

Ovídio José Francisco

**Aplicação de técnicas de Mineração de Textos
para Organização e Extração de Históricos de
Decisões de Documentos de Reuniões**

Sorocaba, SP

19 de setembro de 2017

Sumário

1	PROPOSTA	3
1.1	Módulo de preparação e manutenção	3
1.1.1	Preparação dos documentos	4
1.1.1.1	Segmentação	4
1.1.1.2	Avaliação dos Segmentadores	5
	Referências	7

1 Proposta

Essa seção apresenta as etapas de desenvolvimento do sistema, bem como o seu funcionamento geral, desde a preparação dos documentos até a entrega dos históricos de ocorrência ao usuário. Inicialmente será descrito a seleção e pré-processamento. Em seguida, será relatado como as técnicas de mineração de texto e resgate de informação são utilizadas nesse trabalho.

O objetivo do sistema é permitir ao usuário consultar uma coleção de documentos de reuniões a fim de obter todo o histórico de ocorrências de um determinado tema pesquisado, podendo identificar nos documentos onde o tema foi mencionado como informe ou onde houve uma decisão sobre o tema. Para isso, o sistema é dividido em dois módulos principais: Módulo de preparação e manutenção e Módulo de consulta.

O módulo de preparação e manutenção recebe uma coleção de documentos e produz uma estrutura de dados interna, que é utilizada pelo módulo de consulta, que por sua vez, é responsável por receber a intensão usuário, proceder a busca na estrutura de dados interna e retornar os trechos associados com a intensão do usuário, tanto quanto ao assunto como no tipo de ocorrência. A Figura 1 mostra a visão geral do sistema com suas principais entradas e saídas.

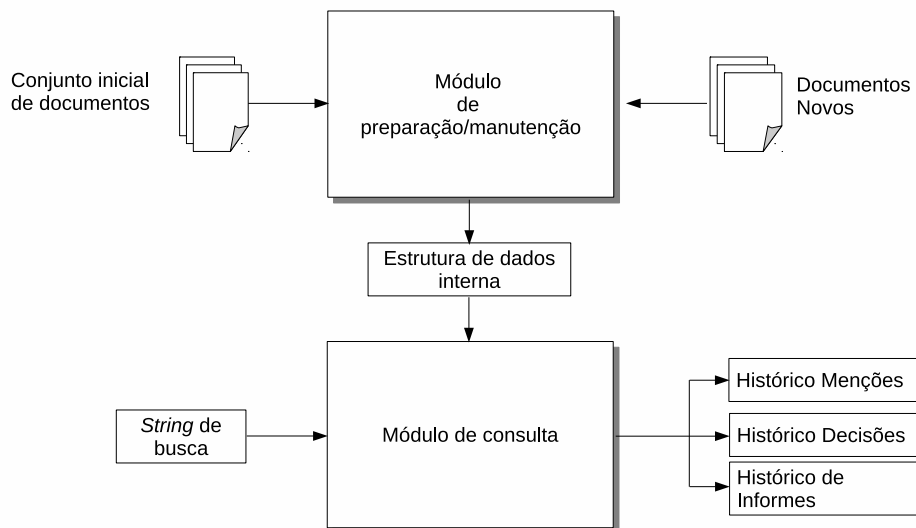


Figura 1 – Visão geral do sistema

1.1 Módulo de preparação e manutenção

O módulo de preparação e manutenção tem como funções principais dividir cada ata em segmentos de texto que contêm um assunto predominante, e descrevê-los por

meio de técnicas de extração tópicos e classificação. Além disso, produz uma estrutura de dados que registra quais assuntos foram tratados na reunião, bem como o trecho do documento onde é discutido.

1.1.1 Preparação dos documentos

As atas são normalmente armazenadas em arquivos do tipo *pdf*, *doc*, *docx* ou *odt* que normalmente possuem formato binário. O texto deve ser preparado para os métodos de MT e RI. Inicialmente, o texto puro é extraído e passa por processos de transformação conforme apresentados a seguir.

1. Remoção de cabeçalhos e rodapés: as atas contém trechos que podem ser considerados pouco informativos e descartados durante o pré-processamento, como cabeçalhos e rodapés que se misturam aos tópicos tratados na reunião, podendo ser inseridos no meio de um tópico prejudicando tanto os algoritmos de MT e RI, quanto a leitura do texto pelo usuário.
2. Identificação de finais sentenças: devido ao estilo de pontuação desses documentos, como encerrar sentenças usando um ";" e inserção de linhas extras, foram usadas as regras especiais para identificação de finais de sentença. Cada final de sentença é identificado e marcado com uma *string* especial, esse processo é melhor descrito na Subseção ??.
3. Redução de termos: eliminou-se as *stop words* por meio de uma lista de 438 palavras. Além disso, eliminou-se a acentuação, sinais de pontuação, numerais e todos os *tokens* menores que três caracteres.
4. *Stemming*: extraiu-se o radical de cada palavra. Para isso, as letras foram convertidas em caixa baixa e aplicou-se o algoritmo *Orengo*¹ para remoção de sufixos.

A Figura 2 mostra a etapa de preparação de um documento em português.

Ao final

1.1.1.1 Segmentação

Como já mencionado, uma ata registra a sucessão de assuntos discutidos em uma reunião, porém apresenta-se com poucas quebras de parágrafo e sem marcações de estrutura, como capítulos, seções ou quaisquer indicações sobre o assunto do texto. Portanto, faz-se necessário descobrir quando há uma mudança de assunto no texto da ata. Para essa tarefa, as técnicas de segmentação de texto recebem uma lista de sentenças, da qual considera

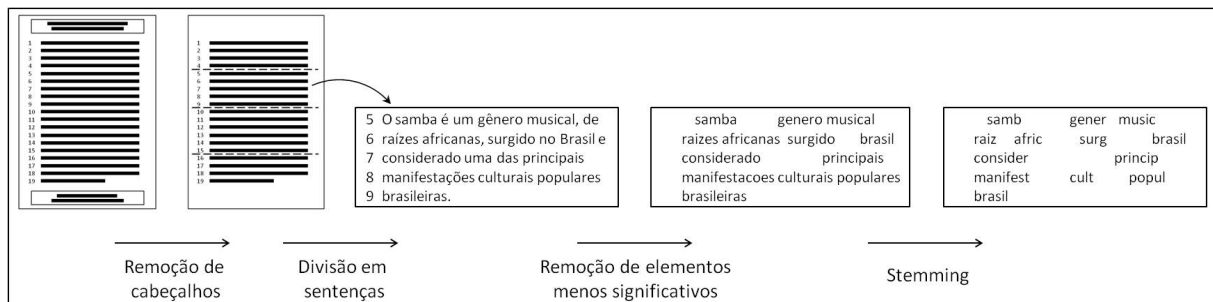


Figura 2 – Exemplo de pré-processamento.

cada ponto entre duas sentenças como candidato a limite, ou seja, um ponto onde há transição entre assuntos.

Entre os principais trabalhos da literatura podemos citar o *TextTiling* (??) e o *C99* (??). O *TextTiling* é um algoritmo baseado em janelas deslizantes, em que, para cada candidato a limite, analisa-se o texto circundante. Um limite ou quebra de segmento é identificado sempre que a similaridade cai abaixo de um limiar. Possui baixa complexidade computacional e acurácia semelhante a algoritmos mais complexos baseados em matrizes de similaridade como o (*C99*).

Para cada posição candidata o *TextTiling* constrói 2 blocos, um contendo sentenças que a precedem e outro com as que a sucedem. O tamanho desses blocos é um parâmetro a ser fornecido ao algoritmo e determina o tamanho mínimo de um segmento.

O *C99* usa matrizes de *rakings* de similaridades e técnicas de *clustering* para encontrar os limites entre os segmentos. Oferece resultados melhores que algoritmos baseados em janelas deslizantes ao custo de maior complexidade computacional.

Inicialmente é construída uma matriz que contém as similaridades de todas as unidades de texto. Em seguida, essa é transformada substituindo-se cada elemento da matriz original pelo número de elementos vizinhos com similaridade inferior. Finalmente, utiliza um método de *clustering* baseado no algoritmo de maximização de Reynar para identificar os limites entre os segmentos.

1.1.1.2 Avaliação dos Segmentadores

Para que se possa avaliar um segmentador automático de textos é preciso uma referência, isto é, um texto com os limites entre os segmentos conhecidos. Essa referência, deve ser confiável, sendo uma segmentação legítima que é capaz de dividir o texto em porções relativamente independentes, ou seja, uma segmentação ideal.

Para este trabalho, um bom método de segmentação é aquele cujo resultado melhor se aproxima de uma segmentação manual, sem a obrigatoriedade de estar perfeitamente alinhado com tal. Ou seja, visto o contexto das atas de reunião, e a subjetividade da tarefa,

não é necessário que os limites entre os segmentos (real e hipótese) sejam idênticos, mas que se assemelhem em localização e quantidade.

Os algoritmos foram comparados com a segmentação fornecida pelos participantes das reuniões e calculou-se as medidas mais aplicadas à segmentação textual, P_k e *WindowDiff*. Além dessas, computou-se também as medidas tradicionais acurácia, precisão, revocação e F^1 para comparação com outros trabalhos que as utilizam.

O algoritmo *C99* obteve melhor desempenho em acurácia, precisão, F^1 , P_k e *WindowDiff*, enquanto o *TextTiling* obteve o melhor desempenho em revocação como pode ser visto na Tabela 1.

Algoritmo	Medida	Média
<i>C99</i>	P_k	0,116
<i>C99</i>	<i>WindowDiff</i>	0,390
<i>C99</i>	Acurácia	0,609
<i>C99</i>	Precisão	0,720
<i>C99</i>	F^1	0,655
<i>TextTiling</i>	Revocação	0,917

Tabela 1 – Melhores resultados obtidos.

Verificou-se que, de maneira geral, o algoritmo *C99* apresenta melhores resultados em relação ao *TextTiling*, contudo, testes estatísticos realizados indicaram que não houve diferença significativa entre os métodos. Nesse trabalho, escolheu-se o algoritmo *C99* por apresentar resultados satisfatórios e sua ligeira superioridade em relação ao *TextTiling*.

Após a extração e preparação do texto, o algoritmo recebe uma lista de sentenças

Após a identificação dos segmentos, o algoritmo retorna uma lista onde cada elemento é um texto com um assunto predominante.

Referências