídas do trabalho de [Larroca 01].

Os resultados obtidos não podem ser considerados satisfatórios: embora da mesma ordem que os obtidos sem a incorporação das características obtidas no grafo, não houve melhora significativa na qualidade dos sumários obtidos. Em comparação com alguns sistemas de sumarização citados no trabalho conforme a tabela 1 os resultados foram significativos, pois a base de documentos utilizadas em outros sistemas são muito específicas e com um número menor de documentos.

O melhor resultado obtido foi de precisão = cobertura = 48,29% para os sumários automáticos com 20% das sentenças do texto. Uma justificativa para este resultado é a de que apenas duas características obtidas a partir do grafo foram efetivamente utilizadas. Desta forma considera-se que o uso de um maior número de características permitirá a obtenção de um melhor desempenho.

A contribuição relevante do trabalho está na construção do grafo de relacionamento semântico entre os elementos textuais. Este procedimento pode ser ampliado facilmente para permitir o uso de outras categorias gramaticais, além dos substantivos. O grafo poderá ser empregado também para outras atividades de processamento textual, visto que sua geração é independente da tarefa a realizar sobre o texto.

Quanto ao sistema de sumarização, considera-se que novas características podem ser extraídas do grafo gerado e incorporadas ao sumarizador. Entre estas, propõe-se incluir algumas outras características, como o número de relacionamentos que um substantivo possui, verificando quais são os substantivos mais próximos dando assim um peso maior a estas palavras.

Em comparação com outros sistemas de sumarização, os resultados apresentados

Como trabalhos futuros, tem-se como prioridade a melhoria na performance da sumarização usando outras características extraídas do grafo de relacionamentos semânticos, além da utilização de outros recursos disponíveis no *WordNet*, tais como as relações de antônimos, que é a relação de oposição existente entre os substantivos e meronímia que é a relação de parte-todo entre os substantivos.

Também poderão ser usados outros classificadores como o C4.5 e o K-NN como sumarizador, além de técnicas de seleção de atributos utilizando algoritmo genético para fazer a avaliação das características que são mais relevantes para serem usadas pelo classificador.

O uso de técnicas de coesão sem a utilização da técnica de aprendizagem de máquina como a utilizada em [Mitra 97] que propõe a construção de um mapa de relacionamentos entre os parágrafos do texto.

# REFERÊNCIAS BIBLIOGRÁFICAS

[Baeza-Yates and Ribeiro-Neto 99] Baeza-Yates, R. and Ribeiro-Neto, B., editors (1999). Modern Information Retrieval. Addison Wesley Longman.

[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.

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

[Edmundon 69] Edmundson, H. P. New Methods in Automatic Extracting. *Journal of the Association for Computing Machinery* 16(2):2644-285.1969.

[Hand 97] Hand, D. J. Construtions and Assessmet of Classification Rules. Willey, Nova Iorque, 1997.

[Kupiec 96] Kupiec, R.; Pedersen, J. O.; Chen, F. A Trainable Document Summarizer. In *Proceedings of the 18h ACM-SIGIR Conference, Association of Computing Machinery, Special Interest group Information Retrieval*, 68-73. 1995.

[Larocca 00] Larocca Neto, Joel; Santos, Alexandre Denes dos; Kaestner, Celso A.; Freitas, Alex A. Document Clustering and Text Summarization. *Proccedings of 4th Int. Conf. Practical Applications of Knowledge and Data Minig (PADD-2000)*, 41-55. London: The Practical Application Company. 2000.

[Larocca 02] Larocca Neto, Joel. "*Contribuição ao Estudo de Técnicas para Sumarização Automática de Textos"*. Dissertação de Mestrado. Departamento de Computação - PPGIA, PUC-PR, 2001.

[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 *Proccedings 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.; Chzanowski, M.; Sundheim, B. *The TIPSTER SUMMAC Text Summarization*

*Evaluation.* MITRE Technical Report MTR 98W0000138. The MITRE Corporation. Oct. 1998.

[Mani 01] Mani, I. (2001). *Automatic Summarization*. Natural Language Processing. John Benjamins Publishing Company.

[Marcu 99] Marcu, D. Discourse trees are good indicators of importance in text. In I. Mani and 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 CLS Report 43*, Cognitive Science Laboratory, Princeton University.

[Mitchell 97] Mitchell, T.M. *Machine Learning*. WCB/McGraw-Hill, 1997.

[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 Summarization*. Madrid, Spain. 1997.

[Morris 92] Morris, A. H.; Kasper, G.M.; Adams, D. A. The effects and limitations of Automated Text Condensig on Reading Comprehension Performance. Information System 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.

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

# APÊNDICE

Esta seção contém um exemplo completo de sumarização em um texto da base TIPSTER [Harman 94]. O texto completo, seguido do sumário provido pelo autor, o sumário gerado automáticamente, partes do grafo gerado a partir do texto e a saída do sumário gerado com as características extraídas do grafo em conjunto com as usadas em [Larocca 02].

**Número do Documento**: ZF109-553-267

**Publicação**: PC Magazine Nov 13 1990 v9 n19 p297(56) * Full Text COPYRIGHT Ziff-Davis Publishing Co. 1990.&M.

**Título**: Dot matrix. (Hardware Review) (overview of 52 evaluations of dot matrix printers)(includes related article on Editor's Choices)

## Texto Completo

[1]It used to be that laser printers were the province of the rich or well-connected.
[2]The rest of us got along on 9- and 24-pin dot matrix printers.
[3]We put up with snailslow output, noise, and nlq type whose acronym might better have stood for: Never Letter Quality.
[4]Try as they might, impact printer makers could come close to, but never achieve, the superior output provided by lasers.
[5]We sighed, we wished, and we suffered.
[6]But a funny thing happened on our way to the dot matrix marketplace.
[7]Lasers became cheap--cheap enough to compete with 24-pin dot matrix printers.
[8]Cheap enough to make those of us who thought laser output was forever out of reach think again.
[9]ASSESSING YOUR NEEDS
[10]If you produce only correspondence, then your dot matrix days are probably over for good.
[11]For the rest of us, the manufacturers of the new or improved dot matrix printers reviewed here are banking on the continuing pervasiveness of those tasks most laser printers cannot perform.
[12]Want technology that can make carbons for you?
[13]No laser can do that, but just about all dot matrix printers can print up to three layers--some can print up to six layers.
[14]Want to print addresses on envelopes and labels, correspondence on cut-sheet stationery, and spreadsheets on perf paper without having to reload your output media?
[15]Lasers can't simultaneously provide all these options, but many of the dot matrix printers in our roundup can--and they can do it at a street price that no laser can match.

[16]This year, both Okidata and Panasonic have joined NEC in producing 24-pin printers that list for under $500.
[17]With street prices running in the $250 to $300 range, these versatile machines make a fine complement to any laser printer.
[18]These ultra-competitive prices will allow you to eat your laser-printer cake and have your dot matrix, too--giving you the best of both printing worlds without bankrupting you in the process.
[19]Now you can do high-quality correspondence and high-speed forms printing for less than the price of a laser printer sold two years ago.
[20]Even the less-expensive dot matrix printers include features not available in low-end models a year ago.
[21]In an effort to provide more options for buyers and to separate themselves from the pack, many 9-pin models, like the ALPS ASP1600 priced at $299, and the Star Micronics XR-1000 Multi Font priced at $499 ($50 color kit optional), include items like zero-clearance forms tear-off (the printer allows the last printed page to be torn off without losing a blank page to a form feed), intelligent paper-parking, front-panel menuing, and font cartridge slots.
[22]Additional emulations, color kits, and LCD readouts--like those found on AEG Olympia's NP 80SE ($499) and NP 136SE ($699), and Citizen's 200GX ($299)--are now items that are necessary in order to insure the survival of the dot matrix printer as a species.
[23]ONCE MORE INTO THE NICHE!
[24]This year, you're more likely to find surprises in the specialty dot matrix printer market.
Epson, for example, the manufacturer who long ago brought you the diminutive MX-80, has added to its line the beefier DFX-8000, a 63.9-pound, 9-pin printer that Epson rates at 1,066 cps--speed that rivals laser printers' at 11 ppm in draft mode.
[25]At $3,699 it won't beat the price of any low-cost laser, but you won't be able to find a laser printer that has this kind of forms-crunching muscle.
[26]And if $3,699 seems too steep for a dot matrix printer, you might take a look at Genicom's $2,595, 18-pin model 3840, which produces a whopping 8.4 ppm.
[27]For those interested in the high-end of high-speed output, see our sidebars on Mannesmann Tally's $5,999 645, whose output is measured in lines per minute--450 lpm, to be exact--and CIE America's $9,995 monster, the CI-1000, which gallops along at 760 lpm.
[28]Another area of burgeoning versatility for dot matrix printers is color. Last year, only 15 of the dot matrix printers that we reviewed came with or offered color as an option, and we relegated those printers to their own separate section.
[29]This year, in the face of increasing competition from laser printers, 24 dot matrix printers offer color output, and this time we have included them in our regular black-and-white-only dot matrix section.
[30]Now no longer low-quality curiosities, color capable dot matrix machines offer a vastly less expensive alternative to color page printers.
[31]Our reviewers found a lot to like in the $499 Citizen GSX-140, a 24-pin printer that, when augmented by its $59 color kit, produced outstanding color graphics with hardly any distortion in the primary or pastel colors.
[32]It may not rival color PostScript printer output, but a street price in the $285-to-$300 range makes this printer and its ilk hard to pass up where price must be taken into consideration.
[33]And if you print a lot of preliminary color drafts, at only a penny or so per page a low-cost color dot matrix printer may end up saving you a ton of money.
[34]Another little-known factor that may affect your choice: the lowly printer ribbon.
[35]Okidata, for instance, attributes the output quality of its revamped Microline 393 Plus printer ($1,499) to improved inking abilities.
[36]The company uses re-inking fabric ribbons and has managed to both decrease the incidence of smudges while at the same time increasing character darkness.
[37]Another lesser-known factor affecting graphics quality in dot matrix printers is printhead positioning technology.
[38]Okidata takes an automotive turn, and puts a rack-and-pinion drive in its Microline 390 Plus and Microline 391 Plus ($699 and $949, respectively).
[39]These alternative drive mechanisms produce exceptionally high-quality graphics because of their precise positioning of the printhead versus standard belt drives.
[40]THE END OF AN ERA

[41]Ultimately, the advances in dot-matrix technology will serve only to stave off the narrowing of a market fast being overtaken by laser printers performing traditional dot matrix tasks.
[42]Advances in hardware like a new Pentax printer that contains a continuous forms laser engine allowing tractor-feeding of single-part forms in a laser environment, and improvements in software, such as Avery's $100 LabelPro that permits easy printing of labels on laser printers, will enable lasers to encroach even further on dot matrix territory.

## Sumário do Autor

Sixty-six 9- and 24-pin dot matrix printers ranging in price from $269 to $23,699 are reviewed.&P.
Dot matrix printers can no longer compete with low-cost personal laser printers for printing correspondence, but fill an important niche because they can print multi-part forms and work simultaneously with different output media.&P.
Okidata, Panasonic and NEC now offer 24-pin printers for under $500; many 9-pin models include the ability to tear off forms with no clearance, intelligent paper parking, elaborate front panel menus and slots for font cartridges.&P;
Epson's new DFX-8000 is rated at a whopping 1,066 characters per second.&P.
Many dot matrix printers now include color capabilities when used with color ribbons.&P;
Five models are rated Editor's Choices: the Citizen GSX-140 and 200GX; the Epson DFX-8000 the Epson LQ-850 and its wide-carriage LQ-1050 version; and the NEC Pinwriter P6200 and wide-carriage Pinwriter P6300.&M.

## Sumário extrativo automático

The rest of us got along on 9- and 24-pin dot matrix printers.
Lasers became cheap--cheap enough to compete with 24-pin dot matrix printers.
Epson, for example, the manufacturer who long ago brought you the diminutive MX-80, has added to its line the beefier DFX-8000, a 63.9-pound, 9-pin printer that Epson rates at 1,066 cps--speed that rivals laser printers' at 11 ppm in draft mode.
Another area of burgeoning versatility for dot matrix printers is color. Last year, only 15 of the dot matrix printers that we reviewed came with or offered color as an option, and we relegated those printers to their own separate section.

## Sumário extraído com as características do Grafo

Another area of burgeoning versatility for dot matrix printers is color. Last year, only 15 of the dot matrix printers that we reviewed came with or offered color as an option, and we relegated those printers to their own separate section.
For the rest of us, the manufacturers of the new or improved dot matrix printers reviewed here are banking on the continuing pervasiveness of those tasks most laser printers cannot perform.
No laser can do that, but just about all dot matrix printers can print up to three layers--some can print up to six layers.
This year, in the face of increasing competition from laser printers, 24 dot matrix printers offer color output, and this time we have included them in our regular black-and-white-only dot matrix section.

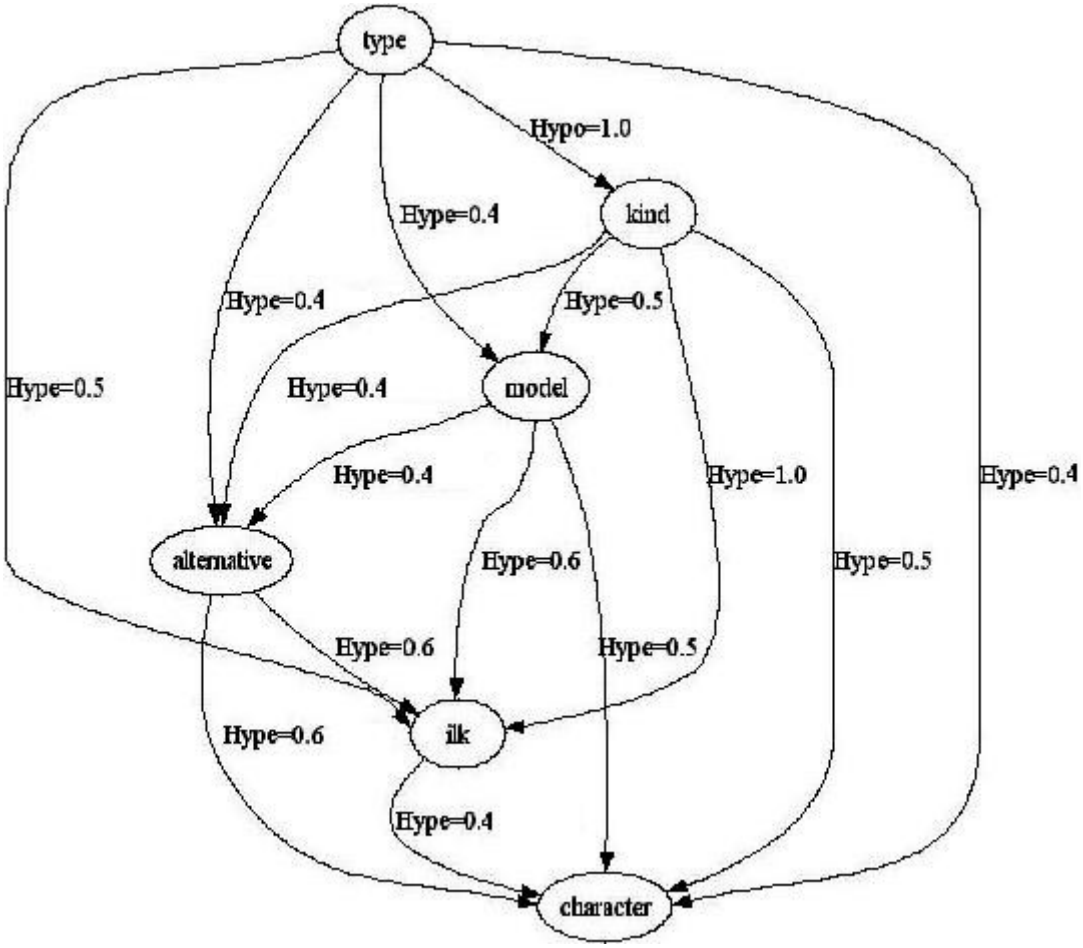


Figura 8: Parte do Grafo gerado a partir dos substantivos do texto.

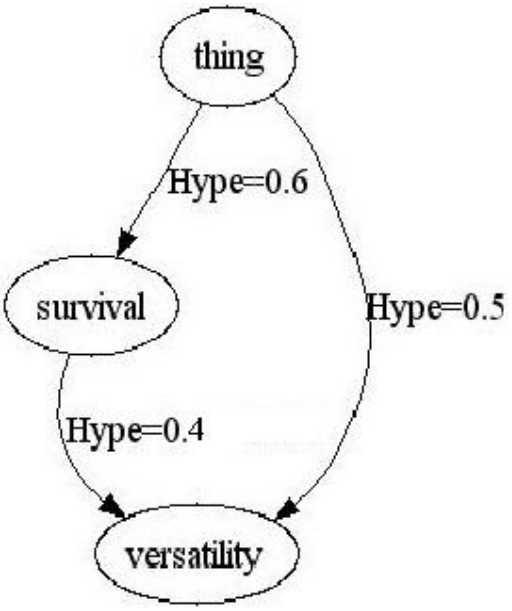


Figura 9: Parte do Grafo gerado a partir dos substantivos do texto

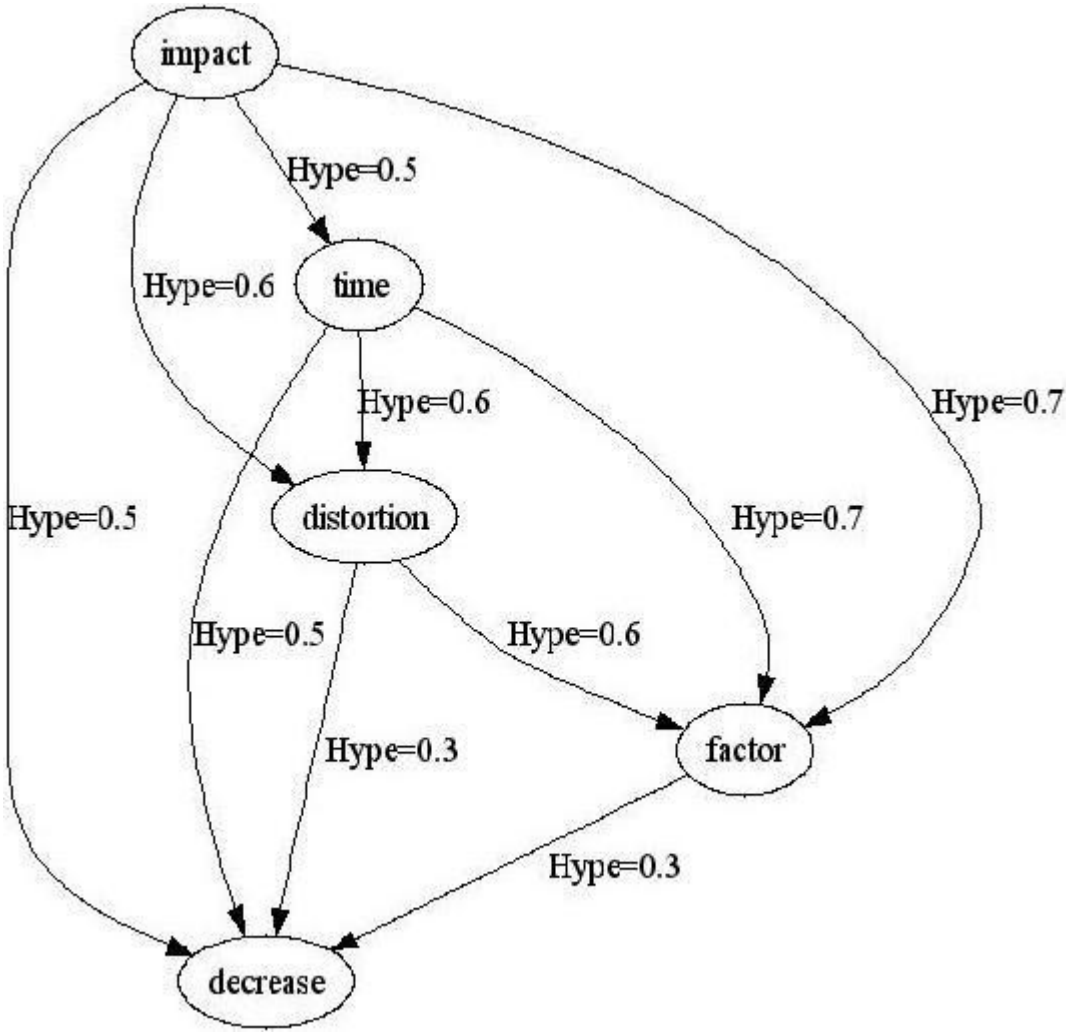


Figura 10: Parte do Grafo gerado a partir dos substantivos do texto