



UNIVERSIDADE FEDERAL DA BAHIA

TRABALHO DE GRADUAÇÃO

Uma abordagem híbrida para organização flexível de documentos

Nilton Vasques Carvalho Junior

Programa de Graduação em Ciência da Computação

Salvador

2 de junho de 2016

NILTON VASQUES CARVALHO JUNIOR

**UMA ABORDAGEM HÍBRIDA PARA ORGANIZAÇÃO FLEXÍVEL
DE DOCUMENTOS**

Este Trabalho de Graduação foi apresentado ao Programa de Graduação em Ciência da Computação da Universidade Federal da Bahia, como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação.

Orientadora: Profa. Dra. Tatiane Nogueira Rios

Salvador
2 de junho de 2016

Ficha catalográfica.

Carvalho, Nilton Vasques Jr.

Uma abordagem híbrida para organização flexível de documentos/ Nilton Vasques Carvalho Junior– Salvador, 2 de junho de 2016.

56p.: il.

Orientadora: Profa. Dra. Tatiane Nogueira Rios.
Monografia (Graduação)– UNIVERSIDADE FEDERAL DA BAHIA, INSTITUTO DE MATEMÁTICA, 2 de junho de 2016.

“1. Fuzzy C Means. 2. Organização flexível de documents. 3. Lógica Fuzzy. 4. Mineração de dados.”.

I. Rios, Tatiane Nogueira. II. UNIVERSIDADE FEDERAL DA BAHIA. INSTITUTO DE MATEMÁTICA. III Título.

NUMERO CDD

TERMO DE APROVAÇÃO

NILTON VASQUES CARVALHO JUNIOR

UMA ABORDAGEM HÍBRIDA PARA ORGANIZAÇÃO FLEXÍVEL DE DOCUMENTOS

Este Trabalho de Graduação foi julgado adequado à obtenção do título de Bacharel em Ciência da Computação e aprovado em sua forma final pelo Programa de Graduação em Ciência da Computação da Universidade Federal da Bahia.

Salvador, 02 de junho de 2016

Profa. Dra. Tatiane Nogueira Rios
Universidade Federal da Bahia

Prof. Dr. Maurício Pamplona Segundo
Universidade Federal da Bahia

Prof. Dr. Ricardo Rios
Universidade Federal da Bahia

Coloque sua DEDICATÓRIA AQUI.

AGRADECIMENTOS

Coloque seus AGRADECIMENTOS AQUI.

O que sabemos é uma gota, o que ignoramos é um oceano.

—ISAAC NEWTON (1687)

RESUMO

Diante da grande quantidade de informações geradas e armazenadas pela humanidade na atualidade, vários métodos foram propostos visando processar esses dados. Dentre esses dados, temos uma imensa quantidade de dados textuais, que por sua vez são não estruturados. Com isso é notória a importância, de organizar de maneira automatizada, esses documentos pelos assuntos ao qual se tratam. Em particular temos um conjunto de técnicas pertencentes ao campo de estudo da mineração de textos, que visam realizar a tarefa de extrair informações relevantes de documentos textuais. Esta tarefa de análise e extração de informações é comumente segmentada nas tarefas de coleta, pré-processamento dos documentos, agrupamento dos dados e por fim a extração de descritores dos grupos obtidos na etapa de agrupamento. Os métodos de agrupamento podem ser separados então pela lógica matemática utilizada, que pode ser a lógica clássica ou a lógica fuzzy. Na lógica clássica, após o agrupamento, cada documento só poderá pertencer a um grupo, enquanto na lógica fuzzy, a pertinência do documento será distribuída entre os grupos. Se analisarmos a diversidade de conteúdo em documentos textuais, é trivial notar que frequentemente um texto aborda um ou mais temas. Com isso é evidente a necessidade de desenvolver-se técnicas para organizar de maneira flexível os documentos. Percebe-se então, que os métodos de agrupamento fuzzy, se mostram coerentes com a realidade multi temática dos documentos textuais. Por sua vez, o método FCM(fuzzy c means), que é uma adaptação do clássico k means, se propõe a identificar e separar uma coleção de documentos em grupos, respeitando a lógica multi valorada, permitindo assim que um documento pertença a um ou mais grupos. No entanto o FCM possui algumas falhas conhecidas, o que motivou a pesquisa e desenvolvimento de métodos alternativos e baseados no FCM, com o propósito de sanar estes problemas. Este é o caso dos métodos PCM(Possibilístico C Means) e PFCM(Possibilístico C Means). Para então avaliarmos corretamente o resultado do agrupamento e a qualidade da organização flexível de documentos, é preciso extrair corretamente os descritores dos grupos obtidos, levando em consideração a relevância de determinado termo para cada grupo. Com isso temos um cenário no qual é preciso combinar métodos de agrupamento fuzzy com métodos de extração de descritores, para obtermos um bom resultado no processo de organização dos documentos. A investigação e refinamento dessa combinação de métodos, foi a motivação do presente trabalho. Como resultado desse trabalho foi, proposto estender os experimentos referentes a organização flexível de documentos, utilizando novos métodos de agrupamento fuzzy existentes na literatura, como o PCM e o PFCM. Assim como também foi proposto os métodos de extração de descritores: i) Mixed-PFDCL (*Mixed - Possibilistic Fuzzy Descriptor Comes Last*), que se utiliza da abordagem híbrida do algoritmo PFCM, misturando assim descritores fuzzy e possibilísticos. ii) MixedW-PFDCL (*Mixed Weighted - Possibilistic Fuzzy Descriptor Comes Last*), onde além de misturar

descritores fuzzy e possibilístico, leva em consideração os parâmetros de ponderação do método PFCM. Além dos métodos de extração de descritores, foi conduzido um estudo dos impactos de se utilizar o algoritmo PCM, no método de agrupamento hierárquico HFCM, o que resultou no método de agrupamento hierárquico HPCM (*Hierarchical Possibilistic C Means*).

Palavras-chave: agrupamento fuzzy, agrupamento possibilístico, organização flexível de documentos, mineração de textos

ABSTRACT

A new powerful and flexible organization of documents can be obtained by mixing fuzzy and possibilistic clustering, in which documents can belong to more than one cluster simultaneously with different compatibility degrees with a particular topic. The topics are represented by clusters and the clusters are identified by one or more descriptors extracted by a proposed method. We aim to investigate whether the descriptors extracted after fuzzy and possibilistic clustering improves the flexible organization of documents. Experiments were carried using a collection of documents and we evaluated the descriptors ability to capture the essential information of the used collection. The results prove that the fuzzy possibilistic clusters descriptors extraction is effective and can improve the flexible organization of documents.

Keywords: fuzzy clustering, possibilistic clustering, flexible organization, documents, text mining

SUMÁRIO

Capítulo 1—Introdução	1
Capítulo 2—Fundamentação Teórica	3
2.1 Considerações iniciais	3
2.2 Conjuntos e Lógica Fuzzy	4
2.2.1 Definição de conjuntos fuzzy	4
2.2.2 Lógica fuzzy	4
2.3 Pré-Processamento	5
2.4 Agrupamento Fuzzy	6
2.4.1 Algoritmo Fuzzy C Means (FCM)	9
2.4.2 Algoritmo Possibilistic C Means (PCM)	11
2.4.3 Algoritmo Possibilistic Fuzzy C Means (PFCM)	14
2.4.4 Algoritmo Hierarchic Fuzzy C Means (HFCM)	15
2.5 Extração de descritores	18
2.6 Considerações finais	19
Capítulo 3—Trabalhos Relacionados	20
3.1 Considerações Iniciais	20
3.2 Organização Flexível de Documentos	21
3.3 Considerações finais	26
Capítulo 4—Abordagem proposta	28
4.1 Considerações iniciais	28
4.2 Informações das bases de dados	29
4.3 Refinamento com o algoritmo PFCM	31
4.4 Uma abordagem híbrida para extração de descritores	37
4.4.1 Investigações na extração de descritores em partições possibilísticas	37
4.4.2 Interpretando os graus de compatibilidade das partições possibilísticas	40
4.4.3 O método PDCL	42
4.4.4 O método Mixed-PFDCL	43
4.4.5 Resultados	44
4.5 Considerações finais	50
Capítulo 5—Conclusão	51

Capítulo 6—Trabalhos Futuros

LISTA DE FIGURAS

2.1	Ilustração denotando os grupos g_1, g_2, g_3 organizados sem sobreposição, para $m = 1$	10
2.2	Ilustração denotando os grupos g_1, g_2, g_3 organizados de maneira fuzzy, com sobreposição, quando $m \rightarrow \infty$	10
2.3	Problema dos elementos equidistantes do algoritmo FCM. Na imagem g_1 e g_2 são grupos, com os seus respectivos protótipos v_1 e v_2 . Enquanto d_1 e d_2 são documentos equidistantes aos protótipos v_1 e v_2 . Portanto $\mu(d_1, g_1) = \mu(d_1, g_2) = \mu(d_2, g_1) = \mu(d_2, g_2) = 0.5$	10
2.4	Resultado do agrupamento de dois conjuntos de coordenadas no R^2 usando o algoritmo FCM ¹	12
2.5	Demonstração de agrupamentos obtidos com os algoritmos FCM ² (a) e PCM ² (b)	14
2.6	Demonstração do agrupamento obtido com os algoritmo PFCM ³ em um conjunto de coordenadas de pontos no R^2	16
2.7	Exemplo de hierarquia de tópicos presentes em uma coleção de textos. . .	16
4.1	Estratégia de organização flexível de documentos adotada ao se misturar abordagens fuzzy e possibilísticas no agrupamento	32
4.2	Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção Opinions	34
4.3	Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção 20Newsgroup	34
4.4	Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção Hitech	35
4.5	Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção NSF	35
4.6	Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção WAP	35
4.7	Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção Reuters-21578	35

4.8	Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção Opinions	46
4.9	Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção 20Newsgroup	46
4.10	Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção Hitech	46
4.11	Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção NSF	46
4.12	Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção WAP	47
4.13	Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção Reuters-21578	47
4.14	Ilustração denotando os grupos g_1, g_2, g_3 organizados sem sobreposição, para $m = 1$	48
4.15	Ilustração denotando os grupos g_1, g_2, g_3 organizados de maneira fuzzy, com sobreposição, quando $m \rightarrow \infty$	49

LISTA DE TABELAS

3.1	Classificação das bases de dados de acordo com o seu tamanho(HAVENS et al., 2012)	22
4.1	Descrição das características objetivas presentes em coleções textuais elencadas para este trabalho	31
4.2	Características das bases de dados utilizadas nesta pesquisa	31
4.3	Quantidade ótima de grupos determinada através do método da silhueta fuzzy para cada algoritmo de agrupamento	33
4.4	Sumário dos resultados da classificação dos descritores	35
4.5	Descritores extraídos com os métodos de agrupamento FCM, PCM e PFCM da coleção Opínosis, onde μ e λ se referem as partições fuzzy e possibilística respectivamente, da qual os descritores foram extraídos.	36
4.6	Matriz de contingência do termo t_k para o grupo g_j para as medidas de recuperação de informação	38
4.7	Exemplo de matriz documentos x termos	39
4.8	Exemplo de matriz documents x grupos com graus de pertinência	39
4.9	Exemplo de matriz documents x grupos com graus de possibilidade	39
4.10	Matriz de contingência do termo t_k para o grupo g_j adaptada para a partição possibilística	42
4.11	Quantidade ótima de grupos determinada através do método da silhueta fuzzy para cada algoritmo de agrupamento no segundo experimento conduzido com os métodos PCM e PFCM	45
4.12	Sumário dos resultados da classificação dos descritores	47

Capítulo

1

Uma breve introdução sobre do que se trata esta monografia e a maneira como o texto está organizado.

INTRODUÇÃO

Diante da grande quantidade de informações geradas e armazenadas pela humanidade na atualidade, vários métodos foram propostos visando processar esses dados. Dentre esses dados, temos uma imensa quantidade de dados textuais, que por sua vez são não estruturados. Com isso é notória a importância, de organizar de maneira automatizada, esses documentos pelos assuntos ao qual se tratam. Em particular temos um conjunto de técnicas pertencentes ao campo de estudo da mineração de textos, que visam realizar a tarefa de extrair informações relevantes de documentos textuais. Esta tarefa de análise e extração de informações é comumente segmentada nas tarefas de coleta, pré-processamento dos documentos, agrupamento dos dados e por fim a extração de descritores dos grupos obtidos na etapa de agrupamento. Os métodos de agrupamento podem ser separados então pela lógica matemática utilizada, que pode ser a lógica clássica ou a lógica fuzzy. Na lógica clássica, após o agrupamento, cada documento só poderá pertencer a um grupo, enquanto na lógica fuzzy, a pertinência do documento será distribuída entre os grupos. Se analisarmos a diversidade de conteúdo em documentos textuais, é trivial notar que frequentemente um texto aborda um ou mais temas. Com isso é evidente a necessidade de desenvolver-se técnicas para organizar de maneira flexível os documentos. Percebe-se então, que os métodos de agrupamento fuzzy, se mostram coerentes com a realidade multi temática dos documentos textuais. Por sua vez, o método FCM(fuzzy c means), que é uma adaptação do clássico k means, se propõe a identificar e separar uma coleção de documentos em grupos, respeitando a lógica multi valorada, permitindo assim que um documento pertença a um ou mais grupos. No entanto o FCM possui algumas falhas conhecidas, o que motivou a pesquisa e desenvolvimento de métodos alternativos e baseados no FCM, com o propósito de sanar estes problemas. Este é o caso dos métodos PCM(Possibilístico C Means) e PFCM(Possibilístico C Means). Para então avaliarmos corretamente o resultado do agrupamento e a qualidade da organização flexível de documentos, é preciso extrair corretamente os descritores dos grupos obtidos, levando em consideração a relevância de determinado termo para cada grupo. Com isso temos um cenário no qual é preciso combinar métodos de agrupamento fuzzy com métodos de

extração de descritores, para obtermos uma bom resultado no processo de organização dos documentos. A investigação e refinamento dessa combinação de métodos, foi a motivação do presente trabalho. Como resultado desse trabalho foi, proposto extender os experimentos referentes a organização flexível de documentos, utilizando novos métodos de agrupamento fuzzy existentes na literatura, como o PCM e o PFCM. Assim como também foi proposto os métodos de extração de descritores: i) Mixed-PFDCL (*Mixed - Possibilistic Fuzzy Descriptor Comes Last*), que se utiliza da abordagem híbrida do algoritmo PFCM, misturando assim descritores fuzzy e possibilísticos. ii) MixedW-PFDCL (*Mixed Weight - Possibilistic Fuzzy Descriptor Comes Last*), onde além de misturar descritores fuzzy e possibilístico, leva em consideração os parâmetros de ponderação do método PFCM. Além dos métodos de extração de descritores, foi conduzido um estudo dos impactos de se utilizar o algoritmo PCM, no método de agrupamento hierárquico HFCM, o que resultou no método de agrupamento hierárquico HPCM (*Hierarchical Possibilistic C Means*).

Este capítulo tem como objetivo fundamentar as bases necessárias dos campos de estudos utilizados nesta monografia.

FUNDAMENTAÇÃO TEÓRICA

2.1 CONSIDERAÇÕES INICIAIS

A lógica fuzzy foi introduzida por Lofti Zadeh em (ZADEH, 1965), onde o autor inicia a discussão definindo os conjuntos fuzzy, as quais constituem classes de objetos com valores contínuos de pertinência. Cada conjunto é caracterizado por uma função de pertinência, a qual atribui a cada objeto do conjunto um grau de pertinência que varia entre zero e um. As operações matemáticas da teoria dos conjuntos, como inclusão, união, intersecção, complemento, relação, etc., também são estendidas aos conjuntos fuzzy, assim como várias propriedades dessas notações são definidas.

Uma das motivações para o uso da lógica fuzzy, vem da maneira como nosso cérebro classifica e rotula o mundo real. Por exemplo, ao rotularmos uma pessoa como alta, estamos atribuindo ela ao grupo de pessoas altas. Assim como quando nos expressamos sobre o quanto um determinado dia está fazendo calor ou frio. O conjunto de pessoas altas ou dias frios, não se enquadra na sua totalidade na lógica clássica, pois essa forma imprecisa de descrever o mundo a nossa volta desempenha um papel fundamental na forma de pensar humana, assim como também nas áreas de reconhecimento de padrões, comunicação e abstração(ZADEH, 1965). Neste sentido, esta seção tem como propósito contextualizar os principais aspectos da lógica fuzzy que a torna tão importante no contexto da organização flexível de documentos. Ressalta-se, portanto, que definições mais aprofundadas sobre a lógica fuzzy fogem do escopo desse texto. Adicionalmente, além dos principais fundamentos necessários para compreensão da abordagem apresentada neste documento, estão a atividade de pré-processamento dos dados, cuja finalidade é filtrar e estruturar os dados para serem processados nas etapas seguintes; os principais algoritmos de agrupamento fuzzy, presentes na literatura, com as suas definições matemáticas e pseudo códigos; e, por fim, a tarefa de rotular os grupos encontrados na etapa de agrupamento com os termos que melhor os representem, permitindo assim a realização de consultas em Sistemas de Recuperação da Informação (SRI).

2.2 CONJUNTOS E LÓGICA FUZZY

2.2.1 Definição de conjuntos fuzzy

Seja X um espaço de objetos, com um elemento genérico x . Sendo $X = \{x\}$. Um conjunto fuzzy A em X é caracterizado por uma função de pertinência $f_A(x)$, a qual associa a cada elemento de X um número real presente no intervalo de $[0, 1]$, sendo o valor de $f_A(x)$ a representação do grau de pertinência de x em A .

2.2.2 Lógica fuzzy

A lógica fuzzy é uma lógica multi valorada, onde os valores das variáveis pertencem ao intervalo de $[0, 1]$, enquanto na lógica clássica os valores verdadeiros só possuem os estados 0 ou 1 (também conhecido como valores *crisp*). Uma das mais importantes aplicações está no tratamento de precisão e incerteza. O que nos permite modelar soluções mais adequadas para ambientes imprecisos e incertos. Antes da lógica fuzzy ser introduzida em (ZADEH, 1965), em 1930 Lukasiewics (CHEN, 2000) desenvolveu a lógica n -valorada para $3 < n < \infty$, utilizando apenas os operadores lógicos de negação – e implicação \Rightarrow . Dado então um inteiro positivo, $n > 3$, a lógica n -valorada assume valores verdade pertencente ao intervalo $[0, 1]$, definidos pela seguinte partição igualmente espaçada:

$$0 = \frac{0}{n-1}, \frac{1}{n-1}, \frac{2}{n-1}, \dots, \frac{n-2}{n-1}, \frac{n-1}{n-1} = 1$$

Para estender a lógica n -valorada para uma lógica com infinitos valores $2 \leq n \leq \infty$, (ZADEH, 1965) modificou a lógica de Lukasiewics definindo os seguintes operadores lógicos:

$$\bar{a} = 1 - a$$

$$a \wedge b = \min\{a, b\}$$

$$a \vee b = \max\{a, b\}$$

$$a \Rightarrow b = \min\{1, 1 + b - a\}$$

$$a \Leftrightarrow b = 1 - |a - b|$$

O objetivo da lógica fuzzy é prover mecanismos para tratar imprecisão e incerteza, se baseando na teoria de conjuntos fuzzy e usando proposições imprecisas, de modo similar a lógica clássica usando proposições precisas baseadas na teoria dos conjuntos. Para entender essa noção, observe a seguir um mesmo exemplo pela ótica do raciocínio clássico e pela ótica das ferramentas usadas para descrever imprecisão da lógica fuzzy.

- Todo texto com 100 palavras ou mais da área jurídica, tem como assunto o direito.
- O texto A com título “as manifestações de junho”, tem 100 palavras da área jurídica.
- O texto B com título “política nas universidades”, tem 99 palavras da área jurídica.
- O texto A tem como assunto o direito e o texto B não tem como assunto o direito.

Essa série de proposições ilustra o raciocínio empregado na lógica clássica e, seguindo as regras de inferência, conseguimos verificar que as sentenças estão corretas. No entanto, é fácil notar que a sentença d) não expressa muito bem o nosso entendimento sobre a temática dos textos. Seria comum alguém substituir a sentença d), por: O texto B fala um pouco sobre direito. Sendo assim é possível adicionar a imprecisão comum no mundo real às sentenças anteriores, conforme reescrito a seguir.

- a) Todo texto que tem entre 50 e 100 palavras da área jurídica fala um pouco sobre direito. Enquanto todo texto que contenha 100 ou mais palavras da área jurídica fala bastante sobre direito.
- b) O texto A com título “as manifestações de junho”, tem 100 palavras da área jurídica.
- c) O texto B com título “política nas universidades”, tem 99 palavras da área jurídica.
- d) O texto A fala bastante sobre direito, enquanto o texto B fala um pouco sobre direito.

Esse tipo de dedução comumente utilizada no nosso dia a dia, não tem como ser tratada pela lógica clássica. No entanto, podemos lidar com esse tipo de inferência imprecisa, empregando a lógica fuzzy, a qual permite o uso de alguns termos linguísticos imprecisos como:

- Predicados fuzzy: antigo, raro, caro, alto, rápido
- Quantificadores fuzzy: muito, pouco, quase, alguns
- Graus de verdade fuzzy: totalmente verdadeiro, verdadeiro, parcialmente falso, falso, definitivamente falso

2.3 PRÉ-PROCESSAMENTO

Pré-processamento dos dados é o processo de limpeza e preparação dos dados para extração de padrões. Para este trabalho, especificamente, considera-se dado como sendo um documento textual e a tarefa de extração de padrões a ser considerada é o agrupamento.

Essa etapa é importante porque algumas palavras em um documento podem causar pouco ou nenhum impacto no significado geral do documento([HADDI et al., 2013](#)). Soma-se a isso o enorme custo computacional do processo de mineração de textos, devido à grande quantidade de atributos presente em dados textuais, visto que quanto maior for a coleção de textos, maior será a quantidade de palavras distintas. Tal dimensionalidade eleva bastante o custo computacional de qualquer tarefa de extração de padrões. Por isso, vários pesquisadores propuseram métodos para tentar simplificar, sintetizar e eliminar redundâncias desnecessárias nas coleções de textos.

A fase de pré-processamento voltada para a mineração de textos, as quais visam preparar dados estruturados para as clássicas operações de mineração de dados, requer técnicas muito específicas para o preparo dos dados não estruturados ([FELDMAN; SANGER, 2007](#)).

Segundo (FELDMAN; SANGER, 2007), é possível categorizar de maneira clara as técnicas de pré-processamento de textos em duas categorias, de acordo com as tarefas realizadas pela técnica e através dos algoritmos e frameworks que a mesma utiliza. Por sua vez, as técnicas categorizadas pelas suas tarefas, geralmente visam realizar a estruturação do documento através de tarefas e sub tarefas. Como por exemplo, realizar a extração de título e sub título de documentos no formato PDF. No entanto, as demais técnicas de pré-processamento são derivadas de métodos formais, e incluem esquemas de classificação, modelos probabilísticos e sistemas baseado em regras.

O processo de pré-processamento de dados textuais, inicia com um documento parcialmente estruturado e avança incrementando a estrutura através do refinamento das características do documento e adicionando novas (FELDMAN; SANGER, 2007). No contexto da mineração de textos, as características dos documentos são as suas palavras (HADDI et al., 2013). Ao final do processo, as palavras mais relevantes são utilizadas, e as demais são descartadas.

O processo como um todo envolve várias etapas, as quais pode-se elencar a remoção de espaços, expansão de abreviações, remoção de *stopwords*, que são palavras que não possuem relevância no significado geral do texto e geralmente são compostas por proposições, pronomes, artigos, interjeições dentre outras (NOGUEIRA, 2013). Assim como também o processo de *stemming* ou lematização, onde se busca encontrar o radical da palavra, visando assim remover palavras que possuam significados similares. Ainda é possível usar as técnicas de NLP (*NaturalLanguageProcessing*) para eliminar sinônimos. Por fim, é realizada a seleção de termos mais característicos para toda a coleção (HADDI et al., 2013).

Diversos métodos foram propostos visando capturar a importância dos termos em coleções textuais. Sendo o método *Term Frequency Inverse Document Frequency* (TF-IDF) um dos mais importantes e frequentemente utilizado na literatura (HADDI et al., 2013). A definição da TF-IDF está na equação (2.1), onde n é o número de documentos da coleção, DF o total de documentos que possuem este termo e FF (*frequency feature*) a frequência do termo no documento.

$$\varphi(t, d) = FF * \log\left(\frac{N}{DF}\right) \quad (2.1)$$

Como resultado final de todo o processo de pré-processamento, obtém-se a matrix D . Onde D representa os n documentos da coleção, sendo cada documento d_i , com $1 \leq i \leq n$, uma linha da matriz D , definido como sendo $d_i = [\varphi(t_1, d_i), \varphi(t_2, d_i), \varphi(t_3, d_i), \dots, \varphi(t_k, d_i)]$, onde t_j é um termo presente na coleção, com $1 \leq j \leq k$.

2.4 AGRUPAMENTO FUZZY

O agrupamento é um processo não supervisionado cujo o objetivo é organizar os objetos similares no mesmo grupo e os objetos com grau de dissimilaridade elevado em grupos distintos (NOGUEIRA, 2013). Este processo é de grande utilidade para diversos campos de estudo da inteligência computacional, como a mineração de dados, recuperação de informação, segmentação de imagens e classificação de padrões. Neste trabalho, os objetos

a serem agrupados são os documentos textuais.

O problema de organizar os documentos de maneira a maximizar a similaridade entre os membros de um mesmo grupo, e minimizar a similaridade entre documentos de grupos distintos, é essencialmente um problema de otimização. Sendo assim, pretende-se otimizar a escolha dos grupos, entre todas as possibilidades de agrupamento, dada uma função objetivo que captura a qualidade dos grupos. Esta função é responsável por atribuir ao conjunto de possíveis grupos um número real, de maneira que quanto melhor for os grupos, maior será o seu valor (FELDMAN; SANGER, 2007).

A medida de similaridade desempenha um papel fundamental no agrupamento, uma vez que ela precisa expressar o quão distante está um elemento do outro na coleção. Assim sendo, para obtermos bons resultados durante o processo de agrupamento é de grande importância a escolha adequada da medida de similaridade, e esta escolha precisa ser feita de acordo com o tipo dos dados. Na literatura, a medida de similaridade mais popular é a distância euclidiana (Equação 2.2), que tem se mostrado bastante adequada em dados com baixa dimensionalidade.

$$D(x_i, x_j) = \sqrt{\sum_k (x_{ik} - x_{jk})^2} \quad (2.2)$$

No entanto, em coleções textuais a matriz documentos x termos é naturalmente esparsa, devido a grande variedade de termos em uma coleção, o que faz com que um determinado documento d_i não contenha diversos termos presentes em um outro documento d_j . Assim, o vetor de características de cada documento acaba sendo preenchido com vários zeros, reduzindo a eficácia da distância euclidiana (Equação 2.2) (NOGUEIRA, 2013).

A medida de similaridade mais comum para coleções textuais é o coeficiente de similaridade de cosseno (FELDMAN; SANGER, 2007). Por sua vez o coeficiente de similaridade de cosseno, desconsidera os diversos zeros presentes nos vetores de termos dos documentos, levando em conta apenas o ângulo formado entre eles (NOGUEIRA, 2013). Na equação (2.3) temos a definição do coeficiente de similaridade de cosseno, onde d_1 e d_2 , são dois documentos quaisquer da coleção de documentos, e $1 \leq t \leq k$, onde k é a quantidade total de termos da coleção, e d_{it} a frequência do termo t no documento d_i .

$$scos(d_1, d_2) = \cos\theta = \frac{d_1 \cdot d_2}{|d_1||d_2|} = \sum_{t=1}^k \varphi(d_{1t}, d_1) \cdot \varphi(d_{2t}, d_2) \in [0, 1] \quad (2.3)$$

Os grupos resultantes de um processo de agrupamento, podem possuir algumas características que estão diretamente relacionadas com o método de agrupamento empregado. Estes podem ser *hard* ou *crisp*, caso o método de agrupamento seja baseado na lógica clássica, assim como podem ser *soft*, caso o método seja baseado na lógica fuzzy. No agrupamento *hard*, cada documento d_i só poderá pertencer a um único grupo g_j (BEZ-DEK et al., 1984). Enquanto em grupos *soft*, cada documento d_i pode pertencer a um ou mais grupos g , com grau de pertinência variados. Além destes, os grupos ainda podem ser *flat* ou hierárquicos, onde no agrupamento *flat* todos os grupos estão no mesmo nível,

enquanto no modelo hierárquico os grupos podem estar dispostos em uma hierarquia, de modo que uma relação de parentesco é definida entre eles.

Para o agrupamento *hard*, considere $G = \{g_1, g_2, g_3, \dots, g_c\}$ os grupos resultantes do agrupamento, sendo c o total de grupos, e a pertinência de cada documento d_i à um grupo g_l , $1 \leq l \leq c$, pode ser representada pela função característica $\kappa(d_i, g_j) \in \{0, 1\}$. Um dos mais populares algoritmos a implementar essa abordagem é o K-Means. Em (BEZDEK et al., 1984)(NOGUEIRA, 2013)(FELDMAN; SANGER, 2007), é apontada uma falha inerente dessa abordagem, pois quando um documento só pode pertencer a um único grupo, fica evidenciado que o mesmo não compartilha nenhuma similaridade com os documentos dos demais grupos, o que não expressa a imprecisão intrínseca da sobreposição dos assuntos em documentos de texto.

Com o objetivo de tratar essa falha da abordagem *hard* e adicionar o tratamento de imprecisão e incerteza no agrupamento, (BEZDEK et al., 1984) utilizou o modelo de partições fuzzy definido em (ZADEH, 1965), para permitir pertinências parciais de um elemento a um grupo, propondo assim o algoritmo Fuzzy C Means (FCM). Sendo assim, a função característica passa a ser considerada como uma função de pertinência de um documento d_i em um grupo g_l , a qual é definida por $\mu(d_i, g_l) \in [0, 1]$, tal que

$$\sum_{l=1}^c \mu(d_i, g_l) = 1.$$

Um desafio sempre presente em métodos de agrupamento é a descoberta do número ideal de grupos em uma coleção. O método de organização flexível proposto em (NOGUEIRA, 2013) utiliza a *Fuzzy Silhouette* (FS) para realizar a validação do agrupamento fuzzy, e por conseguinte encontrar o número de grupos ideal. A função FS é uma adaptação do método de critério de largura média (*Average Silhouette Width Criterion* - ASWC), desenvolvido para o agrupamento *crisp*. A definição da silhueta fuzzy está nas equações (2.4) e (2.5), onde $\alpha(d_i, g_l)$ é a distância média entre o documento d_i e todos os documentos presentes no grupo g_l , enquanto $\beta(d_i, g_l) = \min\{\alpha(d_i, g_h) | 1 \leq h \leq c; h \neq l\}$, é a medida de dissimilaridade de d_i ao grupo vizinho mais próximo de g_l , tal que c é a quantidade de grupos.

$$S(d_i) = \frac{\beta(d_i, g_l) - \alpha(d_i, g_l)}{\max\{\alpha(d_i, g_l), \beta(d_i, g_l)\}} \quad (2.4)$$

$$FS = \frac{\sum_{i=1}^n (\mu_1(d_i) - \mu_2(d_i)) S(d_i)}{\sum_{i=1}^n (\mu_1(d_i) - \mu_2(d_i))} \quad (2.5)$$

Na equação (2.5), $\mu_1(d_i)$ é a maior pertinência do documento d_i em um grupo, enquanto $\mu_2(d_i)$ é a segunda maior. Quanto maior então for o valor da função FS, melhor será o agrupamento. Deste modo, para encontrar o número de grupos ideal, basta executar a função FS variando o número de grupos, e selecionar o agrupamento que tiver o valor máximo de FS.

Toda investigação realizada neste trabalho tomou como base os métodos de agrupamento que derivam do algoritmo FCM (BEZDEK et al., 1984), descrito a seguir, para se beneficiar da capacidade de tratar imprecisão e incerteza da lógica fuzzy, e por con-

seguinte permitir que um mesmo documento seja categorizado em mais de um t pico (grupo), refletindo a realidade dos documentos textuais.

Em todos os experimentos a medida de similaridade de cosseno (Equa  o 2.3) foi utilizada, e a quantidade de grupos ideal foi escolhida utilizando o m todo da silhueta fuzzy (Equa  o 2.5).

2.4.1 Algoritmo Fuzzy C Means (FCM)

Bezdek et al. (1984) descreve um m todo de agrupamento fuzzy que produz como sa da parti  es fuzzy e prot tipos dos grupos. Esse algoritmo desempenha um papel importante no contexto do agrupamento fuzzy, devido ao seu pioneirismo no campo de estudo, possuindo diversas extens es, sendo considerado um dos mais amplamente utilizados m todos de agrupamento fuzzy da literatura (PAL et al., 2005). A maioria dos m todos de agrupamento fuzzy s o deriva  es do FCM (KRISHNAPURAM; KELLER, 1993).

Para compreens o do algoritmo, considere $V = \{v_1, v_2, v_3, \dots, v_c\}$ os prot tipos dos grupos $G = \{g_1, g_2, g_3, \dots, g_c\}$ definidos por

$$V = \left\{ v_j | v_j = \frac{\sum_{i=1}^n [\mu(d_i, g_j)]^m d_i}{\sum_{i=1}^n [\mu(d_i, g_j)]^m}, 1 < j \leq c \right\}, \quad (2.6)$$

tal que v_j seja o prot tipo de g_j , c o n mero de grupos gerados no agrupamento e n o n mero de documentos presentes na cole  o.

Seja $U_{c \times n}$ uma parti  o fuzzy conforme a Equa  o:

$$U_{c \times n} = \{\mu(d_i, g_k) | \mu(d_i, g_k) \in [0, 1], 1 < i \leq n, 1 < k \leq c\} \quad (2.7)$$

Esta parti  o apresenta todas as pertin ncias dos documentos aos grupos, cuja fun  o de pertin ncia $\mu(d_i, g_k)$ de cada documento na cole  o,   definida por

$$\mu(d_i, g_k) = \frac{1}{\sum_{j=1}^n \left(\frac{\text{dist}(d_i, v_k)}{\text{dist}(d_i, v_j)} \right)^{\frac{1}{m-1}}} \quad (2.8)$$

e est  sujeita as restri  es definidas nas equa  es (2.9) e (2.10).

Considera-se m o fator de fuzifica  o, que regula o qu o fuzzy ser  as parti  es finais. De modo que para $m = 1$ a parti  o resultante   totalmente *crisp* (Figura 2.1) e para $m \rightarrow \infty$ a interse  o entre os grupos tende a aumentar (Figura 2.2) (PAL et al., 2005).

Segundo Bezdek et al. (1984) nenhuma teoria ou evid ncia computacional aponta para um valor  timo de m , contudo o autor aponta que a faixa de valores ideais aparenta ser $[1, 30]$. Sendo assim, se existir um conjunto de dados para teste, uma boa estrat gia para a escolha de m   a realiza  o de testes experimentais. Caso contr rio, o intervalo de $[1.5, 3.0]$ aparenta trazer bons resultados para a maior parte dos dados.

Onde usualmente no contexto do agrupamento de cole  es textuais $\text{dist}(d_i, g_k) = \text{scos}(d_i, g_k)$. Adicionalmente, o algoritmo FCM estabelece algumas restri  es: deve-se evitar que o FCM produza grupos vazios (Equa  o (2.9)) e deve-se impor que a soma das pertin ncias seja sempre igual a um (Equa  o (2.10)).

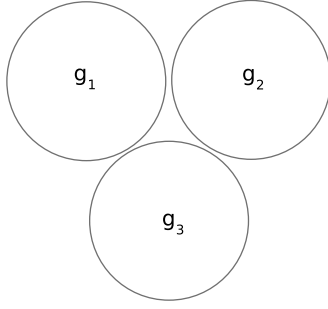


Figura 2.1: Ilustração denotando os grupos g_1, g_2, g_3 organizados sem sobreposição, para $m = 1$.

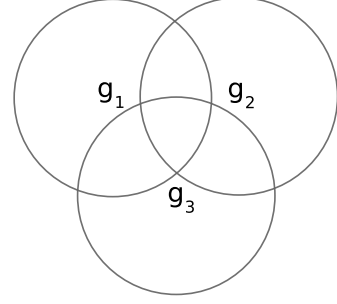


Figura 2.2: Ilustração denotando os grupos g_1, g_2, g_3 organizados de maneira fuzzy, com sobreposição, quando $m \rightarrow \infty$.

$$\sum_{k=1}^c \mu(d_i, g_k) = 1 \quad (2.9)$$

$$0 < \sum_{i=1}^n \mu(d_i, g_k) < n \quad (2.10)$$

A restrição imposta pela Equação (2.10), no entanto, produz um problema em elementos equidistantes aos grupos. Ou seja, quando temos o caso $\mu(d_i, g_1) = \mu(d_i, g_2) = \dots = \mu(d_i, g_c)$. Nessa situação, o grau de pertinência de um elemento a cada grupo será a pertinência média, ou seja $\mu(d_i, g_1) = \mu(d_i, g_2) = \dots = \mu(d_i, g_c) = \frac{1}{c}$. Supondo agora um segundo documento d_2 , que seja mais distante do que d_1 , porém assim como d_2 , também equidistante aos grupos, temos que $\mu(d_2, g_j) = \mu(d_1, g_j) = \frac{1}{c}$, para $1 < j \leq c$. Nesse contexto, a pertinência de d_2 e d_1 , não expressa a distância relativa desses documentos aos grupos. Esse problema está ilustrado na Figura (2.3).

Segundo Bezdek et al. (1984) várias funções de otimização das partições fuzzy produzidas no agrupamento foram propostas. Sendo a minimização da função objetivo $J(U_{c \times n}, G, V, D)$, que visa minimizar a distância entre os documentos e os protótipos dos grupos (NOGUEIRA, 2013), definida na equação (2.11) a mais popular.

$$\min\{J(U_{c \times n}, G, V, D) = \sum_{i=1}^n \sum_{j=1}^c [\mu(d_i, g_j)]^m \text{dist}(d_i, v_j)\} \quad (2.11)$$

O mais utilizado algoritmo para soluções aproximadas da minimização (2.11), é a iteração de Picard¹ (PAL et al., 2005) através das equações (2.6)(2.7). Sendo assim o ciclo de aproximações se dá por $V_{t-1} \Rightarrow U_t \Rightarrow V_t$, onde ao final de cada iteração é verificado se

¹Método iterativo para construção de soluções aproximadas, atribuído ao matemático francês Charles Emile Picard (1856-1941). <http://mathfaculty.fullerton.edu/mathews/n2003/PicardIterationMod.html>

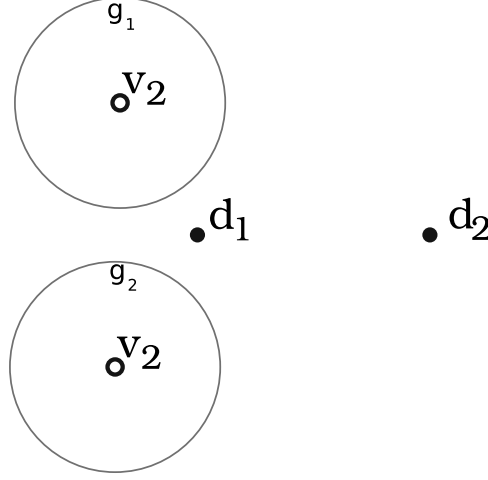


Figura 2.3: Problema dos elementos equidistantes do algoritmo FCM. Na imagem g_1 e g_2 são grupos, com os seus respectivos protótipos v_1 e v_2 . Enquanto d_1 e d_2 são documentos equidistantes aos protótipos v_1 e v_2 . Portanto $\mu(d_1, g_1) = \mu(d_1, g_2) = \mu(d_2, g_1) = \mu(d_2, g_2) = 0.5$.

$\|V_{t-1} - V_t\| < \varepsilon$. A literatura também expressa que os ciclos podem começar pela partição fuzzy, fazendo então $U_{t-1} \Rightarrow V_t \Rightarrow U_t$, e ao final do ciclo checando o erro mínimo com $\|U_{t-1} - U_t\| < \varepsilon$, sendo t o contador de iterações. Contudo existem benefícios em termos de processamento e memória ao utilizar a iteração iniciando e finalizando com V (PAL et al., 2005). (BEZDEK et al., 1984) e (PAL et al., 2005) afirmam que a convergência desse modelo iterativo ocorre em ambos os tipos de ciclo. A partição fuzzy inicial U_0 é comumente inicializada com valores aleatórios (NOGUEIRA, 2013) ou com o resultado de um agrupamento previamente executado (PAL et al., 2005) (KRISHNAPURAM; KELLER, 1993). Nas demais iterações a atualização dos protótipos é realizada a partir da equação (2.6). O pseudo código utilizando essa abordagem iterativa está listado no algoritmo (1), no qual **inicializa-particao-fuzzy**(D, G), pode ser uma das duas formas de inicialização descritas anteriormente.

Data: D, c, m, ε

Result: V, U

$G \leftarrow [g_1, g_2, \dots, g_c]$;

$U_0 \leftarrow \text{inicializa-particao-fuzzy}(D, G)$;

$t \leftarrow 0$;

faça

$V_t \leftarrow$ calcula usando (2.6);

$t \leftarrow t + 1$;

$U_t \leftarrow$ calcula usando (2.7);

enquanto $\|U_{t-1} - U_t\| > \varepsilon$;

retorne (U_t, V_t);

Algoritmo 1: Pseudo código da implementação iterativa do método FCM

Por fim está ilustrado na figura (2.4), os resultados produzido pelo algoritmo FCM,

em dois conjuntos de coordenadas no R^2 . Na figura os pontos foram pintados com a cor correspondente ao grupo em que o mesmo obteve o maior valor de pertinência.

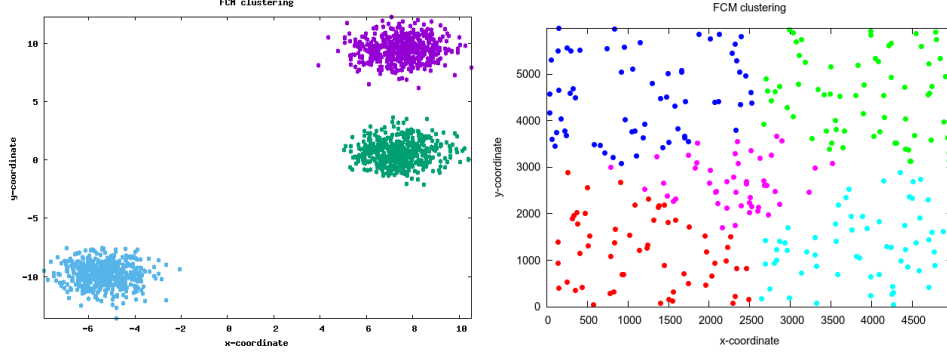


Figura 2.4: Resultado do agrupamento de dois conjuntos de coordenadas no R^2 usando o algoritmo FCM².

2.4.2 Algoritmo Possibilistic C Means (PCM)

A restrição probabilística (2.10) do FCM que obriga a soma das pertinências de um elemento ser igual a um, nem sempre resulta em pertinências que representam bem a realidade dos dados, conforme exemplificado na figura (2.3). Esse problema se agrava ainda mais, em bases com muitos dados ruidosos (*outliers*). Visando então contornar esses problemas do FCM, foi proposto em (KRISHNAPURAM; KELLER, 1993) o algoritmo *Possibilistic C Means* (PCM).

Ao contrário do FCM, o PCM não atribui pertinências dos documentos aos grupos, e sim tipicidades. Que podem ser interpretadas como graus de possibilidade de um elemento pertencer ao grupo. Como consequência a partição resultante é possibilística. Para se adequar a essa abordagem possibilística, a função objetivo do PCM deriva da equação (2.11) do FCM. Tendo também as funções de atualização de prótotipos e atribuição de pertinências modificadas.

Na teoria de conjuntos fuzzy, a pertinência de um elemento a um grupo fuzzy não depende da pertinência desse mesmo elemento em outro grupo. No entanto no modelo FCM, a restrição (2.10) torna dependente a pertinência dos elementos aos grupos. De maneira que se um elemento obtiver um grau elevado de pertinência em um dado grupo, ele não poderá ter uma pertinência também elevada em outro grupo, ou seja $\mu(d_1, g_1) = 1 - \mu(d_1, g_2)$, supondo um agrupamento com dois grupos. Portanto (KRISHNAPURAM; KELLER, 1993) relaxa a restrição (2.10), fazendo com que a pertinência dependa unicamente da distância do elemento ao grupo. Logo as restrições (2.9) e (2.10), são redefinidas nas equações (2.12)(2.13), onde $\lambda(d_i, g_j)$ representa a tipicidade do documento d_i em relação ao grupo g_j .

$$\lambda(d_i, g_j) \in [0, 1], \forall i, j \quad (2.12)$$

$$0 < \sum_{j=1}^n \lambda(d_i, g_j) \leq n, \forall j \quad (2.13)$$

Após relaxar a restrição (2.10) se mantivermos a função objetivo do FCM (equação 2.11), teríamos uma solução trivial, bastando atribuir 0 a todas as pertinências para minimizar $J(U_{c \times n}, G, V, D)$ (KRISHNAPURAM; KELLER, 1993) (NOGUEIRA, 2013). Portanto buscando evitar essa solução trivial, e manter a característica de atribuir aos elementos representativos pertinências elevadas aos grupos e penalizar os elementos não representativos, a função objetivo é reformulada como sendo

$$K_m(P_{c \times n}, G, V, D) = \sum_{j=1}^c \sum_{i=1}^n [\lambda(d_i, g_j)]^m \text{dist}(d_i, v_j) + \sum_{j=1}^c \gamma_j \sum_{i=1}^n [1 - \lambda(d_i, g_j)]^m \quad (2.14)$$

, onde $\gamma_j > 0$ são parâmetros que determinam a distância a qual o valor de pertinência (tipicidade) se torna 0.5. (KRISHNAPURAM; KELLER, 1993) explica que γ_j seja escolhido a depender da faixa de possibilidades (pertinência) desejada para um grupo. Por exemplo, γ_j pode ser igual para todos os grupos, quando se deseja que a forma dos grupos seja similar. Contudo na maioria dos casos se espera que γ_j reflita o formato e tamanho particular de cada grupo. Assim sendo, o autor indica que a definição (2.15) se mostra adequada para maior parte dos dados, onde L é usualmente 1.

$$\gamma_j = L \frac{\sum_{i=1}^n \lambda(d_i, g_j)^m \text{dist}(d_i, v_j)}{\sum_{i=1}^n \lambda(d_i, g_j)^m} \quad (2.15)$$

A partição de pertinências possibilísticas produzidas pelo PCM é então definida como sendo

$$P_{c \times n} = \{\lambda(d_i, g_k) | \lambda(d_i, g_k) \in [0, 1], 1 < i \leq n, 1 < k \leq c, \text{eqs}(2.12)(2.13)\} \quad (2.16)$$

$$\lambda(d_i, g_j) = \frac{1}{1 + \left(\frac{\text{dist}(d_i, g_j)}{\gamma_j} \right)^{\frac{1}{m-1}}} \quad (2.17)$$

, enquanto a atualização de protótipos ocorre de maneira similar a equação (2.6) do FCM, apenas alterando a pertinência $\mu(d_i, g_j)$ por $\lambda(d_i, g_j)$. A síntese do algoritmo PCM está

apresentada em forma de pseudo código no algoritmo (2).

```

pcm(D,c,m, $\varepsilon$ )
início
     $G \leftarrow [g_1, g_2, \dots, g_c]$ ;
     $P_0 \leftarrow \text{inicializa-particao-fuzzy}(D, G)$ ;
     $\gamma_j \leftarrow$  calcula utilizando (2.15);
     $t \leftarrow 0$ ;
    faça
         $V_t \leftarrow$  calcula usando (2.6);
         $t \leftarrow t + 1$ ;
         $P_t \leftarrow$  calcula usando (2.16);
    enquanto  $\| P_{t-1} - P_t \| > \varepsilon$ ;
    retorne( $P_t, V_t$ );
fim

```

Algoritmo 2: Pseudo código da implementação iterativa do método PCM

Por fim está ilustrado na figura (2.5) (a), o resultado do agrupamento obtido pelo método FCM, e em (b) o agrupamento gerado pelo algoritmo PCM em um conjunto de coordenadas no R^2 . Na figura os pontos foram pintados com a cor correspondente ao grupo em que o mesmo obteve o maior valor de pertinência. Observa-se nessa comparação simplificada que o algoritmo PCM tentou maximizar a pertinência dos pontos aos grupos maiores, ocasionando uma maior quantidade de pontos com pertinência elevada em dois grupos, ao contrário do FCM que distribuiu de maneira uniforme os pontos em 4 grupos.

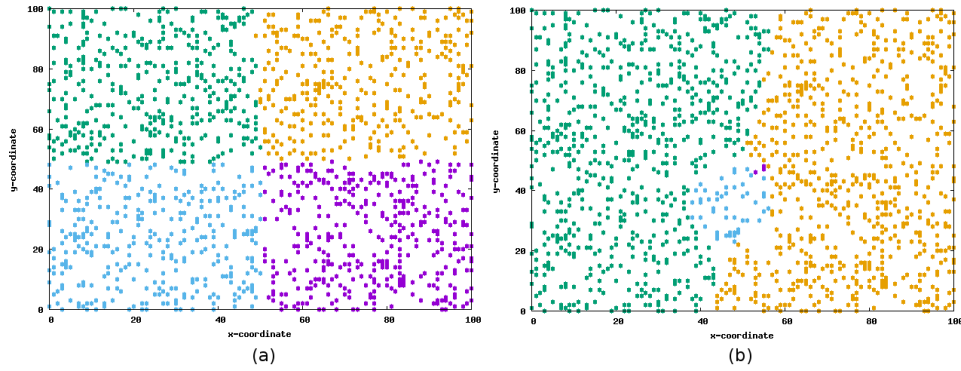


Figura 2.5: Demonstração de agrupamentos obtidos com os algoritmos FCM³(a) e PCM³(b) .

2.4.3 Algoritmo Possibilistic Fuzzy C Means (PFCM)

De acordo com (PAL et al., 2005) o algoritmo PCM pode levar os resultados do agrupamento a conter grupos coincidentes, ou seja quando o protótipo v_i está muito próximo de outro protótipo v_j . Segundo os autores, isto ocorre quando a inicialização da partição inicial não possui protótipos suficientemente separados. Esse problema não é causado por uma escolha ruim da penalidade presente na função objetivo do PCM, o que ocorre

é uma falta de restrições para evitar que isso aconteça.

(CARVALHO et al., 2016) cita que as pertinências do FCM e as tipicidades do PCM são ambas importantes para a correta interpretação das sub estruturas dos dados. Pois quando se tem dados que precisam ser rotulados de maneira *hard*, as pertinências se mostram como uma escolha adequada, de modo que é intuitivo atribuir o elemento ao grupo em que o mesmo possua a menor distância. Por outro lado, durante a atualização dos protótipos, as tipicidades desempenham um papel fundamental para aliviar os efeitos indesejados dos dados ruidosos.

Com o propósito de aproveitar então os benefícios de ambas as abordagens, (PAL et al., 2005) propôs o algoritmo PFCM, que utiliza as pertinências $\mu(d_i, g_j)$ do FCM e as tipicidades $\lambda(d_i, g_j)$ do PCM. Deixando o usuário definir a proporção de cada uma das contribuições com parâmetros que ponderam o peso de ambos. Portanto é realizado uma mistura entre as funções objetivo (2.11) e (2.14) resultando na minimização da função objetivo (2.18), que está sujeita as condições de (2.19), onde $a, b > 0$ e $m, n > 1$. Os parâmetros a, b representam a importância relativa dos valores de pertinência e tipicidades e devem ser definidos pelo usuário de acordo com o problema. Os autores sugerem que b seja maior que a , porém não muito maior, para não eliminar completamente os benefícios do FCM.

$$L_m(U_{c \times n}, P_{c \times n}, G, V, D) = \sum_{j=1}^c \sum_{i=1}^n [a\mu(d_i, g_j)^n + b\lambda(d_i, g_j)^m] \text{dist}(d_i, v_j) + \sum_{j=1}^c \gamma_j \sum_{i=1}^n [1 - \lambda(d_i, g_j)]^m \quad (2.18)$$

$$\sum_{j=1}^c \mu(d_i, g_j) = 1, \forall i, 0 < \mu(d_i, g_j), \lambda(d_i, g_j) \leq 1 \quad (2.19)$$

A mistura e as ponderações adicionados no algoritmo PFCM também são agregadas a função de atualização dos protótipos (2.20), a qual passa a se beneficiar das características de ambos os algoritmos. De maneira a reduzir os efeitos dos dados ruidosos, minimizar o problema dos protótipos coincidentes e evitar a singularidade do FCM. O pseudo código do PFCM é apresentado no Algoritmo 3, onde a função **inicializa-prototipos**(D, G) gera os protótipos iniciais da partição V_0 . Como resultado demonstrativo desse algoritmo está ilustrado na figura (2.6), onde é possível observar que os grupos produzidos são em certa perspectiva um intermédio entre o agrupamento produzido pelo FCM e PCM no mesmo conjunto de dados, apresentados na figura (2.5).

$$V = \left\{ v_j | v_j = \frac{\sum_{i=1}^n [a\mu(d_i, g_j)^n + b\lambda(d_i, g_j)^m] d_i}{\sum_{i=1}^n [a\mu(d_i, g_j)^n + b\lambda(d_i, g_j)^m]}, 1 < j \leq c \right\} \quad (2.20)$$

Data: D, c, m, ε
Result: V, P
 $G \leftarrow [g_1, g_2, \dots, g_c];$
 $V_0 \leftarrow \text{inicializa-prototipos}(D, G);$
 $\gamma_j \leftarrow \text{calcula utilizando (2.15)};$
 $t \leftarrow 0;$
faça
 $U_t \leftarrow \text{calcula com (2.7) usando } V_{t-1};$
 $P_t \leftarrow \text{calcula com (2.16) usando } V_{t-1};$
 $V_t \leftarrow \text{calcula com (2.20)};$
 $t \leftarrow t + 1;$
enquanto $\|V_{t-1} - V_t\| > \varepsilon;$
retorne $(U_t, P_t, V_t);$
Algoritmo 3: Pseudo código da implementação iterativa do método PFCM

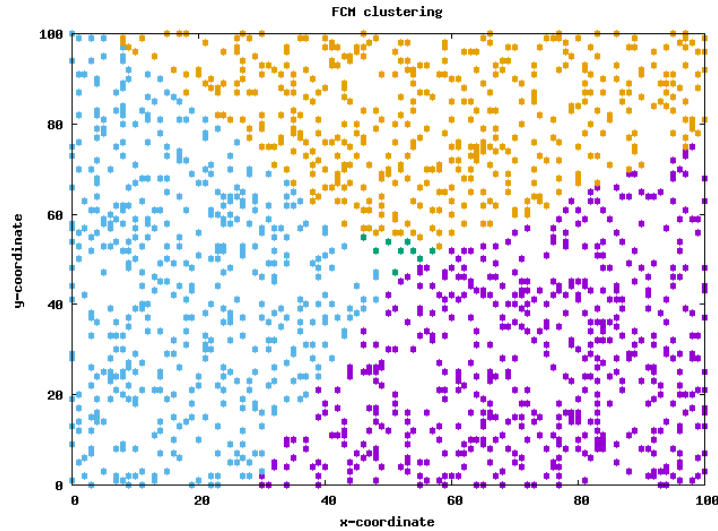


Figura 2.6: Demonstração do agrupamento obtido com os algoritmo PFCM⁴ em um conjunto de coordenadas de pontos no R^2 .

2.4.4 Algoritmo Hierarchic Fuzzy C Means (HFCM)

Documentos de texto tratam de vários temas, como política, esporte, tecnologia e etc. E os métodos de agrupamento *soft*, como FCM e PCM, quando aplicados a coleções textuais, buscam encontrar semelhanças entre os documentos e agrupar por tópicos. Porém acontece, que um tópico pode se dividir em sub temas, como por exemplo esporte, que pode se dividir em futebol, vôlei, tênis e etc. Deste modo os temas presentes em uma coleção textual podem ser organizados em uma hierarquia de tópicos conforme a figura 2.7. Portanto construir hierarquias utilizando métodos de agrupamento fuzzy é o propósito principal do algoritmo HFCM proposto em (PEDRYCZ; REFORMAT, 2006).

O HFCM consegue realizar essa tarefa, expandindo sucessivamente as folhas presentes

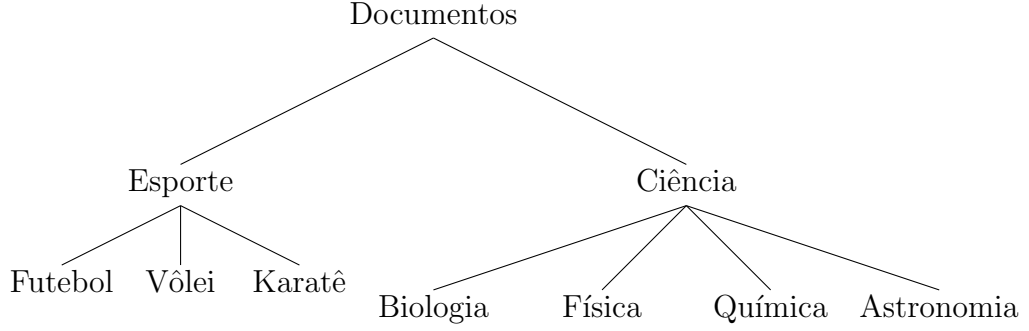


Figura 2.7: Exemplo de hierarquia de tópicos presentes em uma coleção de textos.

na hierarquia em subgrupos mais detalhados (PEDRYCZ; REFORMAT, 2006). Onde a expansão é realizada através de novos agrupamentos com o algoritmo CFCM, que é uma versão condicional do FCM. No entanto o primeiro nível da hierarquia é o resultado direto do algoritmo FCM, enquanto os demais níveis são agrupamentos obtidos com o CFCM, sobre a coleção de documentos filtrada grupo a ser expandido.

Seja então $D = \{d_1, d_2, \dots, d_n\}$ um conjunto com n documentos, então o HFCM executa sobre D o algoritmo FCM produzindo o primeiro nível da hierarquia o método. Como resultado é gerada uma partição fuzzy $U_{c \times n}[1]$ e os protótipos $V[1] = \{v_1[1], v_2[1], \dots, v_c[1]\}$, onde $[1]$ representa o nível da hierarquia.

A expansão ocorre sempre nas folhas da hierarquia, e a decisão de expandir um determinado grupo, se dá pela avaliação do agrupamento, que é realizado através do índice de desempenho Q (2.22)(2.23), onde Q representa a qualidade dos protótipos gerados pelo agrupamento. De maneira que quanto melhor for um grupo mais próximo de zero será o resultado da medida de desempenho Q , e quanto maior for o desempenho pior será o grupo. Portanto o grupo que obtiver o maior valor de Q , será escolhido para a expansão condicional através do CFCM. Considere então j o grupo com maior valor de Q do nível l da hierarquia e

$$D_j[l] = \left\{ d_k \mid \mu(d_k, g_j[l]) \leq \frac{1/l}{c} \right\} \quad (2.21)$$

a coleção de documentos selecionados do grupo j com pertinência maior que a pertinência média. O agrupamento com o algoritmo CFCM é então executado sobre a coleção $D_j[l]$ produzindo c novos grupos para o nível $l + 1$ da hierarquia. Todo o processo se repete então para o nível $l + 1$ da hierarquia, e assim sucessivamente. Segundo os autores o ponto de parada da expansão hierárquica pode ser uma profundidade predefinida pelo usuário, com a estabilização das medidas de desempenho dos grupos ou supervisionada, de modo que o usuário observa a hierarquia que está sendo produzida e interrompe o processo quando desejar (PEDRYCZ; REFORMAT, 2006).

$$Q_j[1] = \sum_{d_i \in g_j} dist(d'_i, d_i)^2 \quad (2.22)$$

$$Q'_j[2] = \sum_{d_i \in g_j} dist(d'_i, d_i)^2 \quad (2.23)$$

$$d'_i = \sum_{h=1}^c \mu(d_i, g_h)[1]v_h[1] \quad (2.24)$$

$$d'_i = \sum_{h=1}^c \mu'(d_i, g_h)[j, 2]v_h[2] \quad (2.25)$$

Segundo (NOGUEIRA, 2013), os protótipos dos grupos representam uma versão condensada dos documentos agrupados, portanto d_i também pode ser representado pela combinação linear das pertinências de d_i com os protótipos, resultando em d'_i . Logo é esperado que d'_i seja o mais próximo possível do documento original d_i . Consequentemente, é utilizado essa noção para estimar a qualidade de um grupo através das equações 2.22 para o nível inicial da hierarquia e 2.23 nos demais níveis, onde Q calcula a soma total das distâncias dos documentos d_i de um grupo com d'_j .

A atualização dos protótipos no algoritmo CFCM ocorre da mesma maneira que o a equação (2.6) do FCM, contudo a função de pertinência é redefinida como sendo

$$\mu(d_i, g_h[l]) = \frac{\mu(d_i, g_j[l-1])}{\sum_{k=1}^c \left(\frac{\text{dist}(d_i, v_h[l])}{\text{dist}(d_i, v_k[l])} \right)^{\frac{1}{m-1}}}, 1 < h \leq c, d_i \in D_j[l-1], \text{eq}(2.27) \quad (2.26)$$

, cujo o valor de l corresponde ao nível da hierarquia e g_j seja o grupo expandido no nível $l-1$. Portanto percebe-se que a pertinência de um documento d_i em um grupo $g_h[l]$, será no máximo a pertinência do documento ao grupo que foi expandido. Com isso temos a restrição 2.10 do FCM adaptada no CFCM para

$$\sum_{h=1}^c \mu(d_i, g_h[l]) = \mu(d_i, g_j[l-1]), d_i \in D_j[l] \quad (2.27)$$

. Logo a soma das pertinências de um documento d_i no nível l da hierarquia terá que ser igual a pertinência desse documento no grupo $g_j[l-1]$ que foi expandido no nível anterior ($l-1$).

O pseudo código do método CFCM é apresentado no Algoritmo 4, de modo a deixar uma representação mais objetiva de como estruturar esses elementos. Enquanto no Algoritmo 5 consta o pseudo código do método HFCM, exemplificando como o mesmo reúne o FCM e o CFCM para produzir uma hierarquia de tópicos. Onde o critério de parada adotado foi a profundidade máxima da hierarquia, representado com o parâmetro $lmax$.

2.5 EXTRAÇÃO DE DESCRITORES

A tarefa de rotular grupos é um dos problemas chaves do agrupamento de textos, pois ao final do processo de agrupamento, os grupos precisam apresentar alguma relevância para o usuário (ZHANG; XU, 2008). Assim como pretende-se que os descritores escolhidos também sejam significativos para os documentos presentes no grupo a ser rotulado.

⁴Resultados obtidos baseados na implementação dos algoritmo FCM e PCM, produzida como parte este trabalho disponível em: <https://github.com/niltonvasques/fcm>

Data: $D_j[l]$, $U[l-1]$, l , c , m , ε
Result: $V[l]$, $U[l]$
 $G[l] \leftarrow [g_1, g_2, \dots, g_c];$
 $V_0[l] \leftarrow \text{inicializa-prototipos}(D, G);$
 $t \leftarrow 0;$
faça
 $U_t[l] \leftarrow \text{calcula com (2.26) usando } V_{t-1};$
 $V_t[l] \leftarrow \text{calcula com (2.6);}$
 $t \leftarrow t + 1;$
enquanto $\|V_{t-1}[l] - V_t[l]\| > \varepsilon;$
retorne $(U_t[l], V_t[l]);$

Algoritmo 4: Pseudo código do método CFCM

Data: D , c , m , ε , $lmax$
Result: Hierarquia
 $l \leftarrow 0;$
Hierarquia $\leftarrow G[l], V[l], U[l] \leftarrow \text{fcm}(D, c, m, \varepsilon);$ Algoritmo 1;
Hierarquia $\leftarrow \text{Hierarquia} + \{U[l], V[l]\};$
 $Q[l] \leftarrow \text{calcula desempenho dos grupos com a equação (2.22);}$
 $g_{max} \leftarrow \text{escolhe o grupo } g_j \text{ com maior valor de } Q;$
 $D_{max} \leftarrow \text{seleciona documentos de } g_{max} \text{ com equação (2.21);}$
 $l \leftarrow l + 1;$
faça
 $Q[l] \leftarrow \text{calcula desempenho dos grupos com a equação (2.23);}$
 $g_{max} \leftarrow \text{escolhe o grupo } g_j \text{ com maior valor de } Q;$
 $D_{max} \leftarrow \text{seleciona documentos de } g_{max} \text{ com equação (2.21);}$
 $G[l], V[l], U[l] \leftarrow \text{cfcm}(D_{max}, U[l-1], l, c, m, \varepsilon);$ Algoritmo 4;
 Hierarquia $\leftarrow \text{Hierarquia} + \{U[l], V[l]\};$
 $l \leftarrow l + 1;$
enquanto $l \leq lmax;$
retorne $(\text{Hierarquia});$

Algoritmo 5: Pseudo código do método HFCM

Essa etapa pode ser realizada manualmente, com o usuário guiando o processo, ou de forma automatizada, que por sua vez é mais interessante para a proposta de organização flexível de documentos. Uma vez que para grandes bases de dados textuais, a tarefa de rotular todos os grupos encontrados durante o agrupamento, pode ser bastante exaustiva para o usuário.

Dentre os métodos automatizados, é encontrado na literatura dois tipos de abordagens, uma baseada em conhecimento interno e a outra baseada em conhecimento externo([NOGUEIRA, 2013](#)). A primeira se utiliza somente de informações que podem ser obtidas na coleção de documentos, como por exemplo a frequência do termo, localização do termo na estrutura do documento. Enquanto a abordagem de conhecimento externo, levam em considerações também fontes de informação externas, para auxiliar a escolha dos termos mais representativos.

Em ambas abordagens a literatura fornece uma ampla gama de métodos, com o objetivo de obter bons descritores dos grupos. Os descritores podem ser extraídos com os termos mais frequentes no grupo, , no entanto o resultado pode ser genérico demais([TREERATPITUK; CALLAN, 2006](#)), ou os descritores podem ser extraídos dos grupos que estão mais próximos do centroide do grupo.

Contudo ([NOGUEIRA, 2013](#)) destaca que grande parte dos métodos de extração de descritores encontrados na literatura, são embutidos na fase de agrupamento. O que justifica a avaliação dos mesmos em função do desempenho do agrupamento. No entanto essa junção da extração de rótulos na fase de agrupamento, dificulta a combinação de diferentes técnicas de agrupamento e consequentemente a escolha de bons descritores. Logo os métodos onde a extração é realizada após a fase de agrupamento, de maneira independente, permitem uma melhor adaptação da proposta de organização flexível de documentos para diferentes contextos. Essa flexibilidade possibilitou que a investigação misturasse diferentes técnicas, permitindo obter melhores resultados.

2.6 CONSIDERAÇÕES FINAIS

Trabalhos relacionados a organização flexível de documentos e sistemas de recuperação de informação.

TRABALHOS RELACIONADOS

3.1 CONSIDERAÇÕES INICIAIS

A proposta de organização flexível de documentos está relacionada a vários campos de estudo, como ficou evidenciado na fundamentação teórica. Por isso a literatura existente para essa proposta é bastante rica e densa. Portanto com o propósito de otimizar a atividade de pesquisa e seleção do conhecimento científico produzido a respeito do tema, foram utilizadas algumas técnicas de revisão sistemática de literatura (*SLR – Systematic Literature Review*) utilizadas em (RIOS; MELLO, 2010). Com o objetivo de estabelecer critérios mais precisos na fase inicial da descoberta de conteúdo científico relacionado ao tema. Foi então adotada uma técnica comum ao método SLR, que consiste na elaboração de uma string de busca, usando operadores lógicos. Estabelecendo assim uma maneira mais objetiva para a obtenção de resultados relevantes a proposta dessa monografia. Portanto, levando em consideração os tópicos chaves e a proposta desse trabalho, foi construída a seguinte string de busca:

*(clustering OR "cluster label *" OR "cluster descriptors") AND fuzzy
AND (document OR "text mining" OR "document organization" OR
"soft document" OR "text data") (3.1)*

Devido o amplo acervo de publicações científicas presentes no repositório IEEEExplore¹, assim como também a possibilidade de se utilizar operadores lógicos e buscas parametrizadas. Foi realizado então uma busca no repositório IEEEExplore, restringindo o período de resultados entre os anos de 2010 e 2016, permitindo então que os resultados obtidos fossem mais recentes.

Com base nos resultados obtidos, foi realizada a leitura dos títulos e resumos dos artigos, com o propósito de descartar resultados com baixa relevância para essa pesquisa.

¹<http://ieeexplore.ieee.org/>

Durante a fase de leitura parcial dos resultados da busca, foram agrupados os artigos em três categorias: agrupamento fuzzy, extração de descritores e organização flexível de documentos. As publicações selecionadas e direcionadas para a categoria de agrupamento fuzzy, foram as que possuíam propostas de alteração de métodos de agrupamento existentes ou novos métodos. Enquanto artigos que tinham como conteúdo a análise dos termos de uma coleção, critério de seleção de termos ou atribuição de termos a grupos de documentos, foram agrupados na categoria de extração de descritores. Por fim, artigos mais gerais, propondo métodos ou realizando revisões de métodos, pertinentes ao processo de organização de documentos textuais, foram categorizados no grupo de organização flexível de documentos.

Para complementar os resultados obtidos foram adicionados artigos de alta relevância para o tema, e que apesar de serem antigos, ainda são amplamente citados em pesquisas recentes. Muitos desses artigos como é o caso do método FCM proposto em (BEZDEK *et al.*, 1984), são pilares fundamentais para o tema.

Nas próximas seções contém a revisão das pesquisas selecionadas, onde é elucidado os pontos chaves de cada pesquisa, a definição das propostas contida nos artigos e por fim a conexão com o objetivo dessa monografia.

3.2 ORGANIZAÇÃO FLEXÍVEL DE DOCUMENTOS

Após a proposição da lógica fuzzy que se propunha a lidar com a incerteza e imprecisão em (ZADEH, 1965), foi possível a elaboração de diversos métodos que se utilizassem dos benefícios da lógica fuzzy e aplicassem a diversos problemas do mundo real. Este é o caso da organização de documentos, que por não ser uma tarefa precisa, necessita de uma certa flexibilidade no processo.

(MATSUMOTO; HUNG, 2010) informa que os mecanismos adotados em sistemas de recuperação de informação (SRI), tais como buscadores web, estão dispostos em dois grupos. Sendo que o primeiro tem como foco o usuário realizando a busca, a qual é comumente chamada de busca web personalizada. Nessa abordagem os resultados obtidos são ordenados de acordo com a relevância do resultado para o usuário. Para calcular essa relevância, os buscadores realizam tarefas de coleta de dados dos usuários e comparação das preferências com demais usuários do sistema. Enquanto na segunda abordagem os resultados da busca é categorizado em grupos, permitindo assim que o usuário decida em qual grupo ele pretende visualizar as informações. Por exemplo, quando um usuário pesquisar pelo termo java, os resultados poderiam ser agrupados nas seções: máquina virtual, linguagem java, programas em java, oracle e etc. Seguindo essa linha de categorização de resultados em SRIs, (MARCACINI; REZENDE, 2010) propõe uma abordagem de agrupamento incremental e hierárquico para construção dos tópicos dos documentos, a qual permite a atualização das categorias a medida que novos documentos são adicionados sem realizar a etapa de agrupamento novamente. É possível a visualização dessa abordagem de categorização hierárquica, através da ferramenta online Torch², dos autores do artigo.

Com o surgimento de várias tecnologias, como mídias sociais, computação ubíqua, internet das coisas e principalmente os dispositivos móveis, que ultrapassou os 7 bilhões

²<http://sites.labic.icmc.usp.br/torch/webcluster/>

de dispositivos³ no ano de 2015. Onde por sua vez, todas essas tecnologias produzem uma abundante quantidade de dados não estruturados, dificultando a tarefa de métodos de mineração de dados, e por consequência também os métodos de organização de documentos já existentes. A esse cenário é usualmente atribuído o nome de *Big Data*. Sendo assim novas pesquisas como (HAVENS et al., 2012) e (KUMAR et al., 2015) tem sido conduzidas, focadas em bases com imensas quantidades de dados. Segundo (HAVENS et al., 2012) existem duas abordagens principais para otimizar o agrupamento de dados que se encaixam na categoria *Very Large* (Tabela 3.1), a primeira consiste na técnica de agrupamento distribuído incremental e o agrupamento por amostragem progressiva ou aleatória. Nos métodos que usam a técnica de amostragem, primeiramente é selecionado uma amostra com os dados representativos da coleção, depois é realizado o agrupamento, e em seguida é generalizado o agrupamento para o restante dos dados. Um dos métodos mais populares baseado em amostragem é o algoritmo *generalized extensible fast FCM* (geFFCM)(HAVENS et al., 2012). O geFFCM utiliza amostragem progressiva para se obter uma versão reduzida dos dados, de maneira que a mesma preserve as características da base original. Porém segundo (HAVENS et al., 2012), a técnica de amostragem do geFFCM é ineficiente para dados na categoria *Very Large*, o que levou os autores a propor uma extensão do geFFCM com uma melhoria na forma de realizar a amostragem dos dados, utilizando uma metodologia de seleção aleatória.

Bytes	10^6	10^8	10^{10}	10^{12}	$10^{>12}$
"tamanho"	medium	large	huge	monster	very large

Tabela 3.1: Classificação das bases de dados de acordo com o seu tamanho(HAVENS et al., 2012)

De acordo com (DENG et al., 2010), a organização flexível de dados através do algoritmo FCM possui uma falta de estabilidade, pois como a inicialização do FCM depende da aleatoriedade, o resultado final do agrupamento pode variar a cada inicialização. Assim como os dados presentes em bases de dados textuais são de alta dimensionalidade. Os autores propuseram então um modelo de inicialização da partição que extrai da coleção medidas de peso, raio e objetos mais representativos para orientar a inicialização da partição inicial. A respeito do problema da dimensionalidade, (DENG et al., 2010) sugere a redução da matrix documentos x termos, usando uma medida estatística para avaliar a qualidade dos termos presentes na coleção, descartando assim os termos considerados de baixa qualidade e consequentemente reduzindo a largura da matriz.

(KARAMI et al., 2015) propõe um modelo para análise textual de documentos médicos. Um dos pontos interessantes propostos pelo autor é a utilização do agrupamento fuzzy na etapa de pré-processamento e ponderação dos termos, antes de realizar o agrupamento e classificação. O agrupamento fuzzy é aplicado a coleção de termos presentes na coleção, e ao contrário do agrupamento na etapa pós processamento, a pertinência ocorre da palavra

³Segundo o relatório do The Mobile Economy disponível em (http://www.gsamobileeconomy.com/GSMA_Global_Mobile_Economy_Report_2015.pdf), a quantidade de dispositivos móveis (smartphones e tablets) atingiu o total de 7,517 bilhões no ano de 2015.

a um tópico ou grupo, de maneira que termos com alta pertinência possuem significados semânticos mais próximos. Essa aproximação semântica é realizada com base em um vocabulário predefinido.

Conforme foi definido no capítulo 2, os algoritmos de agrupamento fuzzy apresentados são baseados na otimização de funções objetivos. Contudo, (CUNHA et al., 2012) descreve que a otimização de funções que possuam vários mínimos locais sem nenhuma tendência global, utilizando as clássicas técnicas de otimização, podem simplesmente não só demorar a convergir, como a convergência pode nunca ser encontrada, ou ainda convergir para um mínimo local. Portanto diante da necessidade de se otimizar funções com tais características, desenvolveu-se estratégias de otimização baseadas em heurísticas, os quais permitem uma análise das características globais da função, de maneira que se possa realizar sucessivas buscas do mínimo global. Entretanto métodos heurísticos não garantem a convergência para o mínimo global, porém de modo geral apresentam boas aproximações. Dentre estas, tem-se heurísticas evolutivas inspiradas no processo de adaptação dos seres vivos. Que é o caso dos algoritmos genéticos, que são métodos de busca de soluções probabilísticos inspirados nos mecanismos genéticos de adaptação dos seres vivos através da seleção natural. Assim sendo (JIANG et al., 2013) traz uma proposta para enriquecer a tarefa de agrupamento textual, baseada na combinação dos algoritmos genéticos, com o método de agrupamento FCM, para evitar o problema de convergência para mínimos locais. O autor destaca ainda que os algoritmos genéticos possuem grande potencial na realização de buscas paralelas globais, principalmente quando trata-se de grandes volumes de dados, com elevados requisitos de classificação e computação paralela, restrições as quais o algoritmo iterativo para otimizar a função objetivo do FCM não é capaz de atender. Este problema de convergência do FCM apresentado por (BEZDEK et al., 1984), ocorre quando algumas condições iniciais são satisfeitas, levando o FCM para convergir para um mínimo local. No artigo o autor explora as qualidades e deficiências do algoritmo *Immune Genetic Algorithm*(IGA), que é uma versão mais atual do algoritmo genético (GA), que garante a diversidade dos indivíduos, aprimorando assim os potenciais de busca pela solução ótima global. Porém o IGA demanda muito custo computacional para garantir a diversidade da população, assim como também existe uma dificuldade em estimar os parâmetros iniciais, levando o IGA a uma finalização precoce, ou uma execução indefinida. Por conta dessas deficiências é investigado o algoritmo *Partheno Genetic Algorithm*(PGA), o qual não necessita da inicialização da população inicial, e também não sofre do problema de finalização prematura do IGA. Portanto o autor propõe o método PGA-FCM, que utiliza o PGA para encontrar solução aproximada global da função de minimização do FCM, e em seguida utiliza o próprio FCM tendo como entrada a partição de saída do PGA, para encontrar a solução ótima, de maneira reduzir o tempo de convergência.

Segundo (SARANYA; ARUNPRIYA, 2014), está presente na literatura uma outra visão do agrupamento textual, que realiza o agrupamento das sentenças contidas nos documentos. Tal abordagem tem-se mostrado fundamental para por exemplo, evitar a sobreposição de informações em técnicas de sumarização de documentos. Assim como também pode ser útil para a obtenção de novas informações a partir de um conjunto de resultados em uma busca na web por exemplo. A sumarização de sentenças tem como

objetivo, produzir um sumário que contenha informações relevantes dos documentos. Onde por sua vez, estratégias de ordenação das sentenças(*ranking*) de acordo com sua relevância, medidas de similaridade de sentenças utilizando informações da semântica dos termos são utilizadas para cumprir o objetivo. O autor também descreve que o agrupamento fuzzy de sentenças, pode ser utilizado para realizar tarefas de análise de contradição do conteúdo com o assunto da coleção de documentos o qual este está inserido. De modo que se possa identificar dados ruidosos que não contribuem com a ideia geral da coleção.

Ainda seguindo essa linha de otimização dos resultados apresentados após uma busca em um SRI, (NOGUEIRA et al., 2012) cita que uma das principais limitações de um SRI, está na maneira rígida de interpretar as strings de busca do usuário. O que pode ocasionar que alguns documentos relevantes para o usuário, não sejam retornados na busca. O autor informa que a literatura usualmente aborda esse problema de duas formas, em que ambas se baseiam na reformulação das strings de busca. Sendo que a primeira procura reformular os termos de uma busca, levando em conta características semânticas dos termos, de modo a encontrar versões mais apropriadas, permitindo então que uma maior quantidade de documentos relevantes seja retornado pela busca. A segunda forma de abordar esse problema consiste em ir ajustando os termos da busca, a partir dos primeiros documentos encontrados.

Portanto, seguindo a abordagem de reformulação semântica, em (MURALI; DAMODARAM, 2015) é apresentado um sistema de recuperação de informação e organização flexível de documentos, de maneira a considerar a informação semântica contida nos termos, de maneira a tornar mais precisa a similaridade entre os documentos, assim como também atuar na reformulação da busca. A estratégia proposta pelos autores, usa identificação de sinônimos baseado em ontologias da WordNet, hiperônimos⁴ e hipônimos⁵, para realizar tarefas de desambiguação semântica entre os termos, durante o pré-processamento. Em seguida os autores utilizam o método de agrupamento PFCM, devido ao potencial de agregar as qualidades e minimizar as deficiências dos algoritmos FCM e PCM. Então a extração de termos descritores dos grupos é realizada, construindo-se um índice de termos para cada grupo. Um dos diferenciais apresentados na proposta de (MURALI; DAMODARAM, 2015) ocorre na fase final, quando o usuário realiza uma busca. As palavras chaves inseridas pelo usuário, são processadas na ontologia do WordNet visando encontrar palavras com significados similares. A partir dessas palavras resultantes é calculado a similaridade com os termos descritores dos grupos, e por final os documentos que estiverem no grupo que obteve a maior similaridade com a busca digitada pelo usuário são retornados.

Com o objetivo de apresentar uma alternativa a essas duas abordagens para prover flexibilidade em um SRI, (NOGUEIRA et al., 2012) propõe um método para gerenciar a precisão e incerteza inerentes a documentos textuais, ainda na fase de organização dos documentos. Evitando assim a dependência da busca e a participação do usuário no

⁴Palavras de sentido genérico, ou seja, possuem significado bem amplo. Por exemplo ferramenta é um hiperônimo de chave de fenda.

⁵Palavras de sentido específico, sendo a palavra que está abaixo de um hiperônimo na hierarquia. Por exemplo, chave de fenda é hipônimo de ferramenta.

processo. Portanto um dos diferenciais apresentados pelos autores é a possibilidade de obtenção dos termos de busca ainda na fase de organização de documentos, de maneira não supervisionada. O que traz uma grande independência ao processo, possibilitando a automatização da organização flexível de documentos com pouca ou nenhuma intervenção do usuário durante o processo. Os termos de busca encontrados na etapa de organização dos documentos, são os descritores dos grupos, que são utilizados para distribuir os documentos em tópicos ou temas, e desde que durante a etapa de organização é utilizado o agrupamento fuzzy, os documentos podem ser rotulados em mais de um tópico. O que por sua vez, essa abordagem fortalece ainda mais a flexibilidade na organização de documentos.

A extração de termos que representem bem os grupos obtidos na fase de agrupamento é de fundamental importância, pois é a partir dos termos extraídos que é possível a recuperação dos documentos através de uma consulta realizada por um usuário. Dada essa importância, (NOGUEIRA, 2013) conduziu uma investigação acerca dos principais métodos de extração de descritores, que resultou na proposta de um modelo de SRI para organização flexível de documentos, com a adição de três métodos de extração de descritores. Os métodos de extração de descritores propostos, se baseiam em medidas na clássica na literatura, que quantizam a efetividade da recuperação da informação de um SRI, as quais são a precisão, revocação e a medida-F. A medida de precisão procura identificar dentre os documentos recuperados a proporção de documentos relevantes. Enquanto a revocação calcula a taxa de documentos relevantes recuperados a partir dos documentos que são previamente definidos como relevantes durante uma consulta em um SRI. A metodologia de extração adotada pelo autor nos métodos propostos, é do tipo *Description Comes Last* (DCL), que realiza a extração de descritores separado do processo de agrupamento, em contraste com outros métodos de extração do tipo *Description Comes First* (DCF), no qual os descritores são extraídos na fase de pré-processamento ou simultaneamente com o agrupamento. (NOGUEIRA, 2013) destaca que utilizar métodos do tipo DCL, traz o benefício de tornar independente o algoritmo de extração de descritores do algoritmo de agrupamento utilizado.

O primeiro método proposto em (NOGUEIRA, 2013) é o SoftO-FDCL (*Soft Organization - Fuzzy Description Comes Last*), que tem como propósito extrair descritores de partições fuzzy de documentos, de maneira a flexibilizar um SRI. A avaliação dos termos candidatos a descritores é feita então com base na pertinência do documento ao qual o termo pertence, onde os termos só são considerados se o documento tiver pertinência maior que um dado limiar, definido como sendo $\delta = \frac{1}{c}$. Ao fazer isso o método penaliza os termos de documentos com baixa pertinência ao grupo e ao mesmo tempo considera os termos de documentos que pertencem também a outros grupos, de modo a conservar a flexibilidade que o agrupamento fuzzy proporciona. Como extensão ao SoftO-FDCL, foi também proposto o método Soft-wFDCL (*Soft Organization - weighted Fuzzy Description Comes Last*), onde é levado em consideração também o grau de pertinência do documento no cálculo de relevância do termo. A adição da pertinência na equação, é justificada como sendo um fator de ponderação da importância de um termo ao grupo. Deste modo os termos que forem considerados relevantes para compor os descritores, ou

seja os que estiverem a cima do limiar definido no método SoftO-FDCL, serão ponderados em função da pertinência.

Os dois métodos anteriores são aplicados a uma organização flexível de documentos organizados de modo *flat*. Portanto com a finalidade de adicionar flexibilidade também a uma organização de documentos hierárquica, como por exemplo a obtida no método de agrupamento HFCM, (NOGUEIRA, 2013) traz o método HSoftO-FDCL (*Hierarchical Soft Organization - Fuzzy Description Comes Last*), como outra extensão do SoftO-FDCL. A modificação realizada nesse método ocorre no limiar de corte dos termos, pois no algoritmo HFCM a pertinência de um documento em um dado grupo, é relativa a pertinência desse mesmo documento no grupo imediatamente superior na hierarquia, conforme foi definido na equação 2.27 na página 17. Assim sendo o limiar do SoftO-FDCL é redefinido como sendo a pertinência do documento d_i no grupo superior dividido pela quantidade de grupos, ou seja $\zeta = \frac{\mu(d_i, g_j[l-1])}{c}$, tal que c seja a quantidade de grupos e l o nível do grupo que está sendo extraído os descritores. Para os grupos do primeiro nível da hierarquia os descritores são extraídos usando o δ .

Em (YAN et al., 2013) é informado que em estudos recentes os métodos de co-agrupamento fuzzy, tem apresentados resultados superiores a extensões do FCM, e métodos de co-agrupamento *crisp*, para algumas bases de dados com alta dimensionalidade. O diferencial dos métodos de co-agrupamento fuzzy em relação ao agrupamento fuzzy clássico (FCM), está no agrupamento simultâneo de documentos e termos, assim como a literatura sugere a relevância desse método para categorização de dados com alta dimensionalidade (YAN et al., 2013). Esse potencial dos algoritmos de co-agrupamento é devido a a sua inerente característica de redução da dimensão dos dados, através do agrupamento também dos termos, que permite melhor capturar a estrutura dos dados. Entretanto base de dados com elevada sobreposição de termos entre os documentos, ou seja termos que aparecem em vários documentos, podem afetar significativamente os métodos de co-agrupamento, assim como a organização dos termos produzidos podem não condizer com a realidade dos dados (TJHI; CHEN, 2008). Assim sendo o método HFCR definido em (TJHI; CHEN, 2008), tenta contornar esses problemas, substituindo o *ranking* dos termos, pelo agrupamento fuzzy, gerando assim duas partições fuzzy, uma de termos e uma de documentos. Segundo os autores, essa estratégia permite o HFCR contornar os problemas comuns dos métodos de co-agrupamento, mantendo a capacidade melhor agrupar dados de alta dimensionalidade e ainda reduzindo a complexidade computacional em relação a outros algoritmos de co-agrupamento presentes na literatura. No entanto a função objetivo do HFCR possui complexidade de $O(c * n * k)$, que maior que a complexidade da função objetivo (2.11) do FCM clássico $O(c * n)$, onde c é o número de grupos, n é o número de documentos e k o número de termos distintos em toda coleção.

3.3 CONSIDERAÇÕES FINAIS

Neste capítulo foi apresentado pesquisas relacionados ao tema desta monografia. Onde foi possível observar a variedade de estratégias adotadas para aumentar a eficiência de sistemas de recuperação de informação quando utilizados no contexto da organização flexível

de documentos. As pesquisas aqui apresentados apresentam propostas de melhorias em todas as etapas presentes na organização flexível de documentos. Onde durante a fase de pré-processamento é possível utilizar abordagens semânticas para a compactação da quantidade de termos. Essa estratégia de avaliar a semântica das palavras também pode ser utilizada na etapa final durante a recuperação da informação. Observa-se também muitas estratégias concentradas na etapa de agrupamento, com inúmeras possibilidades de otimização. As pesquisas aqui apresentadas, exploram a adaptação dos algoritmos fuzzy existentes para o problema do *BigData*, ou propõe a utilizam de outras heurísticas no processo, como a adição dos algoritmos genéticos no agrupamento. Outra alternativa, é a extrapolação do agrupamento, para abarbar também os termos, realizando assim um melhor reconhecimento da estrutura dos grupos, porém no entanto essa extrapolação vem com uma carga de processamento adicional. Outro conjunto de autores focam seus esforços em melhorar a extração de descritores dos grupos obtidos, afinal mesmo com um bom agrupamento realizado, se não se obtiver bons termos para descrever os grupos, a etapa de recuperação dos documentos pode ficar prejudicada.

Portanto as pesquisas relacionadas ao tema, demonstram que a organização de documentos não é um problema já foi exaurido, e que não possui uma solução canônica. Consequentemente se percebe que existe bastante trabalho a ser feito para otimizar a organização flexível de documentos. No próximo capítulo então, será abordado as abordagens propostas como resultado da investigação conduzida nesse trabalho, assim como os resultados dos experimentos.

ABORDAGEM PROPOSTA

4.1 CONSIDERAÇÕES INICIAS

A organização de uma coleção de documentos em vários tópicos, de modo que exista sobreposição entre os grupos é um importante problema em sistemas de recuperação de informação (SRIs). Na literatura diversas estratégias são utilizadas visando otimizar a organização flexível de documentos, conforme foi abordado no capítulo anterior. Soma-se a isso o fato de que a maioria dos métodos que adicionam flexibilidade ao processo, como por exemplo os métodos de agrupamento, nem sempre são desenvolvidos com o foco em documentos textuais. Que conforme foi abordado ao longo do texto, possui características que acrescentam algumas dificuldades no processo, tais como a alta dimensionalidade dos dados, assim como também usualmente são armazenados de maneira não estruturada. E ainda com o crescente aumento do uso de tecnologias de produção de conteúdo, a quantidade de dados textuais alcança grandes volumes de dados o que os enquadra no contexto do *BigData*. Portanto esse cenário fortalece a importância de se conduzir pesquisas e investigações em torno da organização flexível de documentos. Entretanto não é esperado que um método de agrupamento seja totalmente adequado para todos os tipos de dados, incluindo os dados de alta dimensionalidade como os dados textuais (STEINBACH et al., 2003). Desta maneira esse capítulo tem como objetivo detalhar as contribuições desta monografia a organização flexível de documentos, através da investigação dos impactos de se utilizar a estratégia de se misturar o agrupamento fuzzy e possibilístico provida pelo algoritmo PFCM. Onde este método de agrupamento pretende resolver os problemas dos elementos equidistantes e dos grupos coincidentes, apresentados nas partições fuzzy e possibilística respectivamente.

Conforme observa-se no capítulo 2, o algoritmo PFCM produz duas partições, sendo uma fuzzy e outra possibilística, o que induziu o presente trabalho a propor duas extensões do método de extração de descritores SoftO-FDCL proposto por (NOGUEIRA, 2013), que leva em consideração apenas os valores de pertinências presentes na partição fuzzy. A primeira extensão denominada PDCL (*Possibilistic Descriptor Comes Last*), a qual

aborda uma nova estratégia de interpretação dos graus de compatibilidade da partição possibilística. Enquanto a segunda abordagem proposta é o método Mixed-PFDCL (*Mixed - Possibilistic Fuzzy Descriptor Comes Last*), que é uma extensão do PDCL, utilizando uma abordagem híbrida para mesclar as duas partições presentes no algoritmo PFCM assim como também ponderar as contribuições das partições com base nos parâmetros a e b do PFCM.

Na primeira sessão deste capítulo é apresentado informações das bases de dados utilizadas, com as suas características, origem e composição dos documentos. Nas sessões seguintes, iremos fazer um estudo sobre a interpretação dos graus de compatibilidade possibilísticos durante a extração de descritores. Em seguida é definido as propostas sugeridas por essa monografia, que derivam desse estudo. E por fim é apresentado os dados obtidos com os experimentos realizados.

4.2 INFORMAÇÕES DAS BASES DE DADOS

Na mineração de dados e consequentemente nos trabalhos relacionados a organização flexível de documentos, é comum se realizar a avaliação dos métodos propostos, conduzindo-se experimentos sobre bases de dados existentes na literatura com essa finalidade (ROSSI et al., 2013). Para isso, as bases precisam estar apresentadas de maneira estruturada. Assim sendo, nesta pesquisa foi adotado o formato *tf-idf* (2.1) como forma de estruturar os dados presentes nas bases, de modo a capturar a importância relativa dos termos nos documentos e na coleção. Cada coleção foi então disposta em dois arquivos, sendo que o arquivo com extensão *.data* contém n linhas, onde cada linha constitui a representação de um documento da coleção no formato *tf-idf*, para n igual a quantidade de documentos presentes na coleção, enquanto o que arquivo de extensão *.names* possui a descrição dos m termos existentes na coleção dispostos um por linha.

A base Opinosis¹ é composta de opiniões de consumidores a respeito das características de alguns produtos, obtidas dos portais amazon.com, tripadvisor e edmunds.com. As opiniões presentes na base, abordam tópicos como serviços de hospedagem, dispositivos eletrônicos e carros. Sendo que no total as sentenças presentes na coleção estão distribuídas em 51 categorias, onde cada categoria possui 100 sentenças na média. Os dados dessa base foram obtidos no repositório *UCI Machine Learning Repository* (FRANK; ASUNCION, 2010), que mantém várias coleções de dados que são utilizados pela comunidade de aprendizado de máquina para a realização de análises empíricas dos algoritmos.

A coleção de documentos 20Newsgroup² original contém aproximadamente 20000 documentos de notícias, particionados em mais ou menos 20 temas. No entanto para os experimentos realizados nesta pesquisa, foi utilizado uma versão mais compacta da coleção, contendo 2000 documentos pertencentes ao tema ciência, a qual contém os tópicos sci.space, sci.electronics e sci.med. Esta base tem-se mostrado bastante popular em aplicações textuais de aprendizado de máquina (NOGUEIRA et al., 2015), tais como agrupamento e classificação de textos. Essa base foi coletada originalmente por

¹ <http://archive.ics.uci.edu/ml/datasets/Opinosis+Opinion+26frasl3B+Review>

Ken Lang para a pesquisa Newsweeder apresentada em (LANG, 1995).

Os documentos presentes na base de dados Reuters-21578³ apareceram inicialmente na Reuters newswire em 1987. Sendo que os documentos foram coletados e indexados diretamente por membros da Reuters e da *Carnegie Group, Inc.* também em 1987 para o desenvolvimento do CONSTRUE(HAYES; WEINSTEIN, 1990), que foi um sistema de categorização de documentos. Onde no ano de 1990 essa base de dados foi tornada pública pela Reuters, para ser utilizada em pesquisas de recuperação de informação. No entanto as versões iniciais dessa base continham documentos repetidos e ambíguos, o que motivou um grupo de pesquisadores de categorização textual durante a conferência *ACM SIGIR '96*, a realizar uma limpeza na base, possibilitando uma melhor comparação dos resultados entre diferentes estudos. Essa versão final ficou com o total de 21578 documentos, distribuídos entre 43 diferentes categorias. Contudo nesta pesquisa foi utilizada uma amostragem menor da coleção, contendo 1052 documentos, selecionados aleatoriamente de cada classe da coleção.

A base de dados WAP(*WebACE Project*) é composta de um conjunto de páginas web coletadas por Moore et al. (1997), para um projeto de pesquisa de agrupamento, seleção e recuperação de páginas web. Os dados presentes nesta coleção foram obtidos pelos autores do artigo em 98 páginas web, onde posteriormente foram distribuídos em 20 diferentes categorias, que abrangem tópicos como negócios e finanças, tecnologias, trabalho e indústria. O conteúdo obtido está disposto em 1560 documentos na sua versão original, sendo que todos os documentos foram utilizados nessa pesquisa.

A coleção de documentos NSF⁴(*National Science Foundation*) foi obtida do repositório de dados para pesquisas de aprendizado de máquina *UCI Machine Learning Repository*(FRANK; ASUNCION, 2010). O conteúdo dos dados presentes na base é composto de 129000 resumos, sendo um resumo por documento, descrevendo prêmios da NSF para pesquisas básicas. Para os experimentos descritos nesta monografia, foram selecionados 1600 documentos de maneira aleatória entre as categorias apresentadas na coleção.

A base de dados Hitech adquirida em Karypis (2015), é parte de uma coleção de bases da conferência TREC(*Text REtrieval Conference*)⁵. Esta base é composta de um conjunto de notícias da revista *Jose Mercury News*⁶, as quais são distribuídos em categorias distintas. As notícias presentes na coleção abordam temas como computadores, eletrônicos, saúde, medicina, pesquisa e tecnologia. A base originalmente possui 2301 documentos, onde para esta pesquisa foram selecionados uma amostragem aleatória contendo 600 documentos.

Outro aspecto não menos importante, são as características particulares das coleções de dados. Pois ressalta-se que para uma mais apurada análise dos resultados, é pertinente considerar as particularidades de cada base, com a finalidade de encontrar possíveis justificativas para os resultados apresentados, realizando-se indagações comparativas as peculiaridades sabidamente conhecidas dos métodos analisados. Portanto o conjunto de

²<http://qwone.com/~jason/20Newsgroups/>

³<https://archive.ics.uci.edu/ml/datasets/Reuters-21578+Text+Categorization+Collection>

⁴<https://archive.ics.uci.edu/ml/datasets/NSF+Research+Award+Abstracts+1990-2003>

⁵<http://trec.nist.gov/data.html>

⁶<http://www.mercurynews.com/>

documentos	número de documentos presentes na coleção
termos	número de termos existentes na coleção após o pré-processamento
% zeros	número relativo de zeros na <i>tf-idf</i> , ou seja quantifica o quanto a matriz é esparsa
classes	número de classes presentes na coleção
n-gramas	quantidade de termos considerados sequencialmente na coleção

Tabela 4.1: Descrição das características objetivas presentes em coleções textuais elencadas para este trabalho

características particulares de cada base obtidos em [Rossi et al. \(2013\)](#) e adaptados a esta pesquisa, dar-se à como apresentado na Tabela (4.1).

Portanto, uma análise objetiva das características presentes nas seis bases descritas anteriormente está apresentado na (4.2). Onde é possível notar de maneira bem objetiva ao se observar a coluna % zeros da tabela, que todas as bases apresentam uma quantidade de zeros em mais de 90% dos dados, o que caracteriza o peculiar problema dos dados esparsos já caracterizado ao longo do texto, como algo inerente aos dados textuais.

nome	docs	termos	classes	% zeros	n-gramas
Opinois	51	842	3	95,73%	1-grama
20newsgroups	2000	11028	4	99,11%	1-grama
Hitech	600	6925	6	97,93%	1-grama
NSF	1600	2806	16	99,76%	1-grama
WAP	1560	8070	20	98,51%	1-grama
Reuters-21578	1052	3925	43	98,55%	1-grama

Tabela 4.2: Características das bases de dados utilizadas nesta pesquisa

4.3 REFINAMENTO COM O ALGORITMO PFCM

Conforme ficou evidenciado, a tarefa de organizar de maneira flexível um conjunto de documentos textuais, possui diversos desafios. Em particular, ao se agrupar um conjunto de documentos é esperado que os grupos resultantes possuam significado relevante, ou seja o algoritmo de agrupamento precisa detectar a estrutura natural dos dados ([STEINBACH et al., 2003](#)). Alguns desses desafios está na dificuldade em escalar os métodos usuais para bases de dados na categoria *Very Large* conforme a escala apresentada na Tabela 3.1, assim como também a obtenção de mecanismos efetivos para se avaliar a qualidade dos grupos produzidos, técnicas para se medir a interpretabilidade dos resultados, capacidade para estimar os parâmetros dos algoritmos, possibilidade para funcionar de maneira incremental, reduzindo o custo computacional durante a atualização dos grupos com novos dados, e também a capacidade de continuar a produzir bons resultados em cenários compostos de documentos ruidosos ([CARVALHO et al., 2016](#)).

Portanto para Steinbach et al. (2003):

[...] there is no reason to expect that one type of clustering approach will be suitable for all types of data, even all high dimensional data. Statisticians and other data analysts are very cognizant of the need to apply different tools for different types of data, and clustering is no different.

Diante então dos desafios propostos, e com a evidência de que é possível aprimorar os resultados, ao se utilizar novas estratégias de agrupamento. A investigação apresentada nesta seção tem como objetivo analisar de qual forma a organização de documentos pode ser otimizada, ao aplicar na etapa de agrupamento uma estratégia que misture as partições possibilística e fuzzy, através do algoritmo PFCM. A escolha desse algoritmo foi feita devido o seu potencial para absorver as qualidades presentes no FCM contrabalanceando as suas deficiências ao agregar também o PCM e sua partição possibilística. Outro ponto a se considerar são as diversas pesquisas na literatura abordando o desempenho do PFCM, como por exemplo em Pal et al. (2005), Yan e Chen (2009), Subhashini e Kumar (2010), Grover (2014), Popescu et al. (2015).

Com isso foi conduzido um experimento adaptando a estratégia de organização flexível de documentos definida em Nogueira (2013), utilizando na etapa de agrupamento o método PFCM. Porém esse método produz duas partições uma possibilística e uma partição fuzzy. Desse modo foi aplicado o método de extração de descritores SoftO-FDCL na partição fuzzy e outra vez na partição possibilística, produzindo assim dois grupos de descritores. Essa adaptação está ilustrada na Figura (4.1).

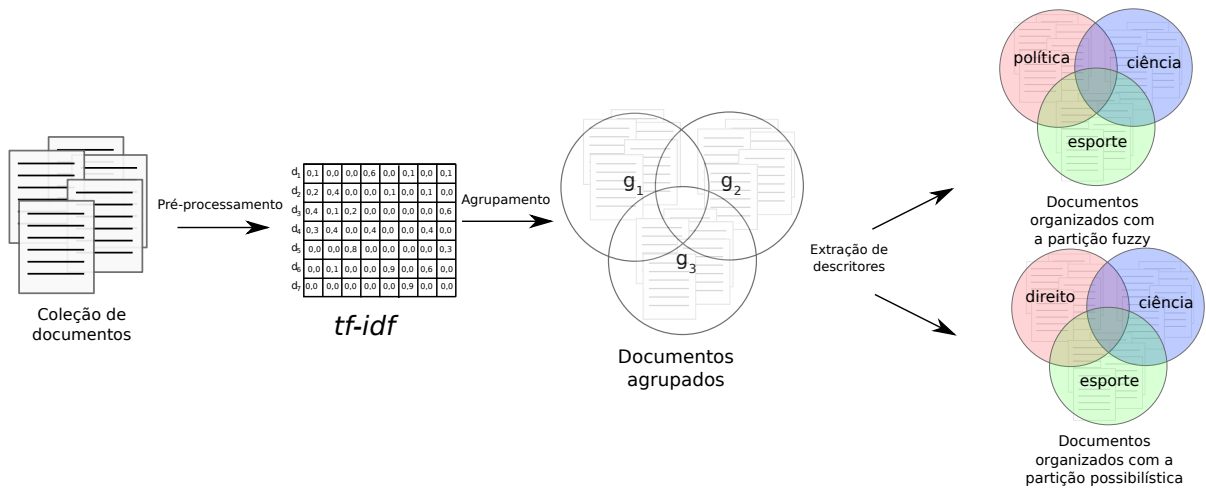


Figura 4.1: Estratégia de organização flexível de documentos adotada ao se misturar abordagens fuzzy e possibilísticas no agrupamento

Para se calcular a quantidade ótima de grupos, para cada base de dados foi utilizado o método da silhueta fuzzy (equação 2.5), método bastante utilizado com o propósito de avaliar o agrupamento de documentos. Assim sendo, o número ideal de grupos é determinado após a execução da silhueta fuzzy variando o número de grupos entre 2 e o número de classes de cada coleção. Ressalta-se que em coleções que os dados não possuem

rótulos, ou seja o número de classes é desconhecido, ainda é possível usar o método da silhueta fuzzy para definir o número ótimo de grupos. No entanto, a quantidade máxima de grupos deve ser definida de modo empírico ou com base em alguma informação prévia a respeito dos dados.

Para permitir uma análise comparativa dos resultados, o experimento foi realizado também com os algoritmos FCM e PCM. Como resultado do agrupamento das bases, está disposto na tabela 4.11, a comparação do número de grupos obtidos no experimento. Nessa comparação nota-se que os algoritmos FCM e PFCM foram os que alcançaram uma quantidade de partições mais próxima da quantidade de classes existentes em cada coleção. Enquanto o PCM manteve uma tendência a produzir uma quantidade menor de grupos em relação aos demais.

Coleção	# classes	FCM	PCM	PFCM
Opinosis	3	3	3	3
20Newsgroup	4	2	2	2
Hitech	6	6	5	5
NSF	16	11	2	16
WAP	20	14	5	16
Reuters-21578	43	22	11	36

Tabela 4.3: Quantidade ótima de grupos determinada através do método da silhueta fuzzy para cada algoritmo de agrupamento

Após agrupar os dados utilizando os métodos FCM, PCM e PFCM, foi aplicado o método de extração de descritores SoftO-FDCL. Para nos permitir avaliar os descritores produzidos, foi verificado o potencial preditivo dos mesmos, possibilitando assim quantificar a qualidade dos termos selecionados para nomear os grupos.

A avaliação do potencial preditivo dos descritores foi realizada, realizando se uma defuzzificação dos grupos produzidos pelo agrupamento, ou seja, se durante o agrupamento foi gerado o conjunto de grupos $G = \{g_1, g_2, \dots, g_c\}$, temos então o conjunto de grupos *crisp* $C' = \{crisp_1, crisp_2, \dots, crisp_c\}$, onde cada $crisp_i$ corresponde ao grupo g_i . A função de defuzzificação adotada para se converter as as partições fuzzy e possibilística, as quais permitem que um documento pertença a um ou mais grupos, considera o grupo *crisp* de um documento d_i , como sendo o grupo $crisp_j$ do respectivo *grupo* _{j} , no qual d_i possua a maior pertinência/tipicidade. Esta definição está formalizada na equação 4.1.

$$crisp(d_i) = \begin{cases} crisp_j, & \mu(d_i, g_j) = \max_{\forall g \in G} \mu(d_i, g), \text{ se a partição for fuzzy} \\ crisp_j, & \lambda(d_i, g_j) = \max_{\forall g \in G} \lambda(d_i, g), \text{ se a partição for possibilística} \end{cases} \quad (4.1)$$

Após se atribuir os documentos aos grupos *crisp*, é produzida uma outra matriz, considerando apenas os termos descritores dos grupos. Logo essa matriz documentos x descritores $D'_{n \times m}$, é uma versão condensada da matriz documentos x termos $D_{n \times k}$, onde seja n a quantidade de documentos, k a quantidade de termos e m a quantidade

descritores. O conteúdo dessa matriz condensada, assim como na matriz original, também é a frequência dos descritores nos documentos.

Visando então avaliar a qualidade dos descritores e permitir uma comparação direta dos impactos dessa abordagem com os resultados publicados em [Nogueira \(2013\)](#) e [Nogueira et al. \(2015\)](#). Submeteu-se a matriz D' aos algoritmos de classificação SVM, Naive Bayes, Multinomial Naive Bayes, KNN e C4.5, que são bem comuns na avaliação de métodos de aprendizado de máquina, e foram os mesmos utilizados em [Nogueira et al. \(2015\)](#).

Nesse contexto, foi utilizado a implementação dos algoritmos de classificação citados anteriormente, presentes na ferramenta WEKA ([HALL et al., 2009](#)). Os algoritmos Naive Bayes (NB), Multinomial Naive Bayes (NB-Multinomial) e o J48 (que é a implementação do C4.5 existente no WEKA), foram executados com os parâmetros padrão da ferramenta. Por outro lado o SVM foi ajustado para usar o *Normalized Polynomial Kernel* com o parâmetro de complexidade sendo $c = 2.0$. O algoritmo IBk (implementação do KNN presente no WEKA) foi executado 7 vezes, variando o parâmetro de vizinhos de 1 até 7, sendo escolhido o melhor resultado. Ressalta-se que foi adotada a técnica *10-fold cross validation* no experimento para melhor capturar a capacidade de generalização do modelo. O resultado dessa classificação está apresentado nas figuras 4.8, 4.9, 4.10, 4.11, 4.12 e 4.13.

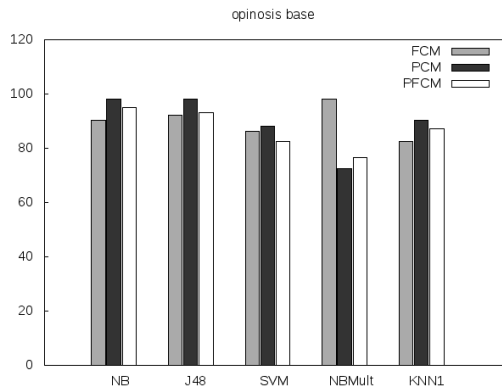


Figura 4.2: Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção Opinois

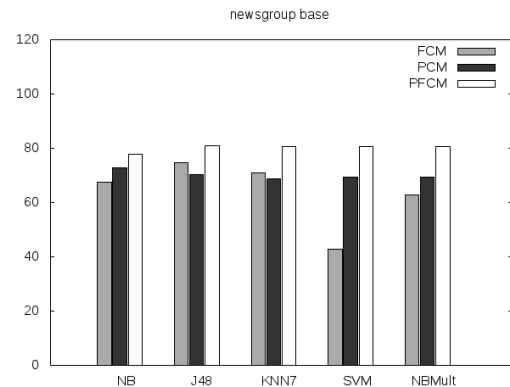


Figura 4.3: Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção 20Newsgroup

O resumo dos resultados do desempenho dos descritores extraídos após o agrupamento com os algoritmos FCM, PCM e PFCM é apresentado na Tabela 4.12. Na tabela, a marcação (✓) denota qual método de agrupamento obteve a maior taxa de classificação dentre os demais.

Esses resultados obtidos reforçam a flexibilidade e adaptação do método SoftO-FDCL ([NOGUEIRA, 2013](#)), a novos algoritmos de agrupamento. De modo que o método se demonstra promissor na tarefa extrair termos relevantes dos grupos produzidos na etapa

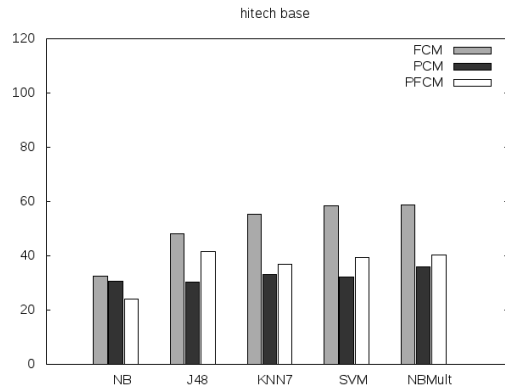


Figura 4.4: Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção Hitech

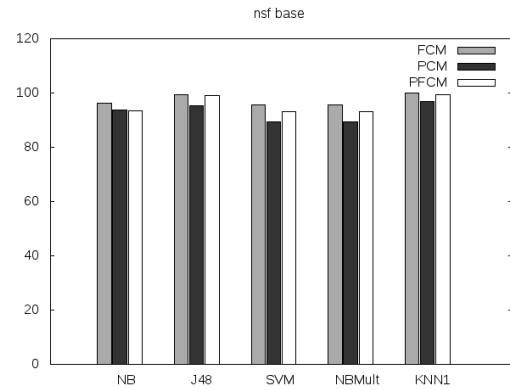


Figura 4.5: Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção NSF

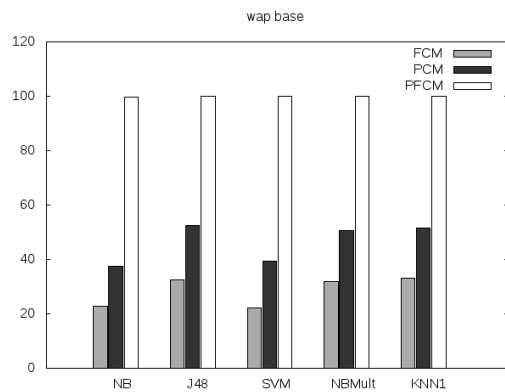


Figura 4.6: Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção WAP

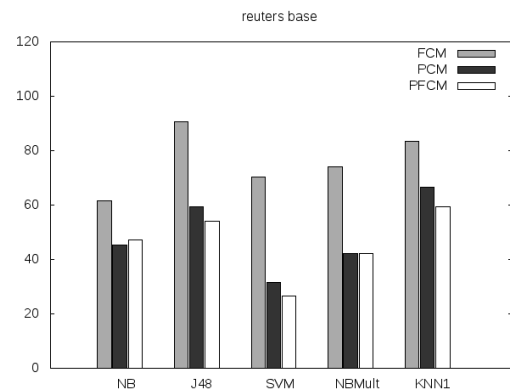


Figura 4.7: Desempenho obtido com os descritores extraídos com o algoritmo SoftO-FDCL a partir dos métodos de agrupamento FCM, PCM e PFCM executados na coleção Reuters-21578

nome	docs	termos	classes	% zeros	FCM	PCM	PFCM
Opinosis	51	842	3	95,73%		✓	
20newsgroups	2000	11028	4	99,11%			✓
Hitech	600	6925	6	97,93%	✓		
NSF	1600	2806	16	99,76%	✓		
WAP	1560	8070	20	98,51%			✓
Reuters-21578	1052	3925	43	98,55%	✓		

Tabela 4.4: Sumário dos resultados da classificação dos descritores

de agrupamento, e isso é demonstrado através do potencial preditivo evidenciado na Tabela 4.12, com as taxas máximas de classificação de mais de 80% para quase todas as bases, com exceção da base Hitech, a qual obteve a taxa máxima de 58.67%. Por outro lado, ressalta-se a importância também de avaliar de maneira subjetiva os descritores selecionados dos grupos, pois isso nos permite compreender melhor se os termos obtidos fazem sentido.

método	<i>crisp₁</i>	<i>crisp₂</i>	<i>crisp₃</i>
FCM	easy, clear, drive, display, control, car, version, nice, work, perfect	fact, import, isn't, model, problem, unit, design, don't, doesn't, found	breakfast, nearby, concierge, eat, bottle, coffee, floor, food, inn, friendly
PCM	easy, read, problem, version, don't, small, nice, car, work, found	fact, back, turn, expect, size, close, quality, review, min, feature	feel, amazing, isn't, extreme, drive, include, point, reason, give, run
PFCM μ	easy, drive, control, don't, version, nice, car, work, perfect, lot	fact, isn't, read, complete, device, display, size, doesn't, found	breakfast, nearby, pleasant, concierge, eat, coffee, floor, clean, friendly, food
PFCM λ	club, immaculate, send, towel, basic, exception, spotl, pillow, typical, fridge	pub, housekeep, holiday, tourist, tea, smoke, pm, renovate, facilitate, london	usual, central, forum, bottle, modern, adult, supply, food, reserve, dinner

Tabela 4.5: Descritores extraídos com os métodos de agrupamento FCM, PCM e PFCM da coleção Opinosis, onde μ e λ se referem as partições fuzzy e possibilística respectivamente, da qual os descritores foram extraídos.

Para realizarmos uma análise subjetiva dos resultados, foi escolhido os descritores obtidos da base de dados Opinosis, por possuir poucos grupos e assim facilitar a análise e a visualização. A coleção Opinosis contém opiniões dos usuários a respeito de serviços de hospedagem, dispositivos eletrônicos e carros. Portanto na Tabela 4.5 temos a seleção de termos descritores escolhidos para cada grupo, pelos respectivos métodos de agrupamento. Ao analisarmos os termos selecionados é possível notar uma tendência geral, do grupo *crisp₁* conter palavras relacionadas a carros, o *crisp₂* abordando opiniões dos usuários sobre dispositivos eletrônicos e o grupo *crisp₃* com opiniões a respeito dos serviços de hospedagem e alimentação. Contudo nota-se que os descritores do PCM e do PFCM λ (descritores da partição possibilística do PFCM) estão um pouco mais misturados, não apresentando uma tendência geral bem definida. Uma explicação possível a esse resultado, pode se encontrar na própria partição possibilística a qual permite que um mesmo documento possua um grau de tipicidade elevado em todos os grupos. Portanto uma solução possível, pode ser uma adaptação do método de extração de descritores SoftO-FDCL voltado para a partição possibilística, assim como também para algoritmos híbridos com

duas partições, que é o caso do PFCM.

De maneira complementar, é importante salientar que o método de extração de descritores SoftO-FDCL é totalmente influenciado pelos valores contidos nas partições fuzzy e possibilística. Portanto, é também importante se realizar uma análise dos métodos de agrupamento utilizados no experimento, com a finalidade de se entender qual método é mais apropriado em determinados contextos. Nesse contexto os resultados apontam que a dimensionalidade das bases foi um fator determinante no desempenho dos métodos de agrupamento no contexto da organização flexível de documentos, e consequentemente a extração de descritores. Por exemplo, do sumário de resultados apresentados na Tabela 4.12, podemos observar que o método PCM obteve o melhor resultado na coleção Opinosis, que possui a menor dimensionalidade (842 termos), enquanto que o algoritmo FCM superou os demais métodos nas bases NSF (2806 termos), Reuters-21578 (3925 termos), Hitech (6925 termos), e por fim o algoritmo PFCM atingiu melhores resultados para as bases WAP (8070 termos) e 20Newsgroup (11028 termos), que são as bases de maior dimensionalidade na coleção.

Na próxima sessão, motivado pelos resultados desse experimento será explorado uma adaptação do método SoftO-FDCL para evitar o processo de extração dupla de descritores em algoritmos que possuam partições fuzzy e possibilística.

4.4 UMA ABORDAGEM HÍBRIDA PARA EXTRAÇÃO DE DESCRITORES

No experimento anterior identificamos um possível problema ao realizar a extração dos descritores de maneira separada em cada partição do PFCM, assim como também foi apontado que o método pode não capturar toda essência da partição possibilística, que difere da partição fuzzy do FCM por não possuir a restrição que obriga a soma das pertinências de um grupo ser igual a um (equação 2.9). Logo, é intuitivo indagar, que para uma melhor interpretação dos grupos produzidos em um método de agrupamento híbrido, seja pertinente utilizar também uma abordagem mista de extração de descritores. Aproveitando-se assim dos benefícios existentes na partição possibilística, a qual penaliza os elementos ruidosos, com baixos valores de tipicidade, sem abrir mão das vantagens presentes na partição fuzzy. Para isso é necessário se compreender os mecanismos de funcionamento do método SoftO-FDCL, para que seja possível propor uma adaptação para este contexto.

4.4.1 Investigações na extração de descritores em partições possibilísticas

O método SoftO-FDCL (*Soft Organization - Fuzzy Description Comes Last*) proposto em Nogueira (2013), é baseado em uma adaptação das medidas clássicas de Recuperação de Informação (RI), para quantificar a relevância dos termos candidatos a descritores dos grupos obtidos na etapa de agrupamento, utilizando a informação de pertinência ou tipicidade advinda do algoritmo de agrupamento. Esse grau de compatibilidade entre o documento e o grupo, desempenha um papel fundamental na seleção dos termos candidatos a descritores de um determinado grupo, pois através deste, é possível penalizar os termos de documentos que possuam baixo grau de compatibilidade com grupo o qual o

descriptor está sendo extraído.

Para realizar a extração dos descritores, inicialmente todos os termos que permanecem na coleção após a etapa de pré-processamento são considerados como candidatos a descritores. Posteriormente o método realiza uma avaliação quantitativa da relevância de cada termo t_k para um grupo g_j , utilizando a medida $f1$ apresentada na equação 4.4, que é a média harmônica da precisão (equação 4.2) e da revocação (equação 4.3). A medida de precisão checa a quantidade de documentos significantes entre os documentos recuperados. Enquanto a medida de revocação calcula a proporção de documentos relevantes recuperados entre todos os documentos relevantes da coleção. Tanto a medida de precisão e de revocação, tomam como base as informações obtidas a partir da matriz de contingência apresentada na Tabela 4.6. Adicionalmente tem-se que um documento d_i é considerado como parte do grupo g_j , caso $\mu(d_j, g_j) \geq \delta$ para valores de pertinências ou $\lambda(d_i, g_j) \geq \delta$ para partição possibilística, onde seja $\delta = \frac{1}{c}$ e c a quantidade de grupos. O limiar δ é uma parte relevante do método SoftO-FDCL, pois ele possibilita se considerar como candidatos a descritores, os termos presentes em documentos que pertençam a mais de um grupo, ao mesmo tempo que também penaliza os termos presentes em documentos com baixa compatibilidade em um grupo.

Portanto, para efetuar a seleção dos termos descritores, é construído um *ranking* dos termos candidatos de cada grupo, ordenados pela pontuação obtida com a medida $f1$ (equação 4.4). A partir dessa pontuação, é selecionado os termos com as maiores pontuações em cada grupo. Nogueira (2013) destaca, que a quantidade de descritores a ser selecionada fica a critério do usuário.

	Documentos do grupo g_j com grau de compatibilidade maior ou igual a δ	Documentos do grupo g_j com grau de compatibilidade menor do que δ
Documentos que possuem o descriptor candidato t_k	<i>ganhos</i>	<i>ruídos</i>
Documentos que não possuem o descriptor candidato t_k	<i>perdas</i>	<i>rejeitos</i>

Tabela 4.6: Matriz de contingência do termo t_k para o grupo g_j para as medidas de recuperação de informação

$$precisão(t_k, g_j) = \frac{ganhos}{ganhos + ruídos} \quad (4.2)$$

$$recuperação(t_k, g_j) = \frac{ganhos}{ganhos + perdas} \quad (4.3)$$

$$f1(t_k, g_j) = \frac{2 * precisão(t_k, g_j) * recuperação(t_k, g_j)}{precisão(t_k, g_j) + recuperação(t_k, g_j)} \quad (4.4)$$

Em síntese o método SoftO-FDCL, cria uma tabela de pontuação dos termos candidatos aos grupos, e seleciona os que obtiverem a maior quantidade de pontos. Bem como a

promoção a termo candidato é feita considerando os termos que ocorrem em documentos com grau de compatibilidade maior que o limiar $\delta = \frac{1}{c}$. Ao se considerar esse limiar em em partições com graus de pertinência, como a partição fuzzy do FCM, esse limiar δ se mostra adequado, pois como foi definido no capítulo 2, o grau de compatibilidade do FCM é restrito a equação 2.9. Portanto ao filtrar os documentos em uma partição de pertinências utilizando δ como critério de corte, na maioria das vezes será descartado algum documento no processo.

Formalizando essa percepção, podemos então definir as duas propriedades que derivam dessa discussão, as quais são as equações (4.5) e (4.6), onde seja c a quantidade de grupos. A primeira propriedade (equação 4.5) expressa que se um documento possuir pertinência maior do que o limiar δ em um grupo g_1 qualquer, obrigatoriamente existirá ao menos um outro grupo g_2 no qual esse mesmo documento terá pertinência inferior ao limiar δ . Portanto, essa propriedade nos indica que na maioria das vezes um ou mais documentos serão descartados na análise dos termos candidatos do grupo, o que reforça a adequação desse limiar para a partição de pertinências. Por outro lado, a segunda propriedade (equação 4.6) denota o único caso particular, no qual um documento d_i não será descartado em nenhum grupo. Isto ocorre apenas se d_i possuir pertinência igual em todos os grupos, o que só acontece em dados ruidosos, que apresentam o problema do elemento equidistante detalhado na Figura 2.3 do capítulo 2.

$$\mu(d_1, g_j) > \delta \rightarrow \exists \mu(d_i, g_2) < \delta \quad (4.5)$$

$$(\mu(d_1, g_1) = \delta) \wedge (\mu(d_i, g_2) = \delta) \wedge \dots \wedge (\mu(d_i, g_{c_1}) = \delta) \rightarrow (\mu(d_i, g_c) = \delta) \quad (4.6)$$

Contudo, para a partição possibilística, essas duas propriedades não são satisfeitas. Isso se deve, por conta da remoção da restrição da equação (2.9), o que possibilita o grau de compatibilidade da partição possibilística variar de maneira independente entre o intervalo de $[0, 1]$, sem ser influenciado pelo grau de compatibilidade do documento nos demais grupos.

Para tornar claro essas análises, vamos considerar uma situação onde temos uma coleção de textos com 3 documentos (Tabela 4.7), onde cada documento possui 3 termos. Essa coleção de documentos foi então agrupada, usando o método PFCM, o qual produziu 2 grupos, com as suas respectivas partições de pertinência (Tabela 4.8) e possibilística (Tabela 4.9).

	t_1	t_2	t_3
d_1	0	0	1
d_2	1	1	0
d_2	1	1	0

Tabela 4.7: Exemplo de matriz documentos x termos

A partir desse agrupamento podemos então aplicar a extração de descritores, utilizando as partições apresentadas nas Tabelas 4.8 e 4.9. Seguindo então o procedimento do método SoftO-FDCL, inicialmente devemos considerar todos os descritores presentes

	g_1	g_2
d_1	0.5	0.5
d_2	0.3	0.7
d_3	0.3	0.7

Tabela 4.8: Exemplo de matriz documents x grupos com graus de pertinência

	g_1	g_2
d_1	0.7	0.7
d_2	0.5	0.8
d_3	0.5	0.9

Tabela 4.9: Exemplo de matriz documents x grupos com graus de possibilidade

na Tabela 4.7, como candidatos a descritores. Então para promovermos os descritores candidatos do grupo g_1 utilizando a partição de pertinências, devemos observar os documentos que possuem pertinência maior ou igual ao limiar, onde no nosso exemplo é igual a $\delta = \frac{1}{2} = 0.5$. Portanto o único documento que se enquadra no filtro do limiar é o documento d_1 , portanto o termo t_3 , que é o único presente no documento d_1 , é promovido a descriptor candidato e consequentemente a descriptor do grupo g_1 . Já o grupo g_2 possui 3 documentos com pertinência maior ou igual ao limiar δ . Sendo assim, os termos de d_1 , d_2 e d_3 são promovidas a descritores candidatos do grupo g_2 . Ao calcularmos a medida $f1$ para cada um deles temos: $f1(t_1, g_2) = 0,8$, $f1(t_2, g_2) = 0,8$, $f1(t_3, g_2) = 0,5$. Logo os descritores selecionados para cada grupo foram $g_1 = \{t_3\}$ e $g_2 = \{t_1, t_2\}$.

Agora ao aplicarmos novamente a extração de descritores sobre a partição possibilística, temos agora os documentos d_1 , d_2 e d_3 sendo considerados parte do grupo g_1 , de acordo com a regra do limiar. Logo devemos considerar os termos dos 3 documentos como descritores candidatos. Portanto, seguem os valores de $f1$ para os 3 termos candidatos do grupo g_1 : $f1(t_1, g_1) = 0,8$, $f1(t_2, g_1) = 0,8$, $f1(t_3, g_1) = 0,5$. Para o grupo g_2 vamos obter os mesmos termos candidatos, pois d_1 , d_2 e d_3 também possuem valores de tipicidade maiores ou igual ao limiar δ , portanto os valores de $f1$ para o grupo g_2 são: g_1 : $f1(t_1, g_1) = 0,8$, $f1(t_2, g_1) = 0,8$, $f1(t_3, g_2) = 0,5$. Foi obtido então a mesma pontuação para os mesmos termos candidatos nos dois grupos. Com isso, fica claro as causas que levaram a extração dos descritores confusos apresentados na Tabela 4.5.

E ainda ressalta-se que esta situação se agrava ainda mais, com o aumento do número de grupos produzidos pelo agrupamento. Pois quanto maior for a quantidade de grupos, menor será o valor do limiar δ . Esta noção fica explícita na terceira propriedade do limiar apresentada na equação (4.7). Ou seja, mais facilmente os graus de compatibilidade possibilísticos irão passar no filtro do limiar. E consequentemente todos os documentos serão considerados relevantes para todos os grupos.

$$\lim_{c \rightarrow \infty} \delta = 0 \quad (4.7)$$

Na próxima sessão será descrito uma possível proposta para interpretação dos graus de compatibilidade das partições possibilística.

4.4.2 Interpretando os graus de compatibilidade das partições possibilísticas

A interpretação direta dos graus de compatibilidade possibilísticos gera um série de problemas na extração de descritores, conforme ficou demonstrado na sessão anterior. Com isso, podemos formular a seguinte pergunta: Como interpretar corretamente os graus de compatibilidade possibilísticos, para corretamente identificar os documentos relevantes de um dado grupo? Conforme sabemos, o valor de tipicidade pode variar livremente entre o intervalo $[0, 1]$, sem a restrição probabilística (equação 2.9) do FCM. Essa é uma característica positiva introduzida em Krishnapuram e Keller (1993), a qual atribui valores de pertinência mais justos aos grupos fuzzy, em consonância com a teoria de conjuntos fuzzy proposta em Zadeh (1965) e brevemente contextualizada no capítulo 2. Para melhor compreendermos esse conceito de valores mais justos, podemos analisar o que os autores defenderam na publicação original:

Since our membership functions correspond more closely to the notion of typicality, the resulting algorithms are naturally more immune to noise. Noise points will have low degrees of compatibility in all clusters, which makes their effect on the clustering negligible (KRISHNAPURAM; KELLER, 1993).

Portanto, a vantagem em deixar os graus de compatibilidade independentes, é a de se expressar a relevância/importância real de um elemento em relação a um grupo, o que consequentemente torna o método mais robusto e menos suscetível a ruídos. Para então identificar a importância de um documento em um grupo, poderíamos por exemplo escolher em qual grupo um documento d_i deveria ser considerado relevante, considerando o grupo em que esse documento d_i possuísse o maior grau de compatibilidade. No entanto, essa estratégia resultaria em uma extração de descritores *crisp*, perdendo assim toda a flexibilidade proporcionada nas partições *soft* (NOGUEIRA, 2013).

Sendo assim, precisamos de uma estratégia que consiga interpretar bem os graus de compatibilidade, de modo a se conservar a flexibilidade inerente a partição fuzzy, sem sacrificar a robustez contra ruídos da tipicidade.

Nesse contexto, propõe-se realizar essa interpretação em duas etapas. A primeira será constituída de uma conversão da tipicidade oriunda do PCM para a pertinência do FCM, de maneira a se satisfazer a restrição probabilística do FCM (equação 2.9). No entanto, ao apenas realizar a conversão iremos perder a robustez contra ruídos do PCM. Por isso, vamos adicionar uma penalidade ao cálculo da pontuação dos termos.

A conversão proposta dos valores de tipicidade para pertinência, dar-se a como apresentado na equação (4.8), a qual satisfaz a condição necessária (equação 4.9) para que o limiar δ seja aplicado, sem sofrer do problema de se considerar um documento como relevante em todos os grupos.

$$\lambda'(d_i, g_j) = \frac{\lambda(d_i, g_j)}{\sum_{k=1}^c \lambda(d_i, g_k)} \quad (4.8)$$

$$\sum_{k=1}^c \lambda'(d_i, g_k) = 1 \quad (4.9)$$

Entretanto, só realizar a conversão não atende aos dois requisitos exigidos. Para isso vamos adaptar a matriz de contingência apresentada na Tabela 4.6, adicionando uma ponderação as medidas, de modo a adicionar uma gratificação nos termos de documentos com elevada tipicidade e penalizar os termos de documentos com baixo valor de tipicidade, uma abordagem similar é encontrada no método SoftO-wFDCL (*Soft Organization - weighted Fuzzy Description Comes Last*) em [Nogueira \(2013\)](#).

	Documentos do grupo g_j com grau de compatibilidade maior ou igual a δ	Documentos do grupo g_j com grau de compatibilidade menor do que δ
Documentos que possuem o descritor candidato t_k	$ganhos(t_i, g_j)$	$ruídos(t_i, g_j)$
Documentos que não possuem o descritor candidato t_k	$perdas(t_i, g_j)$	$rejeitos(t_i, g_j)$

Tabela 4.10: Matriz de contingência do termo t_k para o grupo g_j adaptada para a partição possibilística

A adaptação sugerida da matriz contingência está apresentada na Tabela 4.10, na qual os valores de ganhos, perdas, ruídos e rejeitos foram mapeados para as equações (4.10), (4.11), (4.12) e (4.13) respectivamente. Portanto ao invés de realizar uma contagem discreta dos ganhos, perdas, ruídos e rejeitos, é realizado uma soma contínua das contribuições de cada documento ao grupo, considerando o valor de tipicidade. Desse modo, conseguimos reduzir a contribuição de documentos com baixa tipicidade no grupo e aumentar a contribuição de documentos com alta tipicidade no grupo. Resultando assim, em uma pontuação mais justa e mais coerente dos termos extraídos dos grupos. Os ajustes necessários nas medidas de precisão e recuperação estão detalhados nas equações (4.14) e (4.15), enquanto o cálculo da medida $f1(t_k, g_j)$ permanece como definido na equação (4.4).

$$ganhos(t_i, g_j) = \sum_{d \in D'} \lambda(d, g_j), D' = \{d | d \in D, \lambda'(d, g_j) \geq \delta, \varphi(t_i, d) > 0\} \quad (4.10)$$

$$perdas(t_i, g_j) = \sum_{d \in D'} \lambda(d, g_j), D' = \{d | d \in D, \lambda'(d, g_j) \geq \delta, \varphi(t_i, d) = 0\} \quad (4.11)$$

$$ruídos(t_i, g_j) = \sum_{d \in D'} \lambda(d, g_j), D' = \{d | d \in D, \lambda'(d, g_j) < \delta, \varphi(t_i, d) > 0\} \quad (4.12)$$

$$rejeitos(t_i, g_j) = \sum_{d \in D'} \lambda(d, g_j), D' = \{d | d \in D, \lambda'(d, g_j) < \delta, \varphi(t_i, d) = 0\} \quad (4.13)$$

$$precisão(t_k, g_j) = \frac{ganhos(t_k, g_j)}{ganhos(t_k, g_j) + ruídos(t_k, g_j)} \quad (4.14)$$

$$\text{recuperação}(t_k, g_j) = \frac{\text{ganhos}(t_k, g_j)}{\text{ganhos}(t_k, g_j) + \text{perdas}(t_k, g_j)} \quad (4.15)$$

Nas próximas duas sessões é caracterizado as duas abordagens que derivam dessa interpretação aqui detalhada. A primeira, é uma proposta de extensão do método SoftO-FDCL voltado para a partição possibilística do PCM, com a interpretação apresentada nessa sessão. Enquanto a segunda abordagem, apresenta uma estratégia de extração de descritores para o algoritmo PFCM, interpretando em conjunto as duas partições presentes no algoritmo.

4.4.3 O método PDCL

Agora que a proposta de interpretação da partição possibilística está concluída, vamos reunir todos esses passos em um método para extração de descritores para a partição possibilística, a qual será denominado aqui de PDCL (*Possibilistic Descriptor Comes Last*). Para realizar a extração de descritores o método PDCL, considera inicialmente todos os termos como candidatos, em seguida para cada grupo g_j é calculado a precisão (equação 4.14) e recuperação (equação 4.15) de todos os termos t_k . A partir então destes valores é calculada a pontuação de cada termo t_k no grupo g_j , com a medida $f1(t_k, g_j)$ (equação 4.4). Obtido então essa pontuação por grupo dos termos candidatos, deve-se selecionar os m descritores de maior pontuação em cada grupo. No qual a quantidade m de descritores é definida pelo usuário. A síntese do método PDCL está apresentado no Algoritmo 6.

4.4.4 O método Mixed-PFDCL

Na discussão do experimento da sessão 4.3, onde é apresentado os resultados do refinamento com o método PFCM (*Possibilistic Fuzzy C Means*) na proposta de organização flexível de documentos, foi salientado a importância de se montar uma estratégia que conseguisse capturar a filosofia híbrida do algoritmo PFCM, levando em consideração suas duas partições.

Uma das características presentes no método PFCM, é a adição dos parâmetros a e b que atuam como reguladores da influência do FCM e do PCM no agrupamento obtido. Portanto há de destacar, a importância de se considerar tais parâmetros no processo de extração de descritores, objetivando assim mais coerência com o algoritmo.

Nesse contexto, é apresentado a equação (4.16), a qual combina os valores de pertinência e de tipicidade convertidos, em um único grau de compatibilidade. Essa combinação nada mais é do que a média ponderada pelos parâmetros a e b dos valores de pertinência e tipicidade convertido respectivamente.

$$\mu'(d_i, g_j) = \frac{a\mu(d_i, g_j) + b\lambda'(d_i, g_j)}{a + b} \quad (4.16)$$

Como resultado dessa combinação precisamos apenas adaptar as medidas de ganhos, perdas, ruídos e rejeitos, para considerarem a relevância de um dado documento d em relação ao limiar δ , a partir da pertinência híbrida $\mu'(d_i, g_j)$. Essa adaptação está então

disposta nas equações (4.17), (4.18), (4.19) e (4.20). Ressalta-se que foi mantido no somatório de ganhos, perdas, ruídos e rejeitos, o valor de tipicidade, devido a capacidade do mesmo em expressar melhor a realidade do agrupamento. As demais medidas de precisão, recuperação e $f1$, permanecem com a mesma definição apresentada anteriormente. Para uma melhor compreensão de como se encaixa todas essas partes, um pseudo código do método Mixed-PFDCL está apresentado no Algoritmo 6.

$$ganhos(t_i, g_j) = \sum_{d \in D'} \lambda(d, g_j), D' = \{d | d \in D, \mu'(d, g_j) \geq \delta, \varphi(t_i, d) > 0\} \quad (4.17)$$

$$perdas(t_i, g_j) = \sum_{d \in D'} \lambda(d, g_j), D' = \{d | d \in D, \mu'(d, g_j) \geq \delta, \varphi(t_i, d) = 0\} \quad (4.18)$$

$$ruídos(t_i, g_j) = \sum_{d \in D'} \lambda(d, g_j), D' = \{d | d \in D, \mu'(d, g_j) < \delta, \varphi(t_i, d) > 0\} \quad (4.19)$$

$$rejeitos(t_i, g_j) = \sum_{d \in D'} \lambda(d, g_j), D' = \{d | d \in D, \mu'(d, g_j) < \delta, \varphi(t_i, d) = 0\} \quad (4.20)$$

extrair-descritores(U, P, D, G, m)

início

descritores $\leftarrow \emptyset$;

para cada $g \in G$ **faça**

 candidatos $\leftarrow [t_1, t_2, \dots, t_k]$;

 ranking $\leftarrow \emptyset$;

para cada $t \in candidatos$ **faça**

 precisao \leftarrow equação (4.2) para o PDCL ou (4.14) para o Mixed-PFDCL;

 recuperacao \leftarrow equação (4.3) para o PDCL ou (4.15) para o Mixed-PFDCL;

 pontuacao \leftarrow equação (4.4);

 ranking[t] \leftarrow pontuacao;

fim

 descritores[g] $\leftarrow m$ termos do ranking com maior pontuacao;

fim

retorne (descritores);

fim

Algoritmo 6: Pseudo código da extração de descritores com os métodos PDCL e Mixed-PFDCL. Onde considere U a partição de pertinências fuzzy (equação 2.7), P a partição possibilística (equação 2.16), D a coleção de documentos da coleção, G os grupos produzidos pelo método de agrupamento e m a quantidade descritores desejada por grupo.

4.4.5 Resultados

Para mensurar os impactos das duas estratégias híbridas para extração de descritores apresentadas aqui nesta seção, foi realizado outro experimento, com os algoritmos PCM

e PFCM, com as bases apresentadas na seção 4.2. Durante o experimento fora utilizado os métodos de extração de descritores SoftO-FDCL, PDCL, Mixed-PFDCL.

Desde que as propostas aqui apresentadas, pretendem otimizar os resultados apresentados anteriormente, foi adotada uma metodologia similar ao experimento anterior. Ou seja, o agrupamento final obtido para cada base, é resultado dos grupos que obtiveram o maior valor na medida de silhueta fuzzy. Onde a quantidade de grupos para cada base, variou entre 2 e o número de classes de cada base (Tabela 4.2). Ressalta-se ainda que para minimizar os efeitos da aleatoriedade da partição inicial nesse experimento, o agrupamento foi executado 5 vezes para cada quantidade de grupos na silhueta fuzzy.

Os parâmetros m e n que regulam a variação entre fuzzy e *crisp* das partições resultantes conforme ilustrados nas Figuras 2.1 e 2.2, foram definidos para 1.2 de maneira empírica. De maneira geral, observou-se que as bases de maior dimensionalidade, estavam produzindo grupos coincidentes mais facilmente, quando os valores de m e n eram maior do que 1.5. Por sua vez os parâmetros a e b do algoritmo PFCM, o qual define a influência das pertinências fuzzy do FCM e os graus de compatibilidade possibilísticos do PCM respectivamente, foram definidos como sendo $a = 1,0$ e $b = 1,2$. Essa escolha deriva das conclusões apresentadas em Pal et al. (2005), o qual nos informa que de acordo com a função objetivo (equação 2.18) do PFCM, se utilizarmos valores de a maiores que b , os protótipos dos grupos são mais influenciados pelas pertinências. Contudo, como em coleções textuais naturalmente esparsas, é mais comum haver documentos ruidosos, o qual não se encaixe totalmente em nenhum grupo. Sendo assim, foi adotado o valor de b maior do que a , para reduzir os efeitos indesejados dos ruídos.

As coleções foram então agrupadas com os métodos PCM e PFCM, utilizando a metodologia descrita. A quantidade ótima de grupos, obtida com o método da silhueta fuzzy, está apresentado na Tabela 4.11. Os resultados apresentado nesta tabela, reforçam as conclusões apresentadas experimento anterior, de que a quantidade de grupos ótima do método PFCM tende a se aproximar mais da quantidade original de classes de cada coleção, enquanto o método PCM possui uma tendência em obter um número de grupos bem inferior a quantidade original de classes.

Coleção	# classes	PCM	PFCM
Opinosis	3	2	3
20Newsgroup	4	4	4
Hitech	6	2	6
NSF	16	2	8
WAP	20	2	17
Reuters-21578	43	4	40

Tabela 4.11: Quantidade ótima de grupos determinada através do método da silhueta fuzzy para cada algoritmo de agrupamento no segundo experimento conduzido com os métodos PCM e PFCM

Em seguida, foi realizado a extração dos descritores sobre as partições ótimas encontradas por cada algoritmo de agrupamento, sobre as coleções textuais. Como a motivação

desse experimento, foi avaliar a qualidade dos descritores produzidos pelos métodos PDCL e Mixed-PDCL propostos, a extração de descritores foi também realizada com o método SoftO-FDCL, possibilitando assim compararmos os resultados. E assim como no experimento anterior, essa análise quantitativa dos descritores produzidos, foi feita, utilizando a mesma estratégia de avaliação preditiva, com os 5 algoritmos de classificação do experimento anterior. No entanto, como a proposta de interpretação das duas partições produzidas pelo algoritmo PFCM, contempla um grau de compatibilidade misto entre as duas partições, foi pertinente generalizar essa interpretação também para função de defuzzificação dos grupos, apresentada na equação (4.1). Essa adaptação está apresentada na equação (4.21), onde foi adicionado, o grau de compatibilidade híbrido proposto na equação (4.16).

$$crisp(d_i) = \begin{cases} crisp_j, & \mu(d_i, g_j) = \max_{\forall g \in G} \mu(d_i, g), \text{ se a partição for fuzzy} \\ crisp_j, & \lambda(d_i, g_j) = \max_{\forall g \in G} \lambda(d_i, g), \text{ se a partição for possibilística} \\ crisp_j, & \mu'(d_i, g_j) = \max_{\forall g \in G} \mu'(d_i, g), \text{ se houver duas partições} \end{cases} \quad (4.21)$$

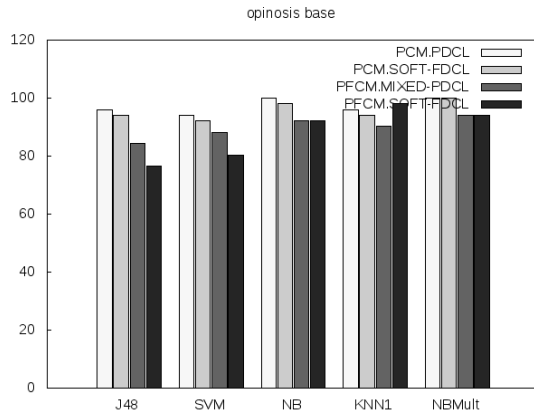


Figura 4.8: Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção Opinois

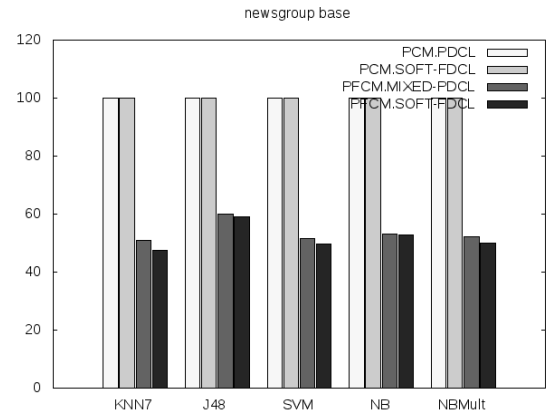


Figura 4.9: Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção 20Newsgroup

4.5 CONSIDERAÇÕES FINAIS

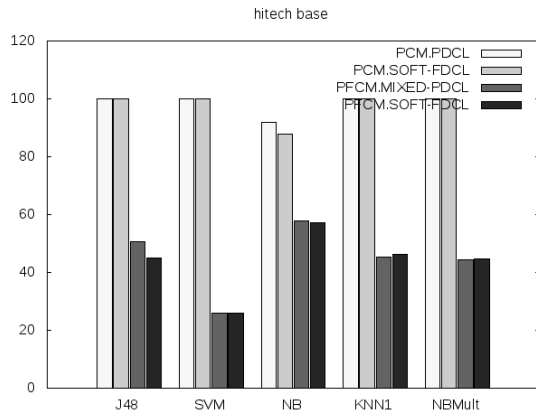


Figura 4.10: Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção Hitech

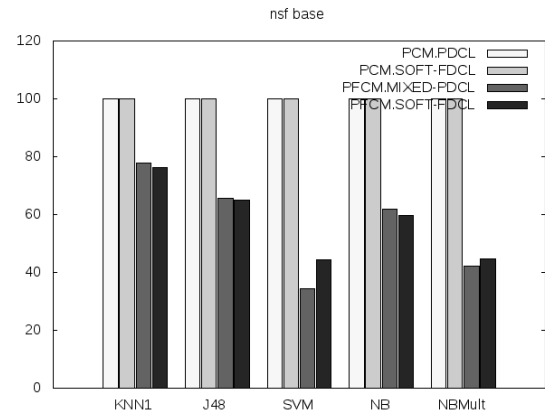


Figura 4.11: Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção NSF

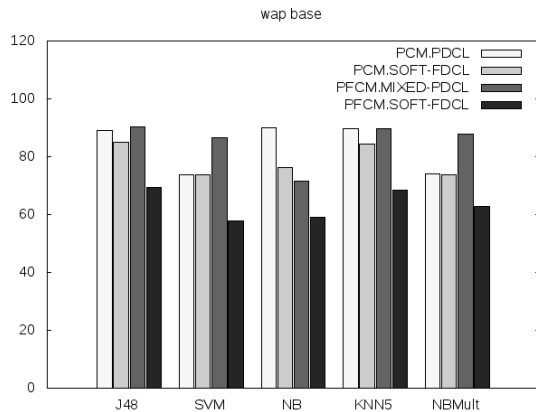


Figura 4.12: Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção WAP

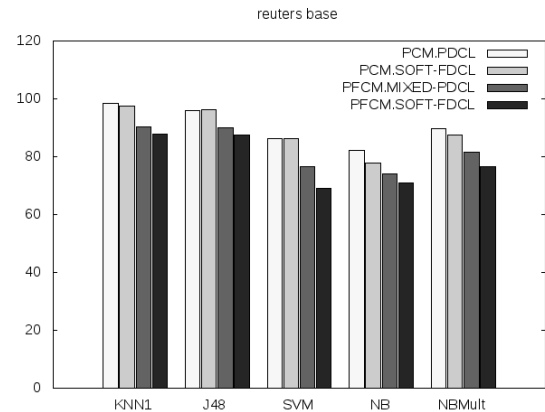
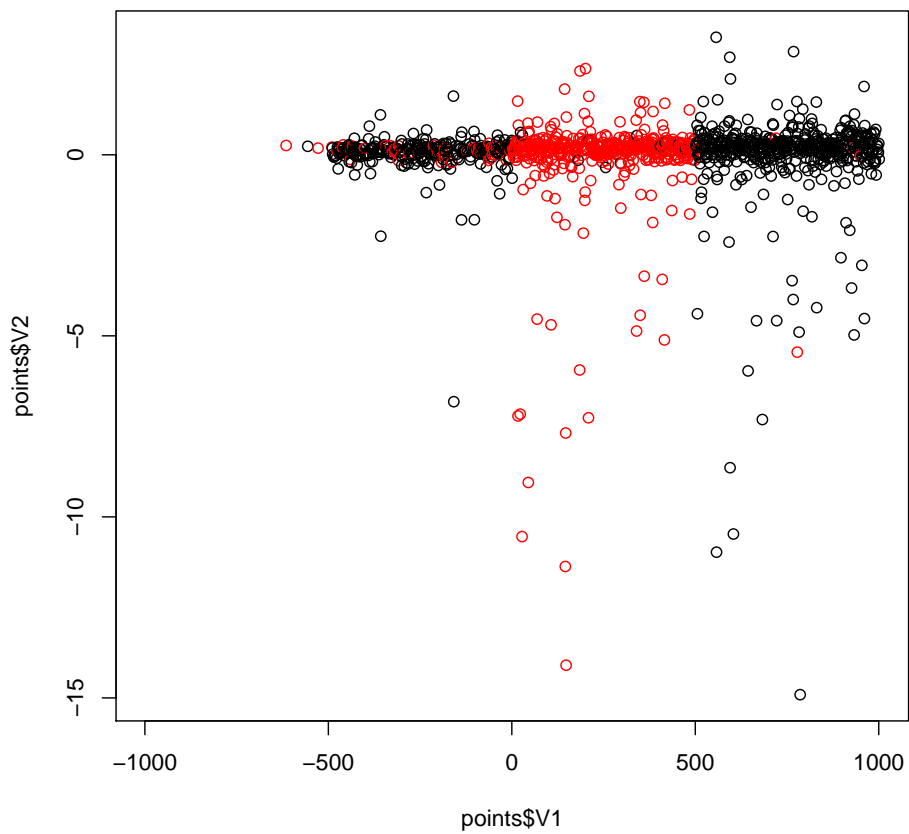


Figura 4.13: Desempenho obtido dos descritores extraídos com os algoritmos SoftO-FDCL, Mixed-PDCL e PDCL sobre o agrupamento produzido pelos métodos PCM e PFCM na coleção Reuters-21578

	PCM		PFCM	
nome	SoftO-FDCL	PDCL	SoftO-FDCL	Mixed-PDCL
Opinosis		✓		✓
20newsgroups	✓	✓		✓
Hitech		✓		✓
NSF	✓	✓		✓
WAP		✓		✓
Reuters-21578		✓		✓

Tabela 4.12: Sumário dos resultados da classificação dos descritores

Figura 4.14: Ilustração denotando os grupos g_1, g_2, g_3 organizados sem sobreposição, para $m = 1$.

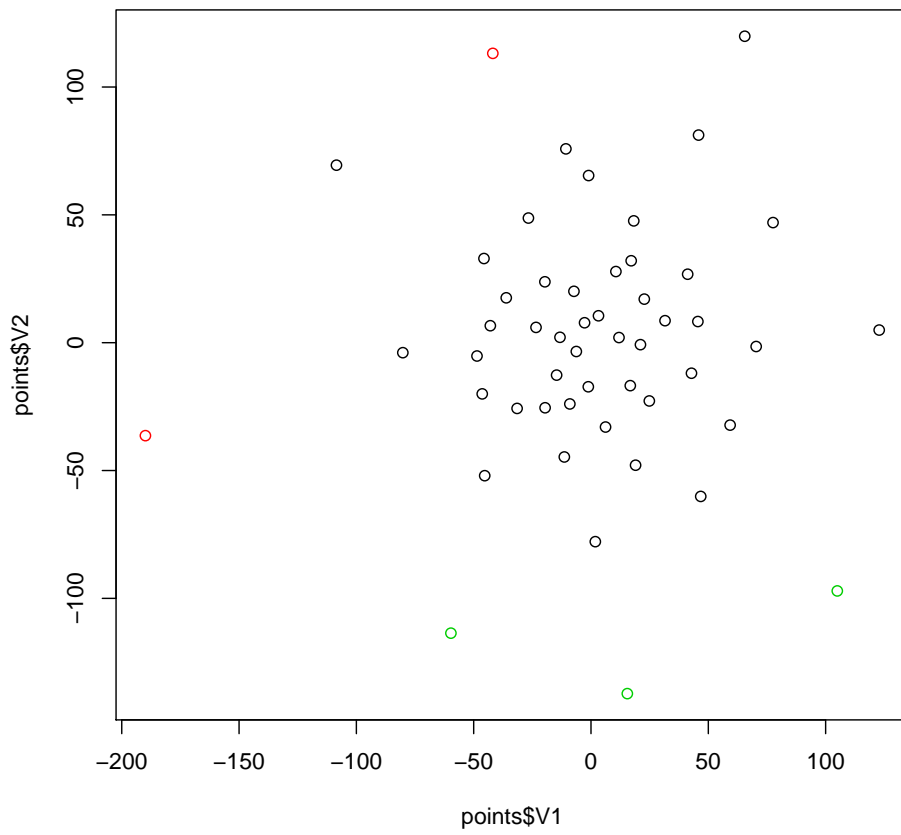


Figura 4.15: Ilustração denotando os grupos g_1, g_2, g_3 organizados de maneira fuzzy, com sobreposição, quando $m \rightarrow \infty$.

Capítulo

5

Síntese da investigação e dos experimentos realizados nesta monografia

CONCLUSÃO

Capítulo

6

Possíveis explorações adicionais que derivam deste trabalho

TRABALHOS FUTUROS

REFERÊNCIAS BIBLIOGRÁFICAS

- BEZDEK, J. C.; EHRLICH, R.; FULL, W. Fcm: The fuzzy c-means clustering algorithm. *Computers & Geosciences*, v. 10, n. 2, p. 191 – 203, 1984. ISSN 0098-3004. Disponível em: <http://www.sciencedirect.com/science/article/pii/0098300484900207>.
- CARVALHO, N. V. J. et al. Flexible Document Organization by Mixing Fuzzy and Possibilistic Clustering algorithms. *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, p. 1–8, 2016.
- CHEN, G. *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*. Hoboken, NJ: CRC Press, 2000. Disponível em: <https://cds.cern.ch/record/1250131>.
- CUNHA, A. G.; TAKAHASHI, R.; ANTUNES, C. H. *Manual de computação evolutiva e metaheurística*. Coimbra University Press, 2012. Disponível em: <http://dx.doi.org/10.14195/978-989-26-0583-8>.
- DENG, J. et al. An improved fuzzy clustering method for text mining. In: *The 2nd International Conference on Networks Security, Wireless Communications and Trusted Computing (NSWCTC), 2010*. [S.l.: s.n.], 2010. v. 1, p. 65–69.
- FELDMAN, R.; SANGER, J. *The text mining handbook: Advanced approaches in analyzing unstructured data*. [S.l.]: Cambridge University Press, 2007.
- FRANK, A.; ASUNCION, A. *UCI Machine Learning Repository*. 2010. Disponível em: <http://archive.ics.uci.edu/ml>.
- GROVER, N. A study of various fuzzy clustering algorithms. *International Journal of Engineering Research*, v. 3, p. 177–181, 2014.
- HADDI, E.; LIU, X.; SHI, Y. The role of text pre-processing in sentiment analysis. *Procedia Computer Science*, v. 17, p. 26 – 32, 2013. ISSN 1877-0509. First International Conference on Information Technology and Quantitative Management. Disponível em: <http://www.sciencedirect.com/science/article/pii/S1877050913001385>.
- HALL, M. et al. The WEKA data mining software: an update. *SIGKDD Explorations*, ACM, New York, NY, USA, v. 11, n. 1, p. 10–18, nov. 2009.
- HAVENS, T. et al. Fuzzy c-means algorithms for very large data. *IEEE Transactions on Fuzzy Systems*, v. 20, n. 6, p. 1130–1146, 2012.

HAYES, P. J.; WEINSTEIN, S. P. Construe/TIS: A system for content-based indexing of a database of news stories. In: *2nd Annual Conference on Innovative Applications of Artificial Intelligence*. [S.l.: s.n.], 1990. p. 1–5.

JIANG, H. et al. An improved method of fuzzy clustering algorithm and its application in text clustering. *JOURNAL OF INFORMATION & COMPUTATIONAL SCIENCE*, *JOURNAL OF INFORMATION & COMPUTATIONAL SCIENCE*, v. 10, n. 2, p. 519, 2013. Disponível em: http://manu35.magtech.com.cn/Jwk_ics/EN/abstract/article.1507.shtml.

KARAMI, A. et al. FLATM: A fuzzy logic approach topic model for medical documents. In: *2015 Annual Conference of the North American Fuzzy Information Processing Society (NAFIPS) held jointly with 2015 5th World Conference on Soft Computing (WConSC)*. Institute of Electrical & Electronics Engineers (IEEE), 2015. Disponível em: <http://dx.doi.org/10.1109/NAFIPS-WConSC.2015.7284190>.

KARYPIS, G. *Cluto - software for clustering high-dimensional datasets*. 2015. <http://glaros.dtc.umn.edu/gkhome/cluto/cluto/download.>

KRISHNAPURAM, R.; KELLER, J. M. A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems*, v. 1, n. 2, p. 98–110, 1993. ISSN 1063-6706.

KUMAR, D. et al. A hybrid approach to clustering in big data. *IEEE Transactions on Cybernetics*, Institute of Electrical & Electronics Engineers (IEEE), p. 1–1, 2015. Disponível em: <http://dx.doi.org/10.1109/TCYB.2015.2477416>.

LANG, K. NewsWeeder: Learning to filter netnews. In: *Machine Learning Proceedings 1995*. Elsevier BV, 1995. p. 331–339. Disponível em: <http://dx.doi.org/10.1016/B978-1-55860-377-6.50048-7>.

MARCACINI, R. M.; REZENDE, S. O. Incremental construction of topic hierarchies using hierarchical term clustering. In: *Proceedings of the 22nd International Conference on Software Engineering & Knowledge Engineering (SEKE'2010), Redwood City, San Francisco Bay, CA, USA, July 1 - July 3, 2010*. [S.l.]: Knowledge Systems Institute Graduate School, 2010. p. 553. ISBN 1-891706-26-8.

MATSUMOTO, T.; HUNG, E. Fuzzy clustering and relevance ranking of web search results with differentiating cluster label generation. In: *FUZZ-IEEE*. IEEE, 2010. p. 1–8. ISBN 978-1-4244-6919-2. Disponível em: <http://dblp.uni-trier.de/db/conf/fuzzIEEE/fuzzIEEE2010.html#MatsumotoH10>.

MOORE, J. et al. Web page categorization and feature selection using association rule and principal component clustering. *IBM shared research report/University of Minnesota (Minneapolis, Mn.)*, Citeseer, v. 98, p. 3, 1997.

MURALI, D.; DAMODARAM, A. Semantic document retrieval system using fuzzy clustering and reformulated query. In: *2015 International Conference on Advances in Computer Engineering and Applications*. Institute of Electrical & Electronics Engineers (IEEE), 2015. Disponível em: <http://dx.doi.org/10.1109/ICACEA.2015.7164788>.

NOGUEIRA, T. M. *Organização Flexível de Documentos*. Tese (Doutorado) — ICMC-USP, 2013.

NOGUEIRA, T. M.; CAMARGO, H. A.; REZENDE, S. O. Flexible document organization: Comparing fuzzy and possibilistic approaches. In: *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. [S.l.: s.n.], 2015. p. 1–8.

NOGUEIRA, T. M.; REZENDE, S. O.; CAMARGO, H. A. Fuzzy cluster descriptors improve flexible organization of documents. In: *2012 12th International Conference on Intelligent Systems Design and Applications (ISDA)*. Institute of Electrical & Electronics Engineers (IEEE), 2012. Disponível em: <http://dx.doi.org/10.1109/ISDA.2012.6416608>.

PAL, N. R. et al. A possibilistic fuzzy c-means clustering algorithm. *IEEE Transactions on Fuzzy Systems*, IEEE Press, v. 13, n. 4, p. 517–530, 2005. ISSN 1063-6706.

PEDRYCZ, A.; REFORMAT, M. Hierarchical FCM in a stepwise discovery of structure in data. *Soft Comput.*, v. 10, n. 3, p. 244–256, 2006. Disponível em: <http://dx.doi.org/10.1007/s00500-005-0478-8>.

POPESCU, M. et al. Random projections fuzzy c-means (rpfc) for big data clustering. In: *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. [S.l.: s.n.], 2015. p. 1–6.

RIOS, A. R.; MELLO, F. R. A systematic literature review on decomposition approaches to estimate time series components. *Journal of Computer Science*, 2010.

ROSSI, R. G.; MARCACINI, R. M.; REZENDE, S. O. *ICMC TECHNICAL REPORT - Benchmarking Text Collections for Classification and Clustering Tasks*. São Carlos, SP, Brasil, 2013. Disponível em: http://www.icmc.usp.br/CMS/Arquivos/arquivos_enviados/BIBLIOTECA_113_RT_395.pdf.

SARANYA, J.; ARUNPRIYA, C. Survey on clustering algorithms for sentence level text. *International Journal of Computer Trends and Technology*, Seventh Sense Research Group Journals, v. 10, n. 2, p. 61–66, apr 2014. Disponível em: <http://dx.doi.org/10.14445/22312803/IJCTT-V10P111>.

STEINBACH, M.; ERTÖZ, L.; KUMAR, V. The challenges of clustering high-dimensional data. In: *In New Vistas in Statistical Physics: Applications in Econophysics, Bioinformatics, and Pattern Recognition*. [S.l.]: Springer-Verlag, 2003. ISBN 978-3-642-07739-5.

SUBHASHINI, R.; KUMAR, V. Evaluating the performance of similarity measures used in document clustering and information retrieval. In: *First International Conference on Integrated Intelligent Computing (ICIIC)*. [S.l.: s.n.], 2010. p. 27–31.

TJHI, W.-C.; CHEN, L. A heuristic-based fuzzy co-clustering algorithm for categorization of high-dimensional data. *Fuzzy Sets and Systems*, Elsevier BV, v. 159, n. 4, p. 371–389, feb 2008. Disponível em: <http://dx.doi.org/10.1016/j.fss.2007.10.003>.

TREERATPITUK, P.; CALLAN, J. Automatically labeling hierarchical clusters. In: FORTES, J. A. B.; MACINTOSH, A. (Ed.). *DG.O. Digital Government Research Center*, 2006. (ACM International Conference Proceeding Series, v. 151), p. 167–176. Disponível em: <http://dblp.uni-trier.de/db/conf/dgo/dgo2006.html#TreeratpitukC06>.

YAN, Y.; CHEN, L. Hyperspherical possibilistic fuzzy c-means for high-dimensional data clustering. In: *Proceedings of the 7th International Conference on Information, Communications and Signal Processing*. [S.l.: s.n.], 2009. (ICICS'09), p. 637–641. ISBN 978-1-4244-4656-8.

YAN, Y.; CHEN, L.; TJHI, W.-C. Fuzzy semi-supervised co-clustering for text documents. *Fuzzy Sets and Systems*, Elsevier BV, v. 215, p. 74–89, mar 2013. Disponível em: <http://dx.doi.org/10.1016/j.fss.2012.10.016>.

ZADEH, L. Fuzzy sets. *Information and Control*, v. 8, n. 3, p. 338 – 353, 1965. ISSN 0019-9958. Disponível em: <http://www.sciencedirect.com/science/article/pii/S001999586590241X>.

ZHANG, C.; XU, H. Clustering description extraction based on statistical machine learning. *Intelligent Information Technology Applications, 2007 Workshop on*, IEEE Computer Society, Los Alamitos, CA, USA, v. 2, p. 22–26, 2008.