Desenvolvimento de Algoritmos para Mineração de Textos

Fernando E. B. Otero Alex A. Freitas

Pontifícia Universidade Católica do Paraná
Programa de Pós-Graduação em Informática Aplicada
Rua Imaculada Conceição, 1155
Curitiba - PR, 80215-901, Brasil
{fbo, alex}@ppgia.pucpr.br
http://www.ppgia.pucpr.br/∼alex

Resumo

Com o aumento das publicações escritas em geral e a grande popularização da Internet, o número de informações disponível e acessível tem aumentado rapidamente. Neste enfoque, uma das técnicas que tem recebido crescente atenção na literatura é a sumarização automática de textos, uma sub-área da mineração de textos. Neste trabalho é apresentado o desenvolvimento de algoritmos para mineração de textos, especificamente de sumarização, para obter conhecimento automaticamente a partir de textos. Os resultados obtidos foram satisfatórios, tendo em vista a dificuldade desta tarefa.

1 Introdução

Sumarização de textos é o processo de reduzir o tamanho de um texto, preservando o seu conteúdo informativo. Produzir um sumário de um texto qualquer é um desafio que idealmente requer o completo entendimento do texto, fato que esta além do estado da arte atual da computação [Brandow et al. 94, Mitra et al. 97]. Segundo [Sparck Jones 99], a tarefa de sumarização pode ser dividida em 3 etapas: (a) criar uma representação do texto de origem; (b) transformar a representação do texto de origem para uma representação de sumário; (c) gerar o sumário a partir da representação.

A grande maioria dos sistemas de sumarização existentes efetuam uma simplificação da tarefa através da sumarização extrativa, na qual sentenças do documento original são selecionandas de acordo com critérios pré-estabelecidos para compor o sumário. Desta forma, elimina-se a necessidade de um total entendimento do texto e da geração de um novo texto em linguagem natural. Os sumários obtidos não possuem necessariamente coerência narrativa, mesmo assim podem ser usados para uma rápida assimilaçã do texto e para o julgamento

de relevância. Um sumário extrativo pode ser definido como um sub-conjunto de frases do texto original que é representativo do seu conteúdo.

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

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

Diversas heurísticas foram propostas para sumarização extrativa, mas nenhum critério claro foi definido para escolher uma delas. Existem evidências [Kupiec et al. 95] que sugerem a combinação de diversas heurísticas para alcançar melhores resultados. As heurísticas são utilizadas para identificar as frases significativas do texto. Foram feitas muitas sugestões de quais carcterísticas contribuem para a relevância das frases, tais como, a presença de palavras indicativas (*cue words*) ou de palavras do título do texto [Edmundson 69], a posição da frase no texto [Edmundson 69], o uso de técnicas de recuperação de informação utilizando estatísticas baseadas na freqüência de palavras para calcular o grau de importância de uma palavra [Luhn 58], assumindo que as frases mais importantes existem em posições mais coesas (frases mais conectadas) do texto [Barzilay & Elahad 97, Mitra et al. 97], incorporando a estrutura de discurso (retórica) do texto [Marcu 99, Teufel & Moens 98].

Em [Kupiec et al. 95] é apresentada uma proposta de geração de sumários extrativos utilizando-se um classificador Bayesiano Simples (Naive-Bayes). As características utilizadas para a classificação pressupõem uma semi-estruturação do texto, como títulos de seção e separação do texto em parágrafos. Uma vez que o reconhecimento de seções e parágrafos em um texto não-estruturado não é uma tarefa trivial, estamos interessados em características de classificação que possam ser obtidas de textos livres, sem uma estruturação especial. Adicionalmente, introduzimos o uso do método do Vizinho Mais Próximo (*Nearest Neighbors*) para a tarefa de classificação.

A estrutura do artigo é descrita a seguir. A seção 2 apresenta as abordagens utilizados em trabalhos relacionados. A seção 3 detalha a metodologia utilizada pelo nosso sistema e a seção 4 apresenta os resultados obtidos. Finalmente, a seção 5 apresenta uma breve discussão sobre os resultados alcançados e a seção 6 conclui o artigo.

2 Trabalhos Relacionados

O trabalho que mais motivou a pesquisa na área de sumarizadores treináveis foi [Kupiec et al. 95], que apresentou o problema de sumarização extrativa como

um problema estatístico de classificação, cuja necessidade é criar uma função que classifique a probabilidade de uma sentença ser incluída no sumário. Posteriormente, as sentenças são ordenadas de acordo com as probabilidades obtidas e o sumário extrativo é gerado. O sistema utilizava 5 características para categorizar as frases, todas de natureza discreta: (a) *Sentence Length Cut-off* – indicando o tamanho da sentença; (b) *Fixed-Phrase* – presença de palavras indicativas (*cue words*) e ocorrência em seções contendo determinadas palavaras-chave (ex. conclusão, resultados); (c) *Paragraph* – posição da sentença no parágrafo e no texto; (d) *Thematic Word* – presença de palavras freqüentes; (e) *Uppercase Word* – presença de palavras em maiúscula (excluindo a primeira palavra de cada setença e abreviaturas comuns). A classificação foi realizada utilizado-se um classificador Bayesiano Simples (Naive-Bayes) para calcular a probabilidade de cada sentença ser incluída no sumário. A base de treinamento continha 188 documentos, com sumários extrativos produzidos através de: (a) alinhamento automático de sentenças com o sumário manual (79% das sentenças); (b) associação realizada por juízes humanos. Os resultados foram calculados através de validação cruzada. A maior taxa de acerto de similaridade entre as sentencças selecionandas pelo sistema e os sumários extrativos foi de 44%, utilizando-se a combinação das características *Sentence Length Cut-off*, *Fixed-Phrase* e *Paragraph*.

Em [Teufel & Moens 98] é apresentada uma extensão da metodologia de [Kupiec et al. 95] para a extração de sentenças. Teufel & Moens augumentaram que o conhecimento da estrutura de discurso (retórica) de artigos científicos é importante para gerar sumários flexíveis e de sub-domínio independente. Foram definidas regras retóricas para classificar as frases em: (a) Introdução; (b) Tópicos; (c) Trabalhos Realcionados; (d) Objetivos e Problema; (e) Solução e Métodos; (f) Resultados; (g) Conclusão. O processo de sumarização foi dividido em duas partes: (1) extração das frases candidatas: identificação das frases que trazem qualquer regra retórica; (2) identificação das regras retóricas: tenta classificar as frases de acordo com sua regra retórica. Para o treinamento do sumarizador foram utilizadas 6 heurísticas, 4 similares a [Kupiec et al. 95] (*indicator phrase*, *location*, *sentence length* e *thematic word*) e 2 adicionais: (a) *Title* – presença de palavras do título do texto na frase; (b) *Header* – guarda a informação da seção retórica da frase, especificando a seção na qual a frase aparece no texto (Introdução, Conclusão, etc.). A base de treinamento utilizada foi a CM-PLG, contendo 201 artigos técnicos. Os resultados foram analisados utilizando-se o classificador Bayesiano Simples (Naive-Bayes) através de validação cruzada. O sistema identificou 68% das frases relevantes do texto e atribuiu corretamente a regra retórica para 68% das frases selecionadas, tendo uma performance geral de 46%.

3 Metodologia

Sumarizadores extrativos geralmente computam uma pontuação para cada frase do documento e selecionam um conjunto de frases com as maiores pontuações para compor o sumário, sem modificá-las. A definição das regras de pontuação

são baseadas em características observadas, sem uma formulação precisa.

Em posse de um conjunto de documentos de treinamento, com seus sumários extrativos respectivos, podemos abordar o problema de sumarização extrativa como um problema de classificação [Kupiec et al. 95], onde o objetivo é classificar cada frase em uma dentre duas classes: frases incluídas e não-incluídas no sumário. Seguindo esta abordagem, definimos um conjunto de características para categorizar as frases e os métodos para a tarefa de classificação das mesmas.

3.1 Características das frases

Definimos 5 características para categorizar as frases: 4 são variações de características utilizadas ou sugeridas por [Kupiec et al. 95] (*Cue Words*, *Position*, *Size* e *Uppercase Word*) e 1 característica adicional (*Average TF-DF*).

A definição foi direcionada procurando-se estabelecer características que não fossem dependentes da estrutura do texto, utilizando informações presentes em qualquer tipo de texto. Porém, dois métodos (*Cue Words* e *Average TF-DF*) necessitam de uma lista pré-definida de palavras, tornando-os dependentes de uma linguagem. Como a base de documentos utilizada contêm textos em inglês, esta linguagem foi adotada.

Average TF-DF – Para cada palavra do texto é calculado um peso baseado na frequência com que ele aparece no texto. Este peso é denominado *TF-DF (Term Frequency - Document Frequency)* e é calculado através da fórmula:

$$TF - DF(p) = tf(p, d) * df(p)$$

O $tf(p, d)$ é o número de ocorrências da palavra p no documento d. O $df(p)$ é o número de documentos em que a palavra p aparece pelo menos uma vez. Como em nossa abordagem temos um conjunto de frases (o texto do documento) em vez de um conjunto de documentos, o que é um documento na fórmula acima passa a ser uma frase no nosso trabalho. Para o cálculo desta característica, o texto é convertido para a representação emphbag-of-words [Joachims 96]. Esta representação consistem em guardar a frequência com que a palavra aparece no texto. Entretanto, nã são armazenadas todas as palavras, sendo o texto submetido a três passos de pré-processamento: *case-folding*, remoção de *stop words* e *stemming* [Witten et al. 94]. *Case folding* consiste em converter todos os caracteres das palavras do texto para a mesma forma, maiúsculos ou minúsculos. Assim, as palavras "the", "The" e "THE" são convertidas para o mesmo formato "the". Remoção de *stop words* consiste em remover as palavras que ocorrem com muita frequência no texto, trazendo pouca informação sobre o conteúdo da frase, baseado numa lista de palavras comuns. *Stemming* consiste em converter a palavra em seu radical, retirando a flexão verbal, o plural e a sua derivação. O algoritmo utilizado para esta tarefa foi desenvolvido por Porter [Porter 80] para a lingua inglesa.

O valor desta característica é a soma dos pesos (*TF-DF*) de todas as

palavras da frase dividido pelo número de palavras da frase. A fim de normalizar os valores, dividimos o *Average TF-DF* de cada frases pelo *Average TF-DF* mais alto dentre todas as frases.

Cue Words – Definimos uma lista de palavras que sugerem uma importância para a frase. Esta característica é verdadeira para todas as frases que contêm qualquer uma das palavras da lista, falsa caso contrário.

Position – As frases mais significativas em um documento tendem a aparecer no começo e no final do texto. Esta característica classifica as frases em: frases iniciais, frases médias e frases finais.

Size – Frases curtas tendem a não ser incluídas no sumário. Este característica é verdadeira para as frases que são maiores que um *threshold* (ex. 5 palavras), falsa caso contrário.

Uppercase Word – Abreviaturas de palavras (ex. PUCPR) geralmente são importantes. Esta característica é verdadeira se a frase contém palavras com todas as letras em maiúsculas e estas ocorrem em pelo menos uma outra frase, falsa caso contrário. Para evitar a ocorrência de abreviaturas de unidades de medida (ex. Kg, C, etc.) e das palavras iniciais de frases, a palavra deve ter pelo menos duas letras maiúsculas propriamente ditas, na faixa de A . . . Z.

3.2 Classificadores

Utilizando as características das frases, necessitamos um método para selecionar as frases que devem ser incluídas no sumário. [Kupiec et al. 95] utilizou um classificador Bayesiano Simples (Naive-Bayes) para computar a probabilidade da frase ser incluída no sumário, selecionando as frases com as maiores probabilidades para compor o sumário. Adicionalmente, utilizamos o método do Vizinho Mais Próximo (*Nearest Neighbors*), que classifica as frases de acordo com as suas frases vizinhas.

3.2.1 Classificador Bayesiano Simples (Naive-Bayes)

Para cada frase s, computamos a probabilidade dela ser incluída no sumário S dadas as suas características A_j, $1 \leq j \leq m$, onde m é o número de características utilizadas. Esta probabilidade é dada pela fórmula de Bayes, que assume que as características são estatísticamente independentes, descrita a seguir:

$$P(s \in S|A_1, A_2, \ldots, A_j) = \frac{\sum_{j=1}^{m} P(A_j|s \in S) * P(s \in S)}{\sum_{j=1}^{m} P(A_j)}$$

onde $P(s \in S)$ é a probabilidade da frase s pertencer ao sumário S, $P(A_j|s \in S)$ é a probabilidade de ocorrência do atributo A_j dado que s pertence ao sumário S e

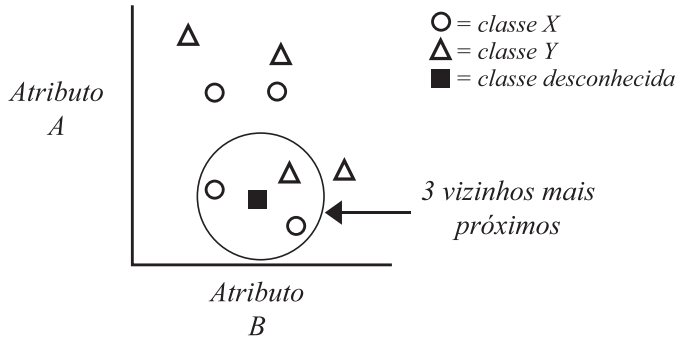

Figura 1: Exemplo do método do Vizinho Mais Próximo

$P(A_j)$ é a probabilidade de ocorrência do atributo A_j levando-se em consideração todas as frases do texto. $P(s \in S)$ é uma constante, $P(A_j)$ e $P(A_j|s \in S)$ podem ser estimadas diretamente dos documentos de treinamento, contando as respectivas ocorrências de valores de atributos. Com isto temos uma função classificadora que para cada frase s gera uma pontuação (probabilidade), sendo que as frases as maiores pontuações serão selecionadas para compor o sumário.

3.2.2 Método do Vizinho Mais Próximo (*Nearest Neighbors*)

Neste método de classificação, cada característica representa uma dimensão no espaço de dados, mostrado na Figura 1. Os dados conhecidos (dados de treinamento) são colocados nesse espaço como pontos baseados em seu valores das características e rotulados com as suas respectivas classes. Um exemplo desconhecido (dados de teste) é colocado no mesmo espaço baseado em seu valores das características e classificado de acordo com seus k vizinhos mais próximos (*K Nearest Neighbors - KNN*), onde k é um número inteiro. Na Figura 1, com $k = 3$, o exemplo desconhecido é rotulado para a classe X baseado no fato que 2 dos 3 vizinhos mais próximos pertencem a classe X.

4 Resultados

Para avaliar o sistema, utilizamos a base de textos da *TIPSTER Text Summarization Evaluation Conference (SUMMAC)* [Mani et al. 98]. Uma vez que esta base contêm documentos mas não define nenhum sumário "ideal" para cada documento, estes sumários foram feitos manualmente por um juiz humano. Foram selecionados 40 documentos de jornais, com uma média de 42 sentenças, e para cada um foi produzido um sumário manual de forma extrativa equivalente à 10% do número de sentenças do texto original.

Como nossa base de dados é pequena para separar dados de treinamento

Tabela 1: Taxa de acerto do classificador Bayesiano Simples

Atributo (*Feature*)	Taxa de Acerto (%)	Taxa de Acerto Acumulada (%)
Position	32.18	32.18
Size	32.18	32.18
Average TF-DF	28.88	33.33
Uppercase word	24.71	32.18
Cue words	21.83	31.01
Baseline	32.13	–

e dados de teste, utilizamos o método "deixa um fora" (*leave-one-out*). Desta forma, cada documento foi selecionado para ser testado utilizando-se os documentos restantes como treinamento. Para cada documento de teste, o sumarizador produziu um sumário com o mesmo número de sentenças do sumário manual. Os resultados foram avaliados através das métricas tradicionais *precision* e *recall*. Entretanto, no nosso caso *precision* = *recall*, logo o termo taxa de acerto foi utilizado.

4.1 Resultados do classificados Bayesiano Simples

A Tabela 1 mostra os resultados obtidos utilizando-se o classificador Bayesiano Simples. A segunda coluna da Tabela 1 mostra a taxa de acerto das características quando utilizadas individualmente, sendo as melhores taxas individuais das características *Position* e *Size*. A terceira coluna da Tabela 1 mostra a taxa de acerto das características combinadas em ordem decrescente, com a melhor taxa acumulada quando combinamos as carcterísticas *Position*, *Size* e *Average TF-DF*. A *baseline* utilizada foi a de selecionar as n primeiras frases do texto, onde n é o número de frases do sumário manual.

4.2 Resultados do método do Vizinho Mais Próximo

Utilizando-se o método do Vizinho Mais Próximo, com $k = 5$, obtemos uma taxa de acerto de 32.45%. A *baseline* utilizada para este método foi a mesma que a anterior.

5 Discussão

Os resultados obtidos são estatísticamente semelhantes à heurística de selecionar as n primeiras frases do texto. Deve-se destacar que esta heurística é especialmente eficiente em textos de jornais (notícias) [Brandow et al. 94], onde as frases mais importantes concentram-se no início do texto. A melhor taxa de acerto obtida pelo sumarizador foi alcançada utilizando-se a combinação dos atribu-

tos *Position*, *Size* e *Average TF-DF* juntamente com o classificador Bayesiano Simples, sendo esta de 33.33%.

Devido a dificuldade de obter-se uma base de textos, com pares de documentos/sumários, não foram realizados testes utilizando-se documentos de outros domínios (ex. artigos técnicos). Na literatura, ainda não existe uma base de textos padrão e nem um critério para a definição de um sumário "ideal", o que dificulta a comparação entre as técnicas de sumarização existentes.

6 Conclusão

Neste trabalho apresentou-se o desenvolvimento de algoritmos para sumarização automática de textos. A geração automática de suários apresenta uma grande flexibilidade, mesmo quando os sumários automáticos são comparados aos sumários fornecidos pelos próprios autores. A maior taxa de acerto obtida (33.33%) não apresenta um ganho significativo em comparação com a heurística de selecionar as primeiras frases do texto (32.13%). Como nossa proposta é independente do domínio do texto, sua aplicação em textos de outros domínios pode apresentar melhores resultados. Os resultados obtidos estimulam o estudo de novos atributos relevantes para sumarização e de outros métodos de classificação.

Referências

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

[Brandow et al. 94] R. Brandow, K. Mitze e L. Rau. Automatic Condensation of Eletronic Publications by Sentence Selection. *Information Processing and Management*, 31(5):675–685. 1994.

[Edmundson 69] H. P. Edmundson. New methods in automatic abstracting. *Journal of the ACM*, 16(2):264-285, Abril 1969. Re-impresso em I. Mani e M. Maybury, (eds.), *Advances in Automatic Text Summarization*, 23–42. The MIT Press. 1999.

[Joachims 96] T. Joachims. 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.

[Kupiec et al. 95] J. Kupiec, J. Pedersen e F. Chen. A Trainable Document Summarizer. Em *Proceedings of the 18th ACM-SIGIR Conderence, Association of Computing Machinery, Special Interest Group Information Retrieval*, 68–73. 1995.

[Luhn 58] H. P. Luhn. The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development* 2(92):159–165. 1958. Re-impresso em I. Mani e M. Maybury, (eds.), *Advances in Automatic Text Summarization*, 15–21. The MIT Press. 1999.

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

[Marcu 99] D. Marcu. Discourse trees are good indicators of importance in text. Em I. Mani e M. Maybury, (eds.), *Advances in Automatic Text Summarization*, 123–136. The MITRE Press. 1999.

[Mitra et al. 97] M. Mitra, A. Singhal e C. Buckley. Automatic Text Summarization by Paragraph Extraction. Em *Proceeding of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*. Madrid, Espanha. 1997.

[Porter 80] M. F. Porter. An algorithm for suffix stripping. *Porgram 14*, 130–137. 1980. Re-impresso em K. Sparck Jones e P. Willet, (eds.), *Readings in Information Retrieval*, 313–316. Morgan Kaufmann. 1997.

[Sparck Jones 99] K. Sparck Jones. Automatic Summarizing: factors and directions. Em I. Mani e M. Maybury, (eds.), *Advances in Automatic Text Summarization*, 1–12. The MIT Press. 1999.

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

[Teufel & Moens 98] S. Teufel e M. Moens. Sentence extraction and rhetorical classification for flexible abstracts. *Intelligent Text Summarization: Papers from the 1998 AAAI Spring Symposium*. Technical Report 55-98-06. AAAI Press, 1998.