Revista de Sistemas de Informacao da FSMA n. 7 (2011) pp. 7-21

SISTEMAS DE INFORMAÇÃO

http://www.fsma.edu.br/si/sistemas.html

TRILHA PRINCIPAL

O uso da Mineração de Textos para Extração e Organização Não Supervisionada de Conhecimento

Solange O. Rezende, Ricardo M. Marcacini, Maria F. Moura
Instituto de Ciências Matemáticas e de Computação - ICMC-USP
Embrapa Informática Agropecuária
solange@icmc.usp.br, rmm@icmc.usp.br, fernanda@cnptia.embrapa.br

***Resumo*—O avanço das tecnologias para aquisição e armazenamento de dados tem permitido que o volume de informação gerado em formato digital aumente de forma significativa nas organizações. Cerca de 80% desses dados estão em formato não estruturado, no qual uma parte significativa são textos. A organização inteligente dessas coleções textuais é de grande interesse para a maioria das instituições, pois agiliza processos de busca e recuperação da informação. Nesse contexto, a Mineração de Textos permite a transformação desse grande volume de dados textuais não estruturados em conhecimento útil, muitas vezes inovador para as organizações. Em especial, o uso de métodos não supervisionados para extração e organização de conhecimento recebe grande atenção na literatura, uma vez que não exigem conhecimento prévio a respeito das coleções textuais a serem exploradas. Nesse artigo são descritas as principais técnicas e algoritmos existentes para extração e organização não supervisionada de conhecimento a partir de dados textuais. Os trabalhos mais relevantes na literatura são apresentados e discutidos em cada fase do processo de Mineração de Textos; e, são sugeridas ferramentas computacionais existentes para cada tarefa. Por fim, alguns exemplos e aplicações são apresentados para ilustrar o uso da Mineração de Textos em problemas reais.**

***Index Terms*—Mineração de Textos, Agrupamento de Documentos, Aprendizado Não Supervisionado, Extração de Metadados, Hierarquias de Tópicos**

I. Introdução

O avanço das tecnologias para aquisição e armazenamento de dados tem permitido que o volume de informação gerado em formato digital aumente de forma significativa nas organizações. Estimativas indicam que, no período de 2003 a 2010, a quantidade de informação no universo digital aumentou de cinco hexabytes (aproximadamente cinco bilhões de gigabytes) para 988 hexabytes [1]. Até o ano de 2008, contabilizou-se que a humanidade produziu cerca de 487 hexabytes de informação digital [2], [3].

Cerca de 80% desses dados estão em formato não estruturado, no qual uma parte significativa são textos [4]. Esses textos constituem um importante repositório organizacional, que envolve o registro de histórico de atividades, memorandos, documentos internos, e-mails, projetos, estratégias e o próprio conhecimento adquirido [5]. A organização inteligente dessas coleções textuais é de grande interesse para a maioria das instituições, pois agiliza processos de busca e recuperação da informação. No entanto, o volume de dados textuais armazenados é tal que extrapola a capacidade humana de, manualmente, analisá-lo e compreendê-lo por completo.

Nesse contexto, a Mineração de Textos permite a transformação desse grande volume de dados textuais não estruturados em conhecimento útil, muitas vezes inovador para as organizações. Até pouco tempo esse fato não era visto como uma vantagem competitiva, ou como suporte à tomada de decisão, como indicativo de sucessos e fracassos. O seu uso permite extrair conhecimento a partir de dados textuais brutos (não estruturados), fornecendo elementos de suporte à gestão do donhecimento, que se refere ao modo de reorganizar como o conhecimento é criado, usado, compartilhado, armazenado e avaliado. Tecnologicamente, o apoio de Mineração de Textos à gestão do conhecimento se dá na transformação do conteúdo de repositórios de informação em conhecimento a ser analisado e compartilhado pela organização.

Uma tendência entre os serviços de consultoria em Mineração de Textos são aplicações que visam aumentar os parâmetros disponíveis para a inteligência competitiva. Resumidamente, a inteligência competitiva consiste em uma empresa descobrir onde leva vantagem ou desvantagem sobre suas concorrentes, utilizando-se a análise de seu ambiente interno versus o ambiente externo. Para construir a base da inteligência competitiva, isto é, sistemas de análise do ambiente interno contra o externo, é necessário organizar o conhecimento do ambiente interno (*business intelligence*). Nesse ponto, a Mineração de Textos pode auxiliar a construção de uma *document warehouse*, isto é, um repositório de documentos que podem incluir informações extensivas sobre os mesmos, como agrupamentos de documentos similares, relações cruzadas entre características de documentos, metadados automaticamente obtidos, e, várias outras informações que possam significar uma melhoria na recuperação da informação em tarefas importantes nas instituições [6].

Entre as diversas maneiras de se instanciar um processo de Mineração de Textos, o uso de métodos não supervisionados para extração e organização de conhecimento recebe grande atenção na literatura, uma vez que não exigem conhecimento prévio a respeito das coleções textuais a serem exploradas. Um processo de Mineração de Textos para extração e organização não supervisionada de conhecimento pode ser dividido em três fases principais: Pré-Processamento dos Documentos, Extração de Padrões com Agrupamento de Textos e Avaliação

do Conhecimento. No Pré-processamento dos Documentos os dados textuais são padronizados e representados de forma estruturada e concisa, em um formato adequado para extração do conhecimento. Assim, na Extração de Padrões, métodos de agrupamento de textos podem ser utilizados para a organização de coleções textuais de maneira não supervisionada [7]. Em tarefas de agrupamento, o objetivo é organizar um conjunto de documentos em grupos, em que documentos de um mesmo grupo são altamente similares entre si, mas dissimilares em relação aos documentos de outros grupos [8]. Os métodos de agrupamento também são conhecidos como algoritmos de aprendizado por observação ou análise exploratória dos dados, pois a organização obtida é realizada por observação de regularidades nos dados, sem uso de conhecimento externo [9]. Por fim, na Avaliação do Conhecimento, os resultados obtidos são avaliados de acordo com o contexto do problema, bem como a novidade e utilidade do conhecimento extraído.

Ao final desse processo, as coleções textuais são organizadas em grupos de documentos. Em especial, busca-se uma organização hierárquica da coleção, na qual os documentos são organizados em grupos e subgrupos, e cada grupo contém documentos relacionados a um mesmo tema [7], [10], [11]. Os grupos próximos à raiz representam conhecimento mais genérico, enquanto seus detalhamentos, ou conhecimento mais específico, são representados pelos grupos de níveis mais baixos. Dessa forma, o usuário pode visualizar a informação de interesse em diversos níveis de granularidade e explorar interativamente grandes coleções de documentos. Os resultados obtidos por meio desse processo auxiliam diversas tarefas de organização da informação textual, partindo-se da hipótese que se um usuário está interessado em um documento específico pertencente a um grupo, deve também estar interessado em outros documentos desse grupo e de seus subgrupos [12], [10].

Em vista das vantagens desse processo de Mineração de Textos e das diversas aplicações e sistemas que se beneficiam dos resultados obtidos, nesse artigo são descritas as principais técnicas e algoritmos existentes para extração e organização não supervisionada de conhecimento a partir de dados textuais. Os trabalhos da literatura mais relevantes são apresentados e discutidos em cada fase do processo de Mineração de Textos. Ainda, são sugeridas ferramentas computacionais existentes para cada tarefa. Por fim, alguns exemplos e aplicações são apresentados para ilustrar o uso da Mineração de Textos aplicados em problemas reais.

II. Objetivos e Justificativa

O objetivo principal da metodologia descrita neste artigo é orientar um processo de extração de informação e estruturação de uma coleção de textos, supondo-se apenas o uso de métodos não supervisionados, e cobrindo as etapas de Pré-Processamento dos Documentos, Extração de Padrões com Agrupamento de Textos e Avaliação do Conhecimento. Como objetivos específicos são citados:

1) Obter atributos que sejam candidatos a termos do domínio de conhecimento da coleção, selecionando palavras ou combinações de palavras estatisticamente mais significantes na coleção;
2) Realizar a identificação e construção de uma organização da coleção, a partir de algoritmos de agrupamento de textos e técnicas para selecionar descrições aproximadas de cada grupo;
3) Auxiliar processos automáticos de recuperação da informação nos textos, a partir dos descritores associados a cada grupo;
4) Ilustrar exemplos de aplicações atuais e úteis que auxiliam processos de apoio à tomada de decisão por meio do conhecimento extraído.

III. Metodologia

A Mineração de Textos pode ser definida como um conjunto de técnicas e processos para descoberta de conhecimento inovador a partir de dados textuais [13]. Em um contexto na qual grande parte da informação corporativa, como e-mails, memorandos internos e blogs industriais, é registrada em linguagem natural, a Mineração de Textos surge como uma poderosa ferramenta para gestão do conhecimento.

Pode-se afirmar que a Mineração de Textos é uma especialização do processo de mineração de dados. A principal diferença entre os dois processos é que, enquanto a mineração de dados convencional trabalha exclusivamente com dados estruturados, a Mineração de Textos lida com dados inerentemente não-estruturados [14]. Logo, na Mineração de Textos o primeiro desafio é obter alguma estrutura que represente os textos e então, a partir dessa, extrair conhecimento.

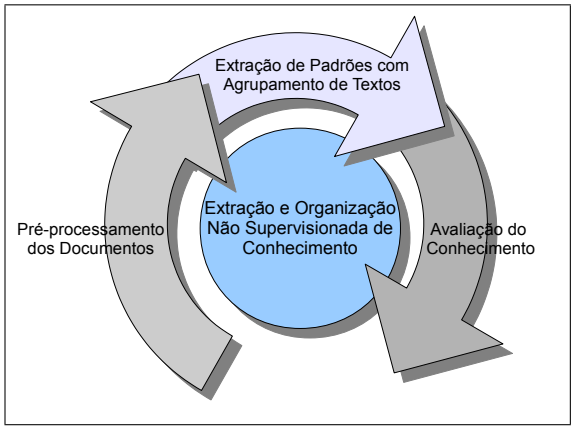

Figura 1. Fases da metodologia para extração e organização não supervisionada de conhecimento

Para a extração e organização não supervisionada de conhecimento a partir de dados textuais, o diferencial está na etapa de extração de padrões, na qual são utilizados métodos de agrupamento de textos para organizar coleções de documentos em grupos. Em seguida, são aplicadas algumas técnicas de seleção de descritores para os agrupamentos formados, ou seja, palavras e expressões que auxiliam a interpretação dos grupos. Após validação dos resultados, o agrupamento hierárquico e seus descritores podem ser utilizados como uma hierarquia de tópicos para tarefas de análise exploratória dos textos [15], [16], [17], além de apoiar sistemas de recuperação de informação [18], [19].

Nas próximas seções são descritos mais detalhes das três principais etapas constituintes do processo de Mineração de

Textos instanciadas para a extração e organização não supervisionada do conhecimento: Pré-processamento dos Documentos, Extração de Padrões com Agrupamento de Textos e Avaliação do Conhecimento (Figura 1).

A. *Pré-processamento de Documentos Textuais*

Na etapa de pré-processamento se encontra a principal diferença entre os processos de Mineração de Textos e processos de mineração de dados: a estruturação dos textos em um formato adequado para a extração de conhecimento. Muitos autores consideram essa etapa a que mais tempo consome durante todo o ciclo da Mineração de Textos. O objetivo do pré-processamento é extrair de textos escritos em língua natural, inerentemente não estruturados, uma representação estruturada, concisa e manipulável por algoritmos de agrupamento de textos. Para tal, são executadas atividades de tratamento e padronização na coleção de textos, seleção dos termos (palavras) mais significativos e, por fim, representação da coleção textual em uma formato estruturado que preserve as principais características dos dados [7].

Os documentos da coleção podem estar em diferentes formatos, uma vez que existem diversos aplicativos para apoiar a geração e publicação de textos eletrônicos. Dependendo de como os documentos foram armazenados ou gerados, há a necessidade de padronizar as formas em que se encontram; e, geralmente, os documentos são convertidos para o forma de texto plano sem formatação.

Um dos maiores desafios do processo de Mineração de Textos é a alta dimensionalidade dos dados. Uma pequena coleção de textos pode facilmente conter milhares de termos, muitos deles redundantes e desnecessários, que tornam lento o processo de extração de conhecimento e prejudicam a qualidade dos resultados.

A **seleção de termos** tenta solucionar esse desafio e tem o objetivo de obter um subconjunto conciso e representativo de termos da coleção textual. O primeiro passo é a eliminação de *stopwords*, que são os termos que nada acrescentam à representatividade da coleção ou que sozinhas nada significam, como artigos, pronomes e advérbios. O conjunto de *stopwords* é a *stoplist*[1]. Essa eliminação reduz significativamente a quantidade de termos diminuindo o custo computacional das próximas etapas [10]. Posteriormente, busca-se identificar as variações morfológicas e termos sinônimos. Para tal, pode-se, por exemplo, reduzir uma palavra à sua raiz por meio de processos de *stemming* ou mesmo usar dicionários ou *thesaurus*. Além disso, é possível buscar na coleção a formação de termos compostos, ou *n-gramas*, que são termos formados por mais de um elemento, porém com um único significado semântico [10], [20].

Outra forma de realizar a seleção de termos é avaliá-los por medidas estatísticas simples, como a frequência de termo, conhecida como TF (do inglês *term frequency*), e frequência de documentos, conhecida como DF (do inglês *document frequency*). A frequência de termo contabiliza a frequência absoluta de um determinado termo ao longo da coleção textual. A frequência de documentos, por sua vez, contabiliza o número de documentos em que um determinado termo aparece.

O método de Luhn [21] é uma técnica tradicional para seleção de termos utilizando a medida TF. Esse método foi baseado na Lei de Zipf [22], também conhecida como Princípio do Menor Esforço. Em textos, ao contabilizar a frequência dos termos e ordenar o histograma resultante em ordem decrescente, forma-se a chamada Curva de Zipf, na qual o k-ésimo termo mais comum ocorre com frequência inversamente proporcional a k. Os termos de alta frequência são julgados não relevantes por geralmente aparecerem na grande maioria dos textos, não trazendo, em geral, informações úteis. Já os termos de baixa frequência são considerados muito raros e não possuem caráter discriminatório. Assim, são traçados pontos de corte superior e inferior da Curva de Zipf, de maneira que termos com alta e baixa frequência são descartados, considerando os termos mais significativos os de frequência intermediária (Figura 2).

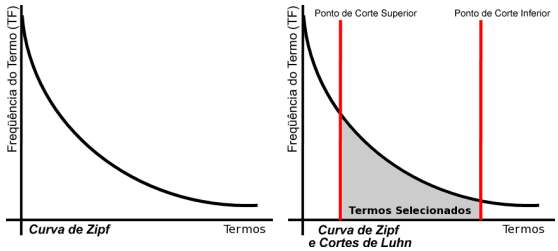

Figura 2. Método de Luhn para seleção de termos (adaptado de [23])

Dado o baixo processamento demandado por esse método, ele é facilmente escalável para coleções textuais muito grandes [23]. Entretanto, os pontos de corte superior e inferior não são exatos, sendo a subjetividade da escolha desses pontos a principal desvantagem do método.

Uma vez selecionado os termos mais representativos da coleção textual, deve-se buscar a **estruturação dos documentos**, de maneira a torná-los processáveis pelos algoritmos de agrupamento que são utilizados para o agrupamento de textos. O modelo mais utilizado para representação de dados textuais é o modelo espaço-vetorial, no qual cada documento é um vetor em um espaço multidimensional, e cada dimensão é um termo da coleção [7]. Para tal, pode-se estruturar os textos em uma *bag-of-words*, na qual os termos são considerados independentes, formando um conjunto desordenado em que a ordem de ocorrência das palavras não importa. A *bag-of-words* é uma tabela documento-termo, como ilustrado na Tabela I na qual d_i corresponde ao i-ésimo documento, t_j representa o *j*-ésimo termo e a_{ij} é um valor que relaciona o *i*-ésimo documento com o *j*-ésimo termo. Observe que na representação aqui apresentada não há informação de classe, uma vez que a tarefa de aprendizado com métodos de agrupamento é não supervisionado.

Por meio da tabela documento-termo, cada documento pode ser representado como um vetor $\vec{d_i} = (a_{i1}, a_{i2}, \ldots, a_{iM})$. Geralmente, o valor da medida a_{ij} é obtido de duas formas:

- um valor que indica se um determinado termo está presente ou não em um dado documento; e

[1]Os arquivos de *stopwords* para a língua portuguesa e inglesa podem ser obtidos em http://sites.labic.icmc.usp.br/marcacini/ihtc

	t_1	t_2	...	t_M
d_1	a_{11}	a_{12}	...	a_{1M}
d_2	a_{21}	a_{22}	...	a_{2M}
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
d_N	a_{N1}	a_{N2}	...	a_{NM}

Tabela I
TABELA DOCUMENTO-TERMO: REPRESENTAÇÃO DA MATRIZ ATRIBUTO×VALOR

- um valor que indica a importância ou distribuição do termo ao longo da coleção de documentos, por exemplo, o valor de TF. Outras formas, baseadas em critérios de ponderação e normalização, podem ser encontradas em [24] e [25]. Entre elas, destaca-se o critério TF-IDF (*Term Frequency Inverse Document Frequency*), que leva em consideração tanto o valor de TF quanto o valor de DF [26].

A representação por meio da tabela documento-termo permite o emprego de um grande leque de algoritmos de agrupamento de textos, além de outras técnicas de extração de conhecimento. Deve-se ressaltar que essa etapa de Pré-Processamento pode ser redefinida e então repetida após as próximas etapas, uma vez que a descoberta de alguns padrões pode levar a estabelecer melhorias a serem empregadas sobre a tabela documento-termo, como, ponderar a importância de cada termo ou até mesmo refinar a seleção dos termos [27].

Outras técnicas de pré-processamento para dados textuais, incluindo diferentes critérios não supervisionados de seleção de termos estão descritas em [28]. Uma ferramenta computacional de uso geral que disponibiliza diversos algoritmos de pré-processamento de textos é a PreText [23][2], que possui suporte a *stemming* para textos em Português, Inglês e Espanhol. Para coleções textuais formadas por artigos científicos pode ser utilizada a ferramenta *IESystem - Information Extraction System*[3], que permite identificar e extrair parte específicas dos artigos, como títulos, resumo e autores.

B. Extração de Padrões usando Agrupamento de Textos

Com o objetivo de realizar a extração de padrões, após a representação dos textos em um formato estruturado, utiliza-se métodos de agrupamento de textos para obter a organização dos documentos.

Em tarefas de agrupamento, o objetivo é organizar um conjunto de objetos em grupos, baseado em uma medida de proximidade, na qual objetos de um mesmo grupo são altamente similares entre si, mas dissimilares em relação aos objetos de outros grupos [8]. Em outras palavras, o agrupamento é baseado no princípio de maximizar a similaridade interna dos grupos (intragrupo) e minimizar a similaridade entre os grupos (intergrupos) [8]. A análise de agrupamento também é conhecida como aprendizado por observação em análise exploratória dos dados, pois a organização dos objetos em grupos é realizada pela observação de regularidades nos dados, sem uso de conhecimento externo [9]. Assim, ao contrário de métodos supervisionados, como algoritmos de classificação, em processos de agrupamento não há classes ou rótulos predefinidos para treinamento de um modelo, ou seja, o aprendizado é realizado de forma não supervisionada [29], [5].

O processo de agrupamento depende de dois fatores principais: **1) uma medida de proximidade** e **2) uma estratégia de agrupamento** [9]. As medidas de proximidade determinam como a similaridade entre dois objetos é calculada. Sua escolha influencia a forma como os grupos são obtidos e depende dos tipos de variáveis ou atributos que representam os objetos. As estratégias de agrupamento são os métodos e algoritmos para definição dos grupos. Em geral, pode-se classificar os algoritmos de agrupamento em métodos particionais e métodos hierárquicos. Um terceiro fator, **3) Seleção de descritores para o agrupamento** também é importante para a etapa de extração de padrões, pois é desejável encontrar descritores que indicam o significado do agrupamento obtido para os usuários.

1) Medidas de Proximidade: a escolha da medida de proximidade para calcular o quão similar são dois objetos é fundamental para a análise de agrupamentos. Essa escolha depende das características do conjunto de dados, principalmente dos tipos e escala dos dados. Assim, existem medidas de proximidade para dados contínuos, discretos e mistura entre dados contínuos e discretos. As medidas de proximidade podem calcular tanto a similaridade quanto dissimilaridade (ou distância) entre objetos. No entanto, as medidas de similaridades podem ser, geralmente, convertidas para medidas de dissimilaridade, e vice-versa.

A seguir, serão descritas duas medidas de similaridade comumente utilizadas em dados textuais: Cosseno e Jaccard. Para tal, considere dois documentos $x_i = (x_{i1}, x_{i2}, ..., x_{im})$ e $x_j = (x_{j1}, x_{j2}, ..., x_{jm})$, representados no espaço vetorial *m*-dimensional, no qual cada termo da coleção representa uma dessas dimensões.

A medida de similaridade **Cosseno** é definida de acordo com ângulo cosseno formado entre os vetores de dois documentos, conforme a Equação 1 [30], [7].

$$cosseno(x_i, x_j) = \frac{x_i \bullet x_j}{\mid x_i \mid\mid x_j \mid} = \frac{\sum_{l=1}^{m} x_{il} x_{jl}}{\sqrt{\sum_{l=1}^{m} {x_{il}}^2} \sqrt{\sum_{l=1}^{m} {x_{jl}}^2}} \quad (1)$$

Assim, se o valor da medida de similaridade Cosseno é 0, o ângulo entre x_i e x_j é $90°$, ou seja, os documentos não compartilham nenhum termo. Por outro lado, se o valor da similaridade for próximo de 1, o ângulo entre x_i e x_j é próximo de $0°$, indicando que os documentos compartilham termos e são similares. É importante observar que essa medida não considera a magnitude dos dados para computar a proximidade entre documentos.

Em algumas situações os vetores são representados por valores binários, ou seja, indicam a presença ou ausência de algum termo. O cálculo da proximidade entre dois documentos representados por vetores binários pode ser realizado pela

[2]PreText: ferramenta para pré-processamento de textos disponível em http://labic.icmc.usp.br/software-and-application-tools

[3]IESystem - Information Extraction System: disponível em http://labic.icmc.usp.br/software-and-application-tools

medida **Jaccard**. Seja x_i e x_j dois documentos, a medida Jaccard pode ser derivada a partir das seguintes contagens:

- f_{11} = número de termos presentes em ambos documentos;
- f_{01} = número de termos ausentes em x_i e presentes em x_j; e
- f_{10} = número de termos presentes em x_i e ausentes x_j.

A partir das contagens, a medida Jaccard é definida na Equação 2 [30], [7].

$$jaccard(x_i, x_j) = \frac{f_{11}}{f_{01} + f_{10} + f_{11}} \quad (2)$$

O valor da medida Jaccard fica no intervalo $[0, 1]$. Quanto mais próximo de 1 maior a similaridade entre os dois documentos.

Pode-se observar que as medidas Cosseno e Jaccard são medidas de similaridade. Conforme comentado anteriormente, as medidas de similaridade podem ser transformadas em medidas de dissimilaridade (ou distância). Na Equação 3 e na Equação 4 são definidas medidas de dissimilaridade baseada na Cosseno e Jaccard, respectivamente.

$$d_{cos}(x_i, x_j) = 1 - cosseno(x_i, x_j) \quad (3)$$

$$d_{jac}(x_i, x_j) = 1 - jaccard(x_i, x_j) \quad (4)$$

A literatura apresenta uma variedade de medidas de proximidades. Nessa seção, foram apresentadas duas medidas relacionadas ao contexto deste artigo. Uma revisão mais extensa está disponível em [8] e [30].

Uma dúvida pertinente que surge com relação às várias medidas de proximidade é qual escolher no processo de agrupamento. Não existe uma regra geral para essa escolha. Geralmente, essa decisão é baseada de acordo com a natureza dos dados a serem analisados, acompanhada de um processo de validação da qualidade do agrupamento obtido.

2) Métodos de Agrupamento: após a escolha de uma medida de proximidade para os documentos, é selecionado um método para o agrupamento. Os métodos de agrupamento podem ser classificados considerando diferentes aspectos. Em [29], os autores organizam os métodos de agrupamento de acordo com a estratégia adotada para definir os grupos. Uma análise de diferentes métodos de agrupamento considerando o cenário de Mineração de Dados é apresentada em [31].

Em geral, as estratégias de agrupamento podem ser organizadas em dois tipos: agrupamento particional e agrupamento hierárquico. No **agrupamento particional** a coleção de documentos é dividida em uma partição simples de k grupos, enquanto no **agrupamento hierárquico** é produzido uma sequência de partições aninhadas, ou seja, a coleção textual é organizada em grupos e subgrupos de documentos [7]. Além disso, o agrupamento obtido pode conter sobreposição, isto é, quando um documento pertence a mais de um grupo ou, até mesmo, quando cada documento possui um grau de pertinência associado aos grupos. No contexto deste trabalho, são discutidas com mais detalhes as estratégias que produzem agrupamento sem sobreposição, também conhecidas como estratégias rígidas ou *crisp* [8]. Assim, se o conjunto $X = \{x_1, x_2, ..., x_n\}$ representa uma coleção de n documentos, uma partição rígida $P = \{G_1, G_2, ..., G_k\}$ com k grupos não sobrepostos é tal que:

- $G_1 \cup G_2 \cup ... \cup G_k = X$;
- $G_i \neq \emptyset$ para todo $i \in \{1, 2, ..., k\}$; e
- $G_i \cap G_j = \emptyset$ para todo $i \neq j$.

As diversas estratégias de agrupamento são, na prática, algoritmos que buscam uma solução aproximada para o problema de agrupamento. Para exemplificar, um algoritmo de força bruta que busca a melhor partição de um conjunto de n documentos em k grupos, precisa avaliar $k^n/k!$ possíveis partições [32]. Enumerar e avaliar todas as possíveis partições é inviável computacionalmente. A seguir, são descritos alguns dos principais algoritmos que são utilizados para agrupamento de documentos.

Agrupamento Particional: o agrupamento particional também é conhecido como agrupamento por otimização. O objetivo é dividir iterativamente o conjunto de objetos em *k* grupos, na qual *k* geralmente é um valor informado previamente pelo usuário. Os grupos de documentos são formados visando otimizar a compactação e/ou separação do agrupamento.

O algoritmo *k-means* [33] é o representante mais conhecido para agrupamento particional e muito utilizado em coleções textuais [34]. No *k-means* utiliza-se um representante de grupo denominado centroide, que é simplesmente um vetor médio computado a partir dos demais vetores do grupo. A Equação 5 define o cálculo do centroide C para um determinado grupo G, em que x representa um documento pertencente a G e o número total de documentos no grupo é $|G|$.

$$C = \frac{1}{|G|} \sum_{x \in G} x \quad (5)$$

Dessa forma, o centroide mantém um conjunto de características centrais do grupo, permitindo representar todos os documentos que pertencem a este grupo. Ainda, é importante observar que o *k-means* só é aplicável em situações na qual a média possa ser calculada.

O pseudocódigo para o *k-means*, contextualizado para agrupamento de documentos, está descrito no Algoritmo 1.

Algoritmo 1: O algoritmo k-means

Entrada:
$X = \{x_1, x_2, ..., x_n\}$: conjunto de documentos
k: número de grupos

Saída:
$P = \{G_1, G_2, ..., G_k\}$: partição com k grupos

```
1 selecionar aleatoriamente k documentos como centróides iniciais;
2 repita
3     para cada documento x ∈ X faça
4         computar a (dis)similaridade de x para cada centroide C ;
5         atribuir x ao centroide mais próximo ;
6     fim
7     recomputar o centroide de cada grupo;
8 até atingir um critério de parada;
```

O critério de parada do *k-means* é dado quando não ocorre mais alterações no agrupamento, ou seja, a solução converge para uma determinada partição. Outro critério de parada pode ser um número máximo de iterações.

Durante as iterações do *k-means*, o objetivo é minimizar uma função de erro E, definida na Equação 6, em que x é um documento da coleção; e C_i é o centroide do grupo G_i. Observe que é utilizado uma medida de dissimilaridade $dis(x, C_i)$ para calcular o valor da função de erro E.

$$E = \sum_{i=1}^{k} \sum_{x \in G_i} \mid dis(x, C_i) \mid^2 \quad (6)$$

Ao minimizar este critério, o *k-means* tenta separar o conjunto de documentos diminuindo a variabilidade interna de cada grupo e, consequentemente, aumentando a separação entre os grupos.

A complexidade do *k-means* é linear em relação ao número de objetos, o que possibilita sua aplicação eficiente em diversos cenários. No entanto, a necessidade de informar com antecedência o número de grupos pode ser vista como uma desvantagem, pois esse valor geralmente é desconhecido pelos usuários. Além disso, o método apresenta variabilidade nos resultados, pois a seleção dos centróides iniciais afeta o resultado do agrupamento. Para minimizar esse efeito, o algoritmo é executado diversas vezes, com várias inicializações diferentes, e a solução que apresenta menor valor de erro E é selecionada.

Agrupamento Hierárquico: os algoritmos de agrupamento hierárquico podem ser aglomerativos ou divisivos. No **agrupamento hierárquico aglomerativo**, inicialmente cada documento é um grupo e, em cada iteração, os pares de grupos mais próximos são unidos até se formar um único grupo [7]. Já no **agrupamento hierárquico divisivo**, inicia-se com um grupo contendo todos os documentos que é, então, dividido em grupos menores até restarem grupos unitários (grupo com apenas um documento) [34], [35].

Tanto os métodos aglomerativos quanto os divisivos organizam os resultados do agrupamento em uma árvore binária conhecida como dendrograma (Figura 3). Essa representação é uma forma intuitiva de visualizar e descrever a sequência do agrupamento. Cada nó do dendrograma representa um grupo de documentos. A altura dos arcos que unem dois subgrupos indica o grau de compactação do grupo formado por eles. Quanto menor a altura, mais compactos são os grupos. No entanto, também se espera que os grupos formados sejam distantes entre si, ou seja, que a proximidade de objetos em grupos distintos seja a menor possível. Essa característica é representada quando existe uma grande diferença entre a altura de um arco e os arcos formados abaixo dele [36].

A partir do dendrograma também é possível obter uma partição com um determinado número de grupos, como nos métodos particionais. Por exemplo, a linha tracejada na Figura 3 indica uma partição com dois grupos de documentos: $\{x_1, x_2, x_3, x_4\}$ e $\{x_5, x_6, x_7\}$.

O pseudocódigo para um algoritmo típico de agrupamento hierárquico aglomerativo está descrito no Algoritmo 2.

A diferença principal entre os algoritmos de agrupamento hierárquico aglomerativo está no critério de seleção do par

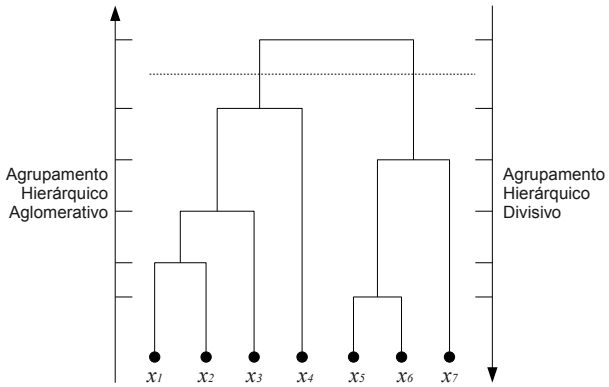

Figura 3. Exemplo de um dendrograma (adaptado de [9])

Algoritmo 2: Agrupamento hierárquico aglomerativo

Entrada:
$X = \{x_1, x_2, ..., x_n\}$: conjunto de documentos
Saída:
$S = \{P_1, ..., P_k\}$: lista de agrupamentos formados

1 fazer com que cada documento $x \in X$ seja um grupo;
2 computar a dissimilaridade entre todos os pares distintos de grupos;
3 **repita**
4 | selecionar o par de grupos mais próximo;
5 | unir os dois grupos para formar um novo grupo G;
6 | computar a dissimilaridade entre G e os outros grupos;
7 **até** *obter um único grupo com todos os documentos*;

de grupos mais próximo (Linha 4 do Algoritmo 2). Os três critérios mais conhecidos são:

- *Single-Link* [8], [37]: utiliza o critério de vizinho mais próximo, no qual a distância entre dois grupos é determinada pela distância do par de documentos mais próximos, sendo cada documento pertencente a um desses grupos. Esse método de união de grupos apresenta um problema conhecido como "efeito da corrente", em que ocorre a união indevida de grupos influenciada pela presença de ruídos na base de dados [38];
- *Complete-Link* [8], [39]: utiliza o critério de vizinho mais distante, ao contrário do algoritmo *Single-Link*, e a distância entre dois grupos é maior distância entre um par de documentos, sendo cada documento pertencente a um grupo distinto. Esse método dificulta a formação do efeito da corrente, como ocorre no *Single-Link*, e tende a formar grupos mais compactos e em formatos esféricos [38]; e
- *Average-Link* [8], [40]: a distância entre dois grupos é definida como a média das distâncias entre todos os pares de documentos em cada grupo, cada par composto por um documento de cada grupo. Esse método elimina muitos problemas relacionados à dependência do tamanho dos grupos, mantendo próxima a variabilidade interna entre eles [38].

A escolha do critério de união de grupos dos algoritmos aglomerativos depende geralmente do conjunto de dados e dos objetivos da aplicação. Por exemplo, em dados textuais, avaliações experimentais têm mostrado o *Average-Link* como a melhor opção entre os algoritmos que adotam estratégias aglomerativas [41].

A maioria dos trabalhos relacionados com agrupamento hierárquico na literatura referenciam as estratégias aglomerativas, mostrando pouco interesse nas estratégias divisivas. A possível causa é a complexidade das estratégias divisivas, que cresce exponencialmente em relação ao tamanho do conjunto de dados, proibindo sua aplicação em conjuntos de dados grandes [42]. Para lidar com esse problema, [34] propuseram o algoritmo *Bisecting k-means*, que basicamente utiliza agrupamento particional baseado no *k-means* sucessivamente, possibilitando sua aplicação em conjuntos de dados maiores, inclusive em coleções textuais. O pseudocódigo do *Bisecting k-means* está ilustrado no Algoritmo 3.

Algoritmo 3: Bisecting k-means

Entrada:
$X = \{x_1, x_2, ..., x_n\}$: conjunto de documentos
Saída:
$S = \{P_1, ..., P_k\}$: lista de agrupamentos formados

1 formar um agrupamento contendo todos os documentos de X;
2 **repita**
3 selecionar o próximo grupo (nó folha) a ser dividido;
4 dividir este grupo em dois novos subgrupos *(k-means com k=2)*;
5 **até** *obter apenas grupos unitários*;

Existem diferentes maneiras para seleção do próximo grupo a ser dividido (Linha 3 do Algoritmo 3). Um critério simples e eficaz é selecionar o maior grupo (de acordo com o número de documentos) ainda não dividido em uma iteração anterior [41]. Uma propriedade interessante do *Bisecting k-means* é o fato de ser menos sensível à escolha inicial dos centroides quando comparado com o *k-means* [30].

Os algoritmos de agrupamento hierárquico aglomerativos e divisivos apresentam complexidade quadrática de tempo e espaço, em relação ao número de documentos. Avaliações experimentais indicam que o *Bisecting k-means* obtém melhores resultados em coleções de documentos, seguido do agrupamento hierárquico aglomerativo com o critério *Average-Link* [41], [35].

Existem várias ferramentas que disponibilizam algoritmos de agrupamento de textos. Entre os mais relevantes, pode-se citar o *Cluto - Clustering Toolkit* [4], que possui suportes ao principais algoritmos de agrupamento da literatura, diversas medidas de similaridade, além de interfaces gráficas para análise dos resultados. É importante observar que, devido a grande aplicação de algoritmos de agrupamento, diversos softwares matemáticos e estatísticos, como o *Matlab*[5] e *R*[6], também disponibilizam algoritmos de agrupamento que podem ser empregados em tarefas de extração de padrões.

Outros Métodos de Agrupamento: até o momento, foram apresentados os algoritmos de agrupamento mais conhecidos e geralmente utilizados na organização de coleções textuais. No entanto, é importante observar que a literatura na área de análise de agrupamento de dados apresenta uma vasta contribuição [9]. A seguir, é realizado uma breve descrição de outros métodos de agrupamento existentes.

- **Agrupamento baseado em densidade:** esses métodos assumem que os grupos são regiões de alta densidade separados por regiões de baixa densidade no espaço dimensional formado pelos atributos dos objetos. A ideia básica é que cada objeto possui um número mínimo de vizinhos dentro de uma esfera com raio pré-definido pelo usuário. Se a esfera contém um número mínimo de objetos, então é considerada uma região com densidade e é utilizada para formação do agrupamento. O algoritmo *DBSCAN* [43] é um exemplo de algoritmo de agrupamento baseado em densidade.
- **Agrupamento baseado em grade:** o diferencial desses métodos é o uso de uma grade para construir um novo espaço aos objetos de forma que todas as operações de agrupamento sejam realizadas em termos do espaço da grade. É uma abordagem eficiente para grandes conjuntos de dados, alta dimensionalidade e para detecção de ruídos. Um algoritmo que realiza o agrupamento baseado em grade é o *CLIQUE* [44].
- **Agrupamento com sobreposição:** os algoritmos mencionados no decorrer deste capítulo obtém grupos exclusivos, ou seja, cada objeto pertence exclusivamente a um único grupo. Os métodos que permitem sobreposição podem associar os objetos a um ou mais grupos. Essa sobreposição pode ser simples, na qual um objeto está em um ou mais grupos ou, ainda, pertencer a todos os grupos com um grau de pertinência/probabilidade. Os algoritmos de agrupamento *fuzzy* e probabilísticos associam níveis de pertinência ou probabilidade dos objetos aos grupos encontrados [45].
- **Agrupamento baseado em redes auto-organizáveis:** também conhecidos como redes *SOM - Self Organizing Map* [46], esses métodos utilizam o conceito de redes neurais para realizar o agrupamento dos dados. A ideia básica é organizar um conjunto de neurônios em um reticulado bidimensional, na qual cada neurônio fica conectado em todas as entradas da rede. Conforme os objetos são apresentados à rede, os neurônios atualizam seus pesos de ligação da rede, ativando uma região diferente do reticulado. No final do processo, cada região do reticulado representa um grupo de objetos. O algoritmo *SomPak* [47] é um exemplo de agrupamento baseado em redes SOM.

3) Seleção de Descritores para Agrupamento: uma vez obtido o agrupamento (particional ou hierárquico) de doc-

[4]Cluto - Clustering Toolkit: http://glaros.dtc.umn.edu/gkhome/views/cluto

[5]Matlab - http://www.mathworks.com/products/matlab/

[6]R Project - http://www.r-project.org/

umentos, deve-se selecionar descritores que auxiliam a interpretação dos resultados. Essa é uma tarefa importante, pois o agrupamento geralmente é utilizado em atividades exploratórias para descoberta de conhecimento e, assim, é necessário indicar o significado de cada grupo para que usuários e/ou aplicações possam interagir com o agrupamento de forma mais intuitiva [10].

Conforme comentado anteriormente, o centroide mantém o conjunto de características centrais do grupo, permitindo representar todos os documentos pertencentes a este grupo. Por este motivo, algumas técnicas utilizam o centroide como ponto de partida para seleção dos descritores de um grupo. Uma estratégia simplista é selecionar os termos mais frequentes de um grupo, porém, a literatura indica que os resultados obtidos por essa estratégia não são satisfatórios [48]. Outra estratégia é selecionar os termos dos j documentos mais próximos ao centroide como descritores [15]. Em [10] é discutido que as técnicas existentes de seleção de atributos em tarefas de aprendizado de máquina podem ser aplicadas na seleção de descritores de agrupamento. Assim, é possível obter um *ranking* dos termos que melhor discriminam um determinado grupo. Abaixo, é descrito uma abordagem genérica para obtenção deste *ranking*.

Seja um grupo G e seu respectivo centroide C. O conjunto de termos que compõem o centroide C é identificado como T, ou seja, os termos que representam as dimensões no espaço vetorial. A ideia básica é obter uma lista ordenada (*ranking*) dos termos contidos em T e selecionar os melhores j termos como descritores do grupo G. Diversos critérios podem ser utilizados para construção do *ranking*, e esses critérios podem ser derivados a partir de uma tabela de contingência.

Assim, para cada termo $t \in T$, realiza-se uma expressão de busca $Q(t)$ sobre toda a coleção de documentos $X = \{x_1, x_2, ..., x_n\}$, recuperando-se um subconjunto de documentos que contêm o termo t. Com o conjunto de documentos recuperados $Q(t)$ e o conjunto de documentos do grupo G, é construído uma tabela de contingência do termo t, conforme ilustrado na Figura 4.

Q(t) \ G	Relevante	Não Relevante
Relevante	*acertos*	*ruído*
Não Relevante	*perda*	*rejeitos*

Figura 4. Tabela de contingência com os possíveis resultados de recuperação por meio da expressão de busca Q(t).

Os itens da tabela de contingência são calculadas da seguinte forma [49]:

- *acertos*: número de documentos recuperados por $Q(t)$ que pertencem a G;
- *perda*: número de documentos em G que não foram recuperados por $Q(t)$;
- *ruído*: número de documentos recuperados por $Q(t)$ que não pertencem a G; e
- *rejeitos*: número de documentos que não pertencem a G e que também não foram recuperados por $Q(t)$.

A partir desses itens, pode-se derivar diversos critérios para avaliar o poder discriminativo do termo t para o grupo G. Um desses critérios é o F-Measure (F_1), descrito na Equação 9, obtido por uma média harmônica entre precisão (Equação 7) e revocação (Equação 8).

$$Precisao(t) = \frac{acertos}{acertos + ruido} \tag{7}$$

$$Revocacao(t) = \frac{acertos}{acertos + perda} \tag{8}$$

$$F - Measure(t) = \frac{2 * Precisao(t) * Revocacao(t)}{Precisao(t) + Revocacao(t)} \tag{9}$$

O valor de F-Measure varia no intervalo $[0, 1]$, e quanto mais próximo de 1, melhor o poder discriminativo de t. O processo é aplicado para todos os termos $t \in T$, e um *ranking* é obtido em ordem decrescente da F-Measure, por exemplo. Dessa forma, os descritores do grupo G são formados pelos melhores j termos. Uma lista de critérios que podem ser derivadas a partir dos itens da tabela de contingência pode ser encontrada em [50].

A seleção de descritores ilustrada aqui é aplicável em agrupamentos particionais e hierárquicos. No entanto, o agrupamento hierárquico tem certas particularidades, pois documentos de um grupo (filho) também estão presentes em seu grupo superior (grupo pai). Nesse sentido, pode-se refinar a seleção dos termos considerando a estrutura hierárquica [51], [17].

C. Avaliação do Conhecimento

A Avaliação do Conhecimento pode ser realizada de forma **subjetiva**, utilizando um conhecimento de um especialista de domínio, ou de forma **objetiva** por meio de índices estatísticos que indicam a qualidade dos resultados. Nesta seção, serão abordados alguns desses índices de validação.

A qualidade da organização dos documentos está diretamente relacionada com a qualidade do agrupamento na extração de padrões. Assim, a validação do conhecimento extraído é realizada por meio de índices utilizados na análise de agrupamentos.

A validação do resultado de um agrupamento, em geral, é realizada por meio de índices estatísticos que expressam o "mérito" das estruturas encontradas, ou seja, quantifica alguma informação sobre a qualidade de um agrupamento [52], [9]. O uso de técnicas de validação em resultados de agrupamento é uma atividade importante, uma vez que algoritmos de agrupamento sempre encontram grupos nos dados, independentemente de serem reais ou não [53].

Em geral, existem três tipos de critérios para realizar a validação de um agrupamento: critérios internos, relativos e externos [8].

Os **critérios internos** obtêm a qualidade de um agrupamento a partir de informações do próprio conjunto de dados. Geralmente, um critério interno analisa se as posições dos

objetos em um agrupamento obtido corresponde à matriz de proximidades. Já os **critérios relativos** comparam diversos agrupamentos para decidir qual deles é o mais adequado aos dados. Finalmente, os **critérios externos** avaliam um agrupamento de acordo com uma informação externa, geralmente uma intuição do pesquisador sobre a estrutura presente nos dados ou um agrupamento construído por um especialista de domínio. Por exemplo, um critério externo pode medir se o agrupamento obtido corresponde com uma parte dos dados já agrupados manualmente.

Alguns trabalhos na literatura descrevem e comparam técnicas e índices de validação. No trabalho de [54], trinta índices de validação são comparados na tarefa de estimar o número de grupos em conjuntos de dados. Uma avaliação similar é realizada por [55], com uma comparação de diversos índices de validade relativa de agrupamento. Uma revisão geral de diversas abordagens para validação de agrupamento é encontrada em [56], [53], [9].

A seguir, serão discutidos três diferentes índices de validação que avaliam a qualidade de agrupamento sobre diferentes perspectivas.

1) Silhueta: a Silhueta [42], [30] é um índice de critério relativo utilizado para avaliar partições. Experimentos recentes comparando vários índices de validade relativa indicaram que a Silhueta é um dos índices de validade mais eficazes [55].

A medida de Silhueta verifica o quão bem os documentos estão situados dentro de seus grupos. Dada uma coleção de n documentos $X = \{x_1, x_2, ..., x_n\}$, o valor de Silhueta $s(x_i)$ do documento x_i é obtido pela Equação 10.

$$s(x_i) = \frac{b(x_i) - a(x_i)}{max\{a(x_i), b(x_i)\}} \tag{10}$$

no qual, $a(x_i)$ é a dissimilaridade média entre x_i e todos os documentos de seu grupo; e $b(x_i)$ a dissimilaridade média entre x_i e todos os documentos do seu grupo vizinho.

O valor do índice de Silhueta fica no intervalo $[-1, 1]$, em que valores positivos indicam que o documento está bem alocado no seu grupo e valores negativos indicam que o documento possivelmente está erroneamente agrupado. Com os valores de Silhueta de cada documento, calcula-se o valor global de Silhueta de uma partição (S_P), por meio da média das Silhuetas, conforme ilustrado na Equação 11.

$$S_P = \frac{\sum_{i=1}^{n} s(x_i)}{n} \tag{11}$$

O índice de Silhueta permite comparar partições obtidas por diferentes algoritmos de agrupamento e diferentes números de grupos. Assim, fixado um conjunto de dados, a Silhueta é uma medida útil para determinar o melhor número de grupos, ou seja, auxilia a estimação de parâmetros de algoritmos de agrupamento.

2) Entropia: o índice de Entropia [34], [30] é empregado como um critério externo de validação, ou seja, utiliza um conhecimento prévio (informação externa) a respeito das categorias ou tópicos dos documentos. A ideia é medir a desordem no interior de cada grupo. Assim, quanto menor o valor de Entropia melhor a qualidade do agrupamento. Um grupo com valor 0 (zero) de Entropia, que é a solução ideal, contém todos os documentos de um mesmo tipo de categoria ou tópico.

Para computar este índice considere que:

- P é uma partição obtida por um determinado algoritmo de agrupamento;
- L_r é uma determinada categoria (informação externa) representando um conjunto de documentos de um mesmo tópico ou classe; e
- G_i é um determinado grupo, e seu respectivo conjunto de documentos, pertencente à partição P.

Assim, a Entropia E de um determinado grupo G_i em relação a uma coleção textual com c categorias é calculada conforme a Equação 12.

$$E(G_i) = -\sum_{r=1}^{c}(\frac{|L_r \cap G_i|}{|G_i|})log(\frac{|L_r \cap G_i|}{|G_i|}) \tag{12}$$

Na Equação 12, $|L_r \cap G_i|$ representa o número de documentos do grupo G_i pertencentes à categoria L_r.

O valor de Entropia global do agrupamento (partição P) é calculado como a soma das entropias de cada grupo ponderada pelo tamanho de cada grupo (Equação 13).

$$Entropia(P) = \sum_{i=1}^{k} \frac{|G_i| * E(G_i)}{n} \tag{13}$$

Na Equação 13, k é o número de grupos na partição P e n é o número de documentos da coleção.

3) FScore: o índice FScore é uma medida que utiliza as ideias de *precisão* e *revocação*, da recuperação de informação, para avaliar a eficácia de recuperação em agrupamentos hierárquico de documentos [57], [41], [30]. É empregada como critério externo de validação, pois utiliza o conhecimento prévio (informação externa) sobre categorias ou tópicos existentes no conjunto de dados. A ideia básica é verificar o quanto o agrupamento hierárquico conseguiu reconstruir a informação de categoria associada a cada documento.

Para o cálculo do índice FScore, considere que:

- H é um agrupamento hierárquico obtido (dendrograma) por um determinado algoritmo;
- L_r é uma determinada categoria (informação externa) representando um conjunto de documentos de um mesmo tópico ou classe; e
- G_i é um determinado grupo, e seu respectivo conjunto de documentos, pertencente ao agrupamento hierárquico H.

Assim, dada uma categoria L_r e um grupo G_i, calcula-se as medidas de precisão P e revocação R conforme a Equação 14 e Equação 15, respectivamente. Em seguida, é obtida a média harmônica F (Equação 16), que representa um balanceamento entre a precisão e revocação.

$$P(L_r, G_i) = \frac{|L_r \cap G_i|}{|G_i|} \tag{14}$$

$$R(L_r, G_i) = \frac{|L_r \cap G_i|}{|L_i|} \tag{15}$$

$$F(L_r, G_i) = \frac{2 * P(L_r, G_i) * R(L_r, G_i)}{P(L_r, G_i) + R(L_r, G_i)} \tag{16}$$

A medida F selecionada para uma determinada categoria L_r é o maior valor obtido por algum grupo da hierarquia *H*, conforme a Equação 17.

$$F(L_r) = \max_{G_i \in H} F(L_r, G_i) \quad (17)$$

Finalmente, o valor FScore global de um agrupamento hierárquico com *n* documentos e *c* categorias, é calculado como o somatório da medida F de cada categoria ponderada pelo número de documentos da categoria (Equação 18).

$$FScore = \sum_{r=1}^{c} \frac{|L_r|}{n} F(L_r) \quad (18)$$

Conforme o agrupamento hierárquico consegue reconstruir a informação das categorias predeterminadas de uma coleção, o valor de FScore se aproxima de 1. Caso contrário, a FScore tem valor 0. Observe que essa medida trata cada grupo da hierarquia como se fosse o resultado de uma consulta e cada categoria predefinida da coleção como o conjunto de documentos relevantes para essa consulta.

IV. Exemplos e Aplicações

A extração e organização não supervisionada de conhecimento pode ser utilizada em diferentes domínios e para várias funcionalidades. Uma aplicação que tem gerado bons resultados é a construção automática de hierarquias de tópicos, descrito na Seção IV-A, que pode, inclusive, ser considerada uma versão inicial para o aprendizado de ontologias de domínios. A organização de resultados de busca também é uma aplicação relevante e tem recebido grande atenção na literatura atualmente. Na Seção IV-B são apresentados exemplos de ferramentas e sistemas para organizar resultados provenientes de máquinas de busca. Já na Seção IV-C é discutido como um processo de Mineração de Textos pode auxiliar a extração de metadados de uma coleção textual de um domínio específico.

Esses exemplos e aplicações descritos a seguir são úteis em vários sistemas de recuperação de informação, como na construção de bibliotecas digitais, em tarefas de *webmining* e engenharia de documentos.

A. Hierarquias de Tópicos

A organização de coleções textuais por meio de hierarquias de tópicos é uma abordagem popular para gestão do conhecimento, em que os documentos são organizados em tópicos e subtópicos, e cada tópico contém documentos relacionados a um mesmo tema [7], [10], [11].

As hierarquias de tópicos desempenham um papel importante na recuperação de informação, principalmente em tarefas de busca exploratória. Nesse tipo de tarefa, o usuário geralmente tem pouco domínio sobre o tema de interesse, o que dificulta expressar o objetivo diretamente por meio de palavras-chave [58]. Assim, torna-se necessário disponibilizar previamente algumas opções para guiar o processo de busca da informação. Para tal, cada grupo possui um conjunto de descritores que definem um tópico e indicam o significado dos documentos ali agrupados.

A construção de hierarquias de tópicos de maneira supervisionada, a exemplo do *Dmoz - Open Directory Project*[7] e *Yahoo! Directory*[8], exige um grande esforço humano. Ainda, essa construção é limitada pela grande quantidade de documentos disponíveis e pela alta frequência de atualização. Desse modo, é de grande importância a investigação de métodos para automatizar a construção de hierarquias de tópicos. O uso da Mineração de Textos para extração e organização não supervisionada de conhecimento, conforme discutido neste artigo, permite a construção de hierarquias de tópicos de forma automática e sem conhecimento prévio a respeito dos documentos textuais.

O processo de construção de hierarquias de tópicos, também conhecido como aprendizado não supervisionado de hierarquias de tópicos, abrange as mesmas fases descritas ao longo deste artigo. Inicialmente, uma coleção de textos de um determinado domínio de conhecimento é selecionada e os documentos são pré-processados. Algoritmos de agrupamento hierárquico são aplicados na etapa de extração de padrões para obter uma organização hierárquica da coleção textual. Em seguida, são selecionados descritores para cada grupo da hierarquia. Um grupo de documentos e seu respectivo conjunto de descritores formam um tópico na coleção, descrevendo um determinado assunto ou tema identificado nos textos. Na avaliação do conhecimento, a qualidade do agrupamento hierárquico é analisada, assim como o desempenho dos descritores selecionados para recuperar os documentos dos grupos.

Um ambiente para aprendizado de hierarquias de tópicos está disponível na ferramenta *Torch - Topic Hierarchies* [59]. A ferramenta foi desenvolvida na linguagem Java, com auxílio de alguns componentes de software livre, disponibilizando em um só ambiente os seguintes recursos:

- Etapa de pré-processamento dos textos: estruturação dos textos, seleção de termos com base em cortes Luhn, remoção de stopwords, radicalização de termos com *stemming* disponível em língua portuguesa e inglesa.
- Etapa de extração de padrões: diversos algoritmos de agrupamento da literatura e módulos de seleção de descritores para agrupamento.
- Etapa de pós-processamento: medidas de validação de agrupamento (Silhueta, Entropia e FScore) e um módulo para exploração visual de hierarquias de tópicos.

A ferramenta Torch está disponível online em http://sites.labic.icmc.usp.br/marcacini/ihtc, publicamente para a comunidade e usuários interessados. Na Figura 5 é apresentada uma tela da ferramenta Torch para exploração visual de hierarquias de tópicos. Percebe-se à esquerda a hierarquia de tópicos (e subtópicos), com seus respectivos descritores, e uma visualização dos grupos de documentos ao centro. Do lado direito da imagem, são apresentados os documentos relacionados ao tópico selecionado com seu respectivo grupo.

Um dos desafios atuais é manter a hierarquia de tópicos atualizada em cenários envolvendo coleções de textos dinâmicas. Em [60], é apresentado e avaliado um algoritmo proposto

[7]Dmoz - Open Directory Project: http://www.dmoz.org/

[8]Yahoo Directory!: http://dir.yahoo.com/

para agrupamento incremental que permite o aprendizado com a inclusão de novos documentos sem necessidade de repetir todo o processo de Mineração de Textos. Em vista disso, a ferramenta Torch já disponibiliza algoritmos de agrupamento incremental, além de técnicas de pré-processamento de textos mais adequadas em cenários dinâmicos e mecanismos de visualização dos resultados.

B. Organização de Resultados de Busca

A organização de resultados de busca em grupos de temas específicos facilita a exploração das páginas retornadas por uma máquina de busca. Uma consulta na web retorna uma lista ordenada e extensa de possíveis resultados e, geralmente, apenas os primeiros resultados são analisados pelos usuários. Com a organização dos resultados, o usuário pode selecionar um subconjunto das páginas retornadas que são mais apropriadas à consulta de interesse.

Na ferramenta *Torch - Topic Hierarchies* foi desenvolvido um protótipo para explorar o uso de agrupamento na organização de resultados de busca. Na Figura 6 é ilustrada a organização dos resultados da busca a partir de uma consulta com o termo "linux". Nesta figura, os principais temas extraídos das páginas recuperadas são exibidos à esquerda, enquanto as páginas web de cada tema selecionado são exibidos à direita.

Nesse tipo de aplicação, os algoritmos de agrupamento empregados devem ter requisitos específicos. Entre eles, o agrupamento deve ser realizado em tempo real durante a consulta do usuário. Uma atenção especial deve ser dada para a identificação dos descritores para os grupos, uma vez que irão guiar a análise dos resultados de busca. Alguns trabalhos da literatura indicam que o uso de frases compartilhadas pelos documentos são úteis tanto para formação do agrupamento quanto para identificar os respectivos temas da organização. A ferramenta online Carrot2[9] é um exemplo de aplicação que utilizam frases compartilhadas para organização de resultados de busca.

C. Extração de Metadados

Um outro uso para os resultados do processo de Mineração de Textos aqui descrito é a identificação de metadados do tipo assunto. Quando se tem uma aproximação indicativa de um tema por meio de grupos de documentos e um conjunto de descritores, geralmente tem-se um conjunto de termos, simples ou compostos, que fornecem indicações do assunto ao qual aquele grupo de documentos corresponde. O assunto de um documento, geralmente, é composto por palavras-chaves e por alguma categoria pré-definidas ou classificadas como termos livres [61]. As pré-definições passam por instrumentos de biblioteconomia que definem vocabulários controlados, que tanto fazem referências às palavras-chaves quanto às categorias, de acordo com suas definições.

Entende-se um vocabulário controlado como um conjunto de termos validado e mantido por uma comunidade representativa de um domínio de conhecimento, que reconhecidamente detém o controle de novos termos e de suas regras de inclusão. Todos os termos dessa coleção devem ter definições não ambíguas, podendo ou não lhes ser associados também significados. A composição desse vocabulário corresponde a uma relação de termos, onde um termo pode ser representado por um conjunto de outros termos que tenham preferência de uso em relação a ele, sejam sinônimos ou termos similares. Esse vocabulário pode estar organizado sob um ***thesaurus***, definido como um vocabulário controlado que representa sinônimos e relacionamentos entre termos [13]; que são do tipo pai-filho, *broader-term* e *narrower-term*, e associativas, *related terms*.

Os termos livres são palavras-chaves ou categorias que não se incluem nos *thesaurus* já consagrados do domínio de conhecimento. Eles são identificados por sua importância estatística junto aos textos de um agrupamento automaticamente gerado ou subjetivamente atribuídos pelo autor ou bibliotecário que catalogou o documento. Esses termos livres são candidatos a serem incorporados a *thesaurus* como novos termos ou relações e, especialmente, auxiliam a recuperação desses documentos, aumentando a precisão ou a revocação de um processo de busca.

Assim, se uma coleção de documentos não passou por um processo de catalogação ou passou, mas este não foi satisfatório ou completo; ou seja, se seus metadados são desconhecidos ou incompletos, um caminho é utilizar os descritores dos agrupamentos gerados automaticamente como indicação dos componentes do metadado assunto. Pode-se incorporar um processo de reconhecimento automático ou semi-automático de elementos de *thesaurus*, ou de elementos próximos a estes, por meio de medidas de similaridade de palavras ou medidas estatísticas de correlação dos descritores com termos de um *thesaurus*. Também pode-se implementar um processo semi-automático de reconhecimento de palavras-chaves, categorias ou termos que possam ser incorporados a uma lista de *stopwords* do domínio trabalhado, escolhidos por um bibliotecário ou especialista no domínio de conhecimento.

Um processo como este último é implementado pela ferramenta TaxEdit [62], ainda em beta-teste, que vem sendo utilizada por um grupo de bibliotecários da Embrapa. A TaxEdit permite que se incorpore o tratamento de um vocabulário controlado, formatado como um *thesaurus* típico, em suas opções; porém, ainda de forma limitada ao uso desses vocábulos como os únicos atributos identificados no processo de Mineração de Textos. Assim, os bibliotecários podem utilizá-la com um *thesaurus* ou gerar os tópicos apenas com os termos estatisticamente significantes na coleção de documentos.

Um exemplo de uso da TaxEdit pode ser observado na Figura 7. Na hierarquia da esquerda não foi utilizado um *thesaurus*, foram gerados 18711 atributos estatisticamente importantes na coleção, porém sem o uso de filtros de seleção de atributos. Na hierarquia da direita utilizaram-se apenas os termos que faziam parte do *thesaurus* como vocabulário, um total de 13884 termos.

Note que os trechos das hierarquias selecionados na figura são bem próximos, compostos pelos mesmos conjuntos de documentos. Os documentos correspondem a *clippings* de jornal sobre cana-de-açúcar e etanol. Utilizando-se o *thesaurus* para controlar os termos gerados, temos que o segundo nó da

[9]Carrot2: http://search.carrot2.org

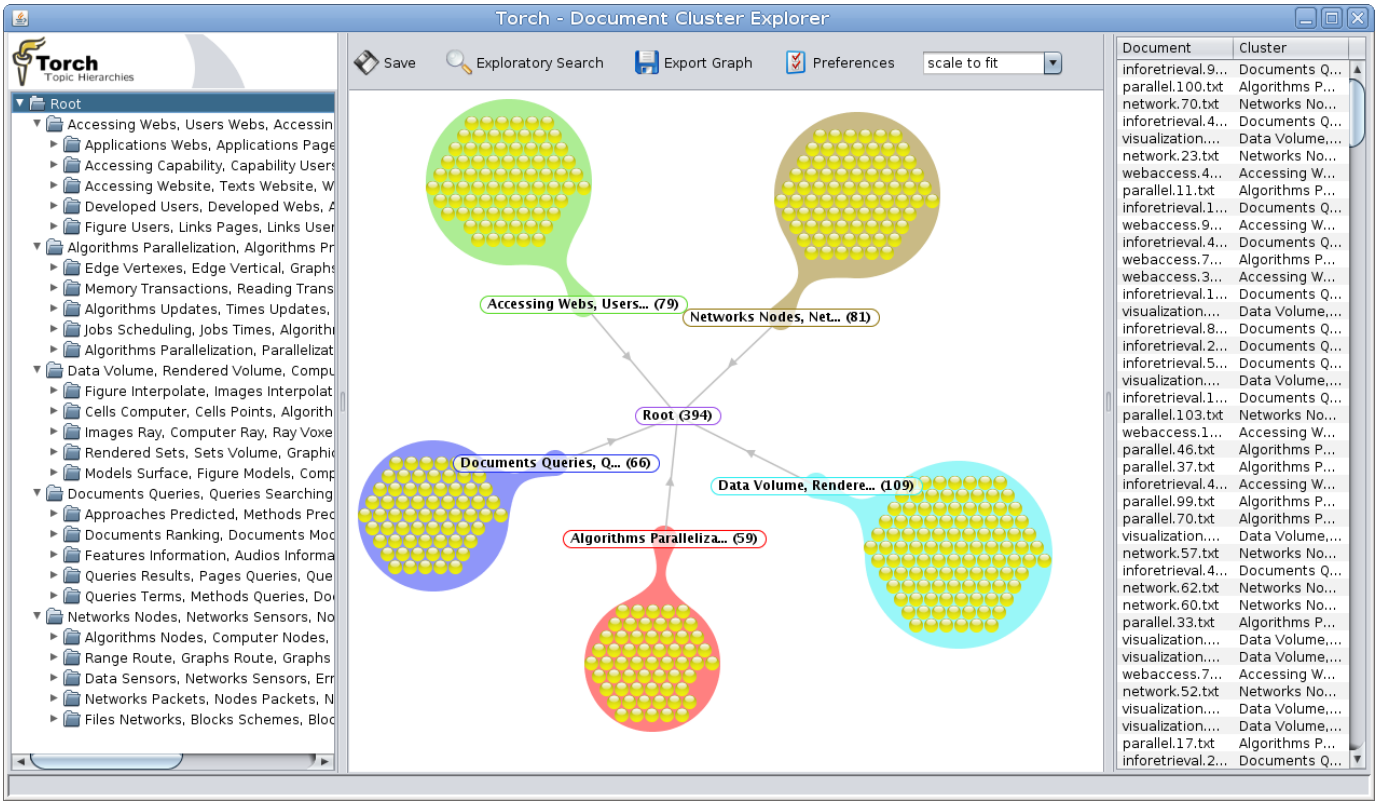

Figura 5. Tela da ferramenta Torch ilustrando a análise visual da hierarquia de tópicos e respectivo agrupamento.

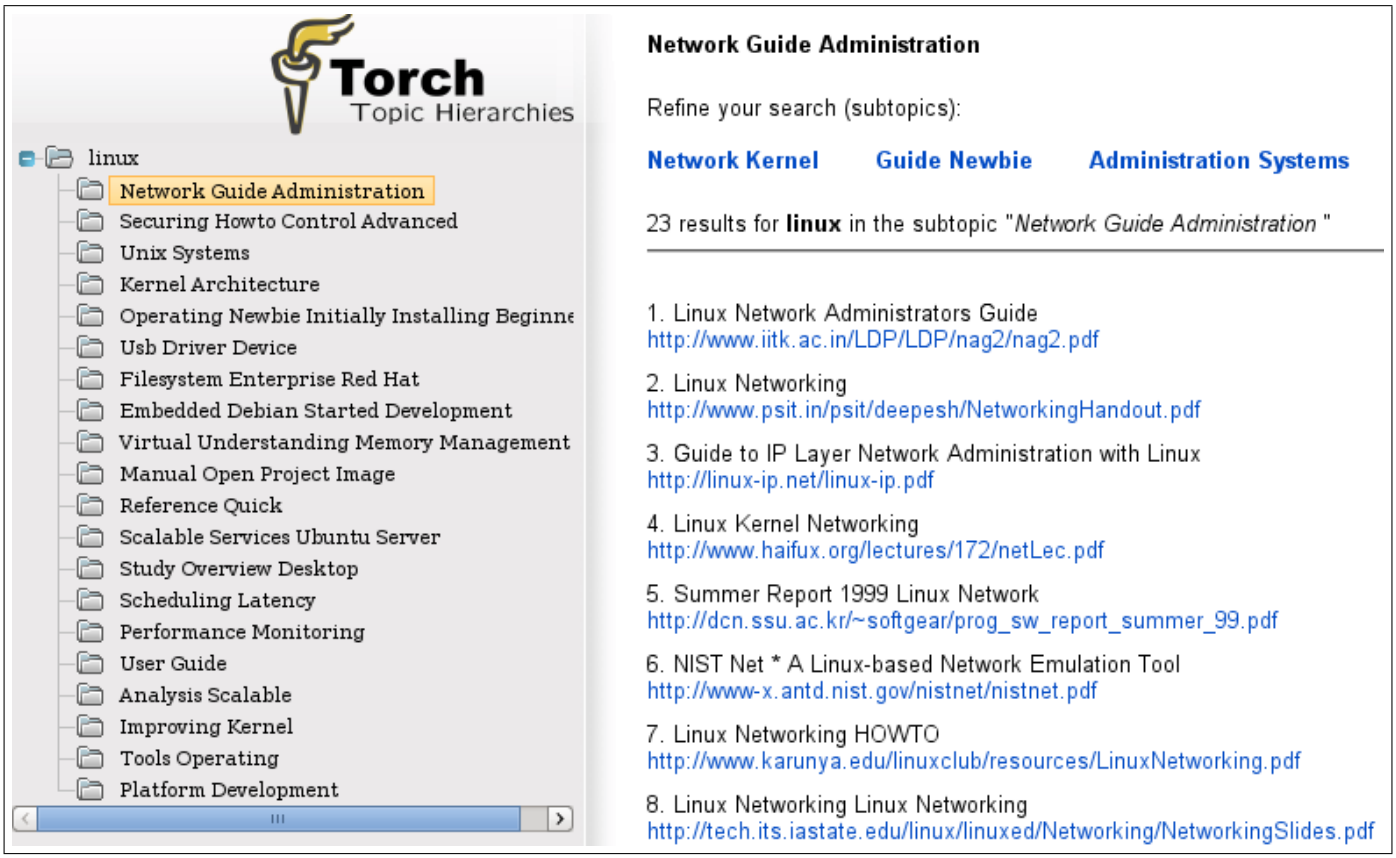

Figura 6. Exemplo de organização de resultados de busca usando uma palavra-chave para consulta na web

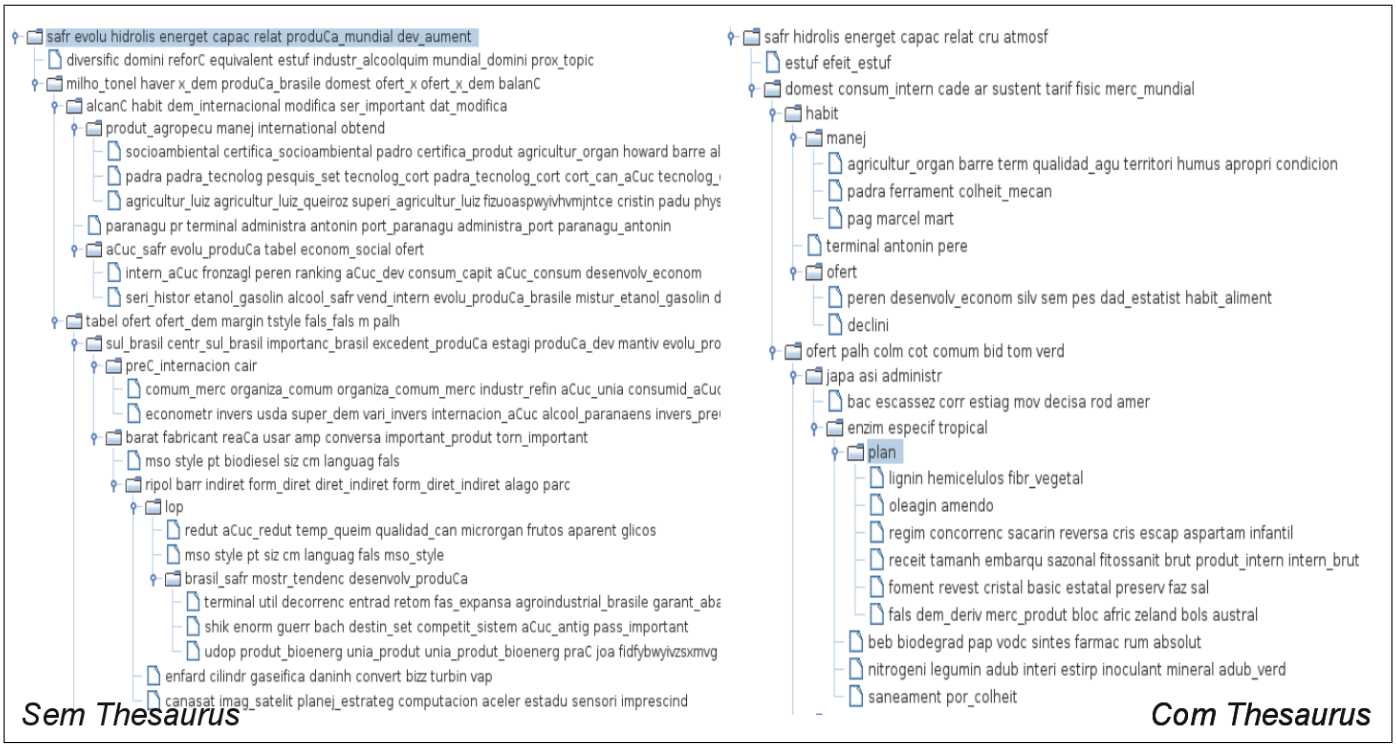

Figura 7. Exemplo de extração de Metadados pela ferramenta TaxEdit

hierarquia na figura refere-se ao assunto “efeito estufa” e o terceiro a “consumo doméstico”, claramente; ou seja, poder-se-ia atribuir essas palavras-chaves ao grupo de documentos sob esses nós. No entanto, como se tratam de notícias e não de publicações técnico-científicas, provavelmente os termos estatisticamente mais significantes indiquem “assuntos” que não se encontram no *thesaurus*, como “mistura-etanol-gasolina”, “evolução da produção brasileira”, “fabricantes”, etc. Nesse caso, o bibliotecário deverá escolher qual a melhor forma de gerar os metadados, quais seriam de maior interesse. Idealmente, no futuro, a ferramenta deve incorporar o tratamento das similaridades, a fim de melhor auxiliar a identificação desses metadados.

V. Conclusões

Em um contexto no qual grande parte das informações armazenadas pelas organizações está na forma textual, faz-se necessário o desenvolvimento de técnicas computacionais para a organização destas bases e a exploração do conhecimento nelas contido. Para tal fim, tarefas eficazes e eficientes de organização do conhecimento textual podem ser aplicadas. Dentre elas, destacam-se iniciativas para extração e organização do conhecimento de maneira não supervisionada, obtendo-se uma organização da coleção em grupos de documentos em temas e assuntos similares. Esta é a forma mais intuitiva de se estruturar o conhecimento para os usuários, uma vez que o agrupamento obtido fornece uma descrição sucinta e representativa do conhecimento implícito nos textos.

Neste artigo, foi descrita uma metodologia para orientar um processo de extração e organização de conhecimento por meio de métodos não supervisionados. Foram apresentados as principais técnicas para pré-processamento dos documentos, visando obter uma representação concisa e estruturada da coleção textual. Para a extração de padrões, discutiram-se os principais algoritmos de agrupamento de textos, bem como técnicas para selecionar descritores para os grupos formados. Alguns índices de validação de agrupamento foram apresentados, discutindo-se sua importância para a avaliação do conhecimento extraído. Por fim, foram ilustrados alguns exemplos e aplicações que se beneficiam de um processo de Mineração de Textos, com o objetivo de mostrar a utilidade prática da metodologia descrita ao longo do artigo.

A área de pesquisa envolvendo técnicas de Mineração de Textos é vasta e está em constante evolução. A riqueza dos textos e a complexidade dos problemas relacionados à linguagem e à dimensionalidade dos dados, são desafios sempre presentes quando se trata de extração de conhecimentos a partir de dados textuais. No entanto, a área tende a continuar com seu crescimento rápido devido, principalmente, à enorme quantidade de documentos publicados diariamente na web, e pela necessidade de transformar esses documentos em conhecimento útil e inovador. A proliferação das redes sociais, o aumento de repositórios públicos de notícias e artigos científicos, a convergência de armazenamento de dados na web, e o crescimento no uso de sistemas distribuídos, são exemplos de novas plataformas e desafios para a Mineração de Textos.

Referências

[1] J. F. Gantz, C. Chute, A. Manfrediz, S. Minton, D. Reinsel, W. Schlichting, and A. Toncheva, “IDC - The Diverse and Exploding Digital Universe,” *External Publication of IDC (Analyse the Future) Information and Data*, pp. 1–10, May 2008.

[2] J. F. Gantz and D. Reinsel, "As the economy contracts, the digital universe expands," *External Publication of IDC (Analyse the Future) Information and Data*, pp. 1–10, May 2009.

[3] J. F. Gantz and D. Reinsel, "The digital universe decade - are you ready?" *External Publication of IDC (Analyse the Future) Information and Data*, pp. 1–16, 2010.

[4] W. L. Kuechler, "Business applications of unstructured text," *Communications of ACM*, vol. 50, no. 10, pp. 86–93, 2007.

[5] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2nd ed. Morgan Kaufmann, 2006.

[6] D. Sullivan, *Document Warehousing and Text Mining*. John Wiley ad Sons, 2001.

[7] R. Feldman and J. Sanger, *The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge University Press, 2006.

[8] B. S. Everitt, S. Landau, and M. Leese, *Cluster Analysis*. Arnold Publishers, 2001.

[9] R. Xu and D. Wunsch, *Clustering*. Wiley-IEEE Press, IEEE Press Series on Computational Intelligence, October 2008.

[10] C. D. Manning, P. Raghavan, and H. Schütze, *An Introduction to Information Retrieval*. Cambridge University Press, 2008.

[11] B. C. M. Fung, K. Wang, and M. Ester, *The Encyclopedia of Data Warehousing and Mining*. Hershey, PA: Idea Group, August 2009, ch. Hierarchical Document Clustering, pp. 970–975.

[12] S. Chakrabarti, *Mining the web: discovering knowledge from hypertext data*. Science & Technology Books, 2002.

[13] N. F. F. Ebecken, M. C. S. Lopes, and M. C. de Aragão Costa, "Mineração de textos," in *Sistemas Inteligentes: Fundamentos e Aplicações*, 1st ed., S. O. Rezende, Ed. Manole, 2003, ch. 13, pp. 337–370.

[14] S. M. Weiss, N. Indurkhya, T. Zhang, and F. J. Damerau, *Text Mining: Predictive Methods for Analizing Unstructured Information*. Springer Science Media, 2005.

[15] D. R. Cutting, J. O. Pedersen, D. Karger, and J. W. Tukey, "Scatter/gather: A cluster-based approach to browsing large document collections," in *SIGIR'92: Proceedings of the 15th Annual International Conference on Research and Development in Information Retrieval*, 1992, pp. 318–329.

[16] M. Sanderson and B. Croft, "Deriving concept hierarchies from text," in *SIGIR '99: Proceedings of the 22nd International Conference on Research and Development in Information Retrieval*. New York, NY, USA: ACM, 1999, pp. 206–213. [Online]. Available: http://doi.acm.org/10.1145/312624.312679

[17] M. F. Moura and S. O. Rezende, "A simple method for labeling hierarchical document clusters," in *IAI'10: Proceedings of the 10th IASTED International Conference on Artificial Intelligence and Applications*, Anaheim, Calgary, Zurich : Acta Press, 2010, 2010, pp. 363–371.

[18] H.-J. Zeng, Q.-C. He, Z. Chen, W.-Y. Ma, and J. Ma, "Learning to cluster web search results," in *SIGIR'04: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval*. New York, NY, USA: ACM, 2004, pp. 210–217.

[19] C. Carpineto, S. Osiński, G. Romano, and D. Weiss, "A survey of web clustering engines," *ACM Computing Surveys*, vol. 41, pp. 1–17, 2009.

[20] M. S. Conrado, R. M. Marcacini, M. F. Moura, and S. O. Rezende, "O efeito do uso de diferentes formas de geração de termos na compreensibilidade e representatividade dos termos em coleções textuais na língua portuguesa," in *WTI'09: II International Workshop on Web and Text Intelligence*, 2009, pp. 1–10.

[21] H. P. Luhn, "The automatic creation of literature abstracts," *IBM Journal os Research and Development*, vol. 2, no. 2, pp. 159–165, 1958.

[22] G. K. Zipf, *Selective Studies and the Principle of Relative Frequency in Language*. Harvard University Press, 1932.

[23] M. V. B. Soares, R. C. Prati, and M. C. Monard, "Pretext ii: Descrição da reestruturação da ferramenta de pré-processamento de textos," Instituto de Ciências Matemáticas e de Computação, USP, São Carlos, Tech. Rep. 333, 2008.

[24] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *An International Journal of Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.

[25] L. Liu, J. Kang, J. Yu, and Z. Wang, "A comparative study on unsupervised feature selection methods for text clustering," in *NLP-KE '05. Proceedings of 2005 International Conference on Natural Language Processing and Knowledge Engineering*, 2005, pp. 597–601.

[26] G. Salton, J. Allan, and A. Singhal, "Automatic text decomposition and structuring," *Information Processing & Management*, vol. 32, no. 2, pp. 127–138, 1996.

[27] S. O. Rezende, J. B. Pugliesi, E. A. Melanda, and M. F. Paula, "Mineração de dados," in *Sistemas Inteligentes: Fundamentos e Aplicações*, 1st ed., S. O. Rezende, Ed. Manole, 2003, ch. 12, pp. 307–335.

[28] B. M. Nogueira, M. F. Moura, M. S. Conrado, R. G. Rossi, R. M. Marcacini, and S. O. Rezende, "Winning some of the document preprocessing challenges in a text mining process," in *Anais do IV Workshop em Algoritmos e Aplicações de Mineração de Dados - WAAMD, XXIII Simpósio Brasileiro de Banco de Dados-SBBD*. Porto Alegre : SBC, 2008, pp. 10–18. [Online]. Available: http://www.lbd.dcc.ufmg.br:8080/colecoes/waamd/2008/002.pdf

[29] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: a review," *ACM Computing Surveys*, vol. 31, no. 3, pp. 264–323, September 1999.

[30] P.-N. Tan, M. Steinbach, and V. Kumar, *Introduction to Data Mining*. Boston, MA, USA: Addison-Wesley Longman Publishing, 2005.

[31] P. Berkhin, "A survey of clustering data mining techniques," in *Grouping Multidimensional Data*, J. Kogan, C. Nicholas, and M. Teboulle, Eds. Berlin, Heidelberg: Springer-Verlag, 2006, ch. 2, pp. 25–71.

[32] C. L. Liu, *Introduction to combinatorial mathematics*. New York, USA: McGraw-Hill, 1968.

[33] J. B. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*, L. M. L. Cam and J. Neyman, Eds., vol. 1. University of California Press, 1967, pp. 281–297.

[34] M. Steinbach, G. Karypis, and V. Kumar, "A comparison of document clustering techniques," in *KDD'2000: Workshop on Text Mining*, 2000, pp. 1–20.

[35] Y. Zhao, G. Karypis, and U. Fayyad, "Hierarchical clustering algorithms for document datasets," *Data Mining and Knowledge Discovery*, vol. 10, no. 2, pp. 141–168, 2005.

[36] J. Metz, "Interpretação de clusters gerados por algoritmos de clustering hierárquico," Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação - ICMC - Universidade de São Paulo - USP, 2006.

[37] P. Sneath, "The application of computers to taxonomy," *Journal of General Microbiology*, vol. 17, no. 1, pp. 201–226, 1957.

[38] D. T. Larose, *Discovering Knowledge in Data: An Introduction to Data Mining*, 1st ed. Wiley-Interscience, 2004.

[39] T. Sorensen, "A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on Danish commons." *Biologiske Skrifter*, vol. 5, no. 5, pp. 35–43, 1948.

[40] R. R. Sokal and C. D. Michener, "A statistical method for evaluating systematic relationships," *University of Kansas Scientific Bulletin*, vol. 28, pp. 1409–1438, 1958.

[41] Y. Zhao and G. Karypis, "Evaluation of hierarchical clustering algorithms for document datasets," in *CIKM '02: Proceedings of the 11th international conference on Information and Knowledge Management*. New York, NY, USA: ACM, 2002, pp. 515–524.

[42] L. Kaufman and P. Rousseeuw, *Finding Groups in Data: An Introduction to Cluster Analysis*. New York: Wiley Interscience, 1990.

[43] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *KDD'96: Proceedings of 2nd International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 226 – 231.

[44] R. Agrawal, J. Gehrke, D. Gunopulos, and P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining applications," *ACM SIGMOD Record*, vol. 27, no. 2, pp. 94–105, 1998.

[45] I. H. Witten and E. Frank, *Data Mining: Practical Machine Learning Tools and Techniques*, 2nd ed. San Francisco, CA: Morgan Kaufmann, 2005.

[46] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.

[47] T. Kohonen, J. Hynninen, J. Kangas, and J. Laaksonen, "SOM PAK: The self-organizing map program package," *Relatório Técnico A31, Helsinki University of Technology, Laboratory of Computer and Information Science*, 1996.

[48] S.-L. Chuang and L.-F. Chien, "A practical web-based approach to generating topic hierarchy for text segments," in *CIKM '04: Proceedings of the thirteenth ACM International Conference on Information and Knowledge Management*. New York, NY, USA: ACM, 2004, pp. 127–136.

[49] H. Chu, *Information Representation and Retrieval in the Digital Age*. Information Today, 2003.

[50] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *Journal of Machine Learning Research*, vol. 3, pp. 1289–1305, March 2003.

[51] M. F. Moura, R. M. Marcacini, and S. O. Rezende, "Easily labelling hierarchical document clusters," in *WAAMD'08: IV Workshop em Algoritmos e Aplicações de Mineração de Dados, XXIII Simpósio Brasileiro de Banco de Dados - SBBD*. Porto Alegre : SBC, 2008, pp. 37–45.

[52] K. Faceli, A. C. P. L. F. Carvalho, and M. C. P. Souto, "Validação de algoritmos de agrupamento," Instituto de Ciências Matemáticas e de Computação - ICMC - USP, Tech. Rep. 254, 2005.

[53] M. Halkidi, Y. Batistakis, and M. Vazirgiannis, "On clustering validation techniques," *Journal of Intelligent Information Systems*, vol. 17, no. 2-3, pp. 107–145, 2001.

[54] G. Milligan and M. Cooper, "An examination of procedures for determining the number of clusters in a data set," *Psychometrika*, vol. 50, no. 2, pp. 159–179, June 1985. [Online]. Available: http://dx.doi.org/10.1007/BF02294245

[55] L. Vendramin, R. J. G. B. Campello, and E. R. Hruschka, "Relative clustering validity criteria: A comparative overview," *Statistical Analysis and Data Mining*, vol. 3, no. 4, pp. 209–235, 2010.

[56] A. K. Jain and R. C. Dubes, *Algorithms for clustering data*. Upper Saddle River, NJ, USA: Prentice-Hall, 1988.

[57] B. Larsen and C. Aone, "Fast and effective text mining using linear-time document clustering," in *SIGKDD'99: Proceedings of the 5th ACM International Conference on Knowledge Discovery and Data Mining*, 1999, pp. 16–22.

[58] G. Marchionini, "Exploratory search: from finding to understanding," *Communications of ACM*, vol. 49, no. 4, pp. 41–46, 2006.

[59] R. M. Marcacini and S. O. Rezende, "Torch: a tool for building topic hierarchies from growing text collection," in *WFA'2010: IX Workshop de Ferramentas e Aplicações. Em conjunto com o XVI Simpósio Brasileiro de Sistemas Multimídia e Web (Webmedia)*, 2010, pp. 1–3.

[60] R. M. Marcacini and S. O. Rezende, "Incremental Construction of Topic Hierarchies using Hierarchical Term Clustering," in *SEKE'2010: Proceedings of the 22nd International Conference on Software Engineering and Knowledge Engineering*. Redwood City, San Francisco, USA: KSI - Knowledge Systems Institute, 2010, pp. 553–558.

[61] M. I. F. Souza and M. D. R. Alves, "Representação descritiva e temática de recursos de informação no sistema agência embrapa: uso do padrão dublin core," *Revista Digital de Biblioteconomia e Ciência da Informação, Campinas*, vol. 7, no. 1, pp. 208–223, 2009.

[62] M. F. Moura, E. Mercanti, B. M. Peixoto, and R. M. Marcacini, "Taxedit - taxonomy editor. versão 1.0," Campinas: Embrapa Informática Agropecuária. CD-ROM., 2010.