GRÂNULOS DE PALAVRAS PARA REPRESENTAÇÃO DE TEXTO

Patrícia Fiuza de Castro

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia de Sistemas e Computação, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia de Sistemas e Computação.

Orientador: Geraldo Bonorino Xexéo

Rio de Janeiro

Abril de 2013

GRÂNULOS DE PALAVRAS PARA REPRESENTAÇÃO DE TEXTO

Patrícia Fiuza de Castro

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA DE SISTEMAS E COMPUTAÇÃO.

Examinada por:

Prof. Geraldo Bonorino Xexéo, D.Sc.

Prof. Marco Antônio Casanova, Ph.D.

Profa. Adriana Santarosa Vivacqua, D.Sc.

Prof. Geraldo Zimbrão da Silva, D.Sc.

Prof Jano Moreira de Souza, Ph.D.

RIO DE JANEIRO, RJ - BRASIL

ABRIL DE 2013

Castro, Patrícia Fiuza de

Grânulos de Palavras para Representação de Texto/Patrícia Fiuza de Castro. – Rio de Janeiro: UFRJ/COPPE, 2013.

XIV, 123 p.: il.; 29,7 cm.

Orientador: Geraldo Bonorino Xexéo

Tese (doutorado) – UFRJ/ COPPE/ Programa de Engenharia de Sistemas e Computação, 2013.

Referências Bibliográficas: p. 114-123.

1. Modelo de Documento. 2. Computação Granular. 3. Agrupamento Espectral. I. Xexéo, Geraldo Bonorino. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Sistemas e Computaçãol. III. Título.

A memória de meu avô, Hilton, por tudo o que ele representa neste caminho até aqui.

Agradecimentos

Ao professor Geraldo Xexéo, pela orientação e pelo conhecimento transmitido.

A UFRJ, especialmente ao PESC/COPPE, por ter me acolhido e pela qualidade do curso ministrado.

A FAPERJ pela bolsa concedida.

A todos os meus professores, pelos conhecimentos transmitidos.

Aos professores membros da banca, por aceitarem participar da avaliação deste trabalho.

A todos os meus amigos, pelo carinho e incentivo durante esse período.

Aos meus cães, pela parceria incondicional.

A todos que, direta ou indiretamente, me ajudaram a chegar até aqui.

But what is meaning?

... For a start, I take it that meaning as carried by words and words string is what allows modern humans to engage in verbal thought and rich interpersonal communication. But this, of course, still begs the question of what meaning itself is.

Philosophers, linguists, humanists, novelists, poets and theologians have used the word "meaning" in a plethora of ways, ranging, for example, from the truth of matters to intrinsic properties of objects and happenings of the world, to mental constructions of the outside world, to physically irreducible mystical essences, as in Plato´s ideas, to symbols in a internal communicatio nand reasoning system, to potentially true but too vague notions such as how words are used. Some assert that meaning are abstract concepts or properties of the world that exist prior to and independently of any language-dependent representation. This leads to assertion that by nature or definition computers cannot create meaning from data; meaning must exist first. Therefore, what a computer creates, store, and uses cannot, ipso facto, be meaning itself.In our view, however, what goes on in the mind (and, by identity, the brain) in direct visual or auditory, or any other perception, is fundamentally the same as what goes on in any others form of cognition and has no necessary priority over others resources of knowledge, such as - in particular - autonomous manipulations of strings of words that convey abstract combinations of ideas such as imaginary numbers. Of course, strings of words must somehow be able to represent and convey both veridical and hypothetical information about our inner and outer worlds; otherwise, language would not be very useful. Certainly, that is, much perceptual experience must map onto linguistic expression And many linguistics expressions must map onto perceptual experience. However, once the the mappings have been obtained through the cultural evolution of a language, there is no necessity that most of the knowledge of meaning cannot learned from exposure to language itself ...

.... This puts the causal situation in a different light. We may often first learn relations of most words and passages to each other from our matrices of verbal experiences and attach them to perceptual experience by embedding them in the abstract word space."

(Thomas Landauer – LSA as a Theory of Meaning)

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

GRÂNULOS DE PALAVRAS PARA REPRESENTAÇÃO DE TEXTO

Patrícia Fiuza de Castro

Abril/2013

Orientador: Geraldo Bonorino Xexéo

Programa: Engenharia de Sistemas e Computação

A quantidade de dados disponíveis em formato semi-estruturados ou não cresce exponencialmente. A área de mineração de texto visa a descoberta de conhecimento a partir de dados deste tipo. A maioria dos trabalhos nessa área utiliza o modelo conhecido como saco de palavras para representar os textos. Esta forma de representação, apesar de eficaz, minimiza a qualidade do conhecimento descoberto uma vez que não é capaz de capturar as características essenciais deste tipo de dados, tais como a semântica e contexto. O paradigma de computação granular tem sido demonstrado eficaz no tratamento de problemas complexos de processamento de informação e pode produzir resultados significativos em ambientes de larga escala, tais como a Internet. Este trabalho explora o processo de granulação de palavras com vista à sua aplicação na melhoria subseqüente em representação de texto.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

GRANULES OF WORDS TO TEXT REPRESENTATION

Patrícia Fiuza de Castro

April/2013

Advisor: Geraldo Bonorino Xexéo

Department: Computer Science and Engineering

The amount of data available in semi-structured or unstructured format grows exponentially. The area of text mining aims at discovering knowledge from data of this type. Most work in this area uses the model known as bag of words to represent the texts. This form of representation, although effective, minimizes the quality of knowledge discovered because it is not able to capture essential characteristics of this type of data such as semantics and context. The paradigm of granular computing has been shown effective in the treatment of complex problems of information processing and can produce significant results in large-scale environments such as the Web. This work explores the granulation process of words with a view to its application in the subsequent improvement in text representation.

Sumário

Índice de Figuras

Índice de Tabelas

1. INTRODUÇÃO

1.1 Motivação

Na maioria dos sistemas de recuperação de informação, os usuários fornecem uma descrição da informação que necessitam e o sistema recupera os documentos armazenados que coincidem com estas descrições. Trata-se de uma abordagem baseada em consulta e é predominante entre os sistemas tradicionais (RIJBERGSEN, 1975) (SALTON, 1973). Neste contexto, vários modelos de recuperação foram propostos, tais como, o modelo booleano (SALTON, 1973), o modelo probabilístico (CROFT, 1992) e o modelo vetorial (SALTON, 1973). Nestes modelos, os documentos são representados como *bag of words* (sacos de palavras) ou termos que aparecem nos documentos.

Outras abordagens exploram a capacidade humana de reconhecimento da relevância dos documentos ao invés da sua descrição. Consistem na organização dos documentos em algum tipo de estrutura que permite aos seus usuários a exploração ou navegação através da coleção de documentos. Sistemas de hipertexto (NIELSEN, 1990), hierarquias de grupos ou clusters (CROUCH, 1989) e interfaces de recuperação baseadas em objetos (LUCARELLA , 1993) podem ser vistos como variações desta abordagem. Também nesta abordagem, quando a tarefa de organização é automatizada, os documentos são representados em um esquema de *Bag of Words* (BoW).

Geralmente, este BoW é obtido por análise estatística do documento, com base em algum esquema de pesagem dos seu termos. Este tipo de análise, apesar de efetiva, é incapaz de capturar muitas das características essenciais de um documento e de uma coleção. Tal característica pode, então, reduzir a eficiência semântica da recuperação da informação ou, por outro lado, da exploração da coleção de documentos.

Um emergente paradigma de tratamento de informação, denominado computação granular, tem chamado a atenção de muitos pesquisadores. Segundo (YAO, 2007), computação granular agrega um conjunto de teorias, metodologias, técnicas e ferramentas que fazem uso de grânulos para a solução de problemas complexos. Segundo (PEDRYCZ, 2005), os grânulos permeiam qualquer tarefa humana. Humanos estão, constantemente, abstraindo e formulando conceitos a partir destes grânulos, processando estes conceitos e devolvendo os resultados deste tratamento. Como exemplo, podemos citar a capacidade humana de lidar com imagens.

Em momento algum, seus pixels são considerados individualmente. A todo instante, agrupamentos destes pixels são formados, segundo alguma semântica capaz de transmitir noções de textura, cor, etc. Da mesma forma, ao analisar textos, as palavras não são consideradas individualmente. Agrupamentos destas palavras, representando alguma semântica, transmitem o seu conteúdo. Humanos são capazes de perceber o mundo real através de muitos níveis de granularidade (abstração) e podem, facilmente, alternar entre estes vários níveis. Consequentemente, podem abstrair e considerar apenas aquilo que serve para um interesse específico e ignorar aquilo que é irrelevante (YAO & ZHONG, 2002) (HOBBS, 1985) (YAO, 2007a). Por conseguir focar em diferentes níveis de granularidade, são capazes de obter diferentes níveis de conhecimento bem como de ter uma compreensão profunda da estrutura inerente a cada tipo de conhecimento. O raciocínio granular é, então, essencial para a inteligência humana e, segundo (ZHONG & YAO, 2008), pode causar um impacto significativo sobre as metodologias de solução de problemas, principalmente em ambientes de larga escala como a Web.

Existem muitas questões fundamentais em computação granular tais como a granulação do universo, a descrição dos grânulos, os relacionamentos entre os grânulos e a computação com os grânulos. Tais aspectos podem ser estudados a partir de dois aspectos relacionados, a construção dos grânulos e a computação com estes grânulos. O primeiro trata da formação, representação, e interpretação dos grânulos, enquanto o último trata da utilização destes grânulos na solução de problemas. Usando técnicas e princípios de computação granular, podemos estudar modelos para sistemas que suportem a recuperação de informação e a manipulação de dados no formato textual sob vários pontos de vista. Estes novos pontos de vista podem levar a construção de ferramentas mais eficientes na manipulação de grandes volumes de documentos.

1.2 Abordagens Relacionadas

Modelos para documentos, que sejam alternativos ao tradicional baseado em *bag of words* , têm despertado o interesse de pesquisadores de diversas áreas. Entre os trabalhos com este objetivo e que se baseiam no modelo vetorial essencialmente estão, por exemplo, o de (LIU, 1994), que apresenta o modelo vetorial semântico (Semantic

Vector Space Model – SVSM) e o de (BILLHART, 2002) ,que apresenta o modelo vetorial de contexto (Context Vector Model - CVM). Os dois métodos capturam a semântica dos documentos através da redução do vetor e pelo cálculo de co-ocorrências dos termos nos documentos. Outras abordagens também sido propostas.

(DOAN, 2005) propõe a modelagem de textos baseada na teoria de conjuntos fuzzy. Um novo algoritmo de escolha dos termos que caracterizam o documento é dada sob o ponto de vista fuzzy. Resultados experimentais, alcançados através de testes efetuados para a tarefa de categorização e apoiados por *feedback* de relevância técnica, mostram que o método proposto reduz o número de dimensões e atinge melhor desempenho se comparados a outros métodos. Além disso, produz resultados muito favoráveis quando comparados a outros métodos de seleção de termos.

(KHALLED, 2006), em sua tese de doutorado, apresenta um novo paradigma para mineração de documentos capaz de explorar características semânticas dos documentos. O esquema de representação, baseado em grafos, é construído através de sucessivas etapas de análise sintática e semântica. Uma medida de distância é apresentada para determinar similaridades entre os conteúdos dos documentos.

(WANG, 2008) propõe um modelo de representação que utiliza uma rede léxica para representar texto e reter sua estrutura. De acordo com diferentes níveis de inter-relações entre as palavras, redes de co-ocorrência, redes sintáticas e redes semânticas são introduzidas agregando maior significado a representação.

(INGERSEN, 2008), baseia-se nos aspectos cognitivos da recuperação de informação para a proposta de seu modelo. O trabalho apresenta o princípio da polirepresentação, que representa os documentos, e sua respectiva semântica, através de diversos aspectos intra e inter documentos.

(SERRANO, 2006), propõe um modelo computacional de leitura de texto denominado *Cognitive Reading Indexing* (CRIM), inspirado em alguns aspectos da cognição humana tais como, percepção sequencial, temporalidade, memória, esquecimentos e inferências. O modelo não produz vetores, mas sim redes de conceitos.

(RANZATO, 2008), propõe um algoritmo para aprender representação de texto com base em aprendizado semi-supervisionado e formação de redes profundas (deep networks). O modelo é treinado sobre um corpus parcialmente rotulado, produzindo representações compactas dos documentos, enquanto retém informações sobre classes e acrescenta estatísticas sobre os termos.

(FISHBEIN, 2008), propõe um esquema de representação baseado em Representações Holográficas Reduzidas (Holographic Reduced Representation – HRR) para codificar tanto a estrutura semântica quanto a estrutura sintática dos documentos. Segundo o autor, mesmo agregando uma quantidade maior de informação, o método não causa aumento nos vetores de documentos, garantindo computação e armazenamento eficientes.

De forma semelhante à proposta apresentada neste trabalho, Lin (LIN, 2007) emprega conceitos de computação granular no tratamento do problema. Nesta proposta, os documentos assumem uma representação que inclui o conhecimento. Os grânulos são formados por conjuntos de palavras-chave com co-ocorrência freqüente. A representação granular é formada pela associação das palavras-chave a um conjunto de vértices em um complexo simplicial. (NGUYEAN, 2008) utiliza conjuntos aproximados (rough sets) nesta representação. Neste caso, nenhum modelo de representação específico é descrito. Apenas os resultados do processo de granulação são utilizados para agrupar documentos e enriquecer a representação dos documentos baseada no modelo tradicional baseado em "*bag of words*". (YAO & ZHONG, 2007) Yao utiliza *lattices* de conceitos nesta representação granular de documentos.

Dentre todas estas técnicas empregadas para a modelagem de documentos, destacam-se duas, atualmente muito utilizadas com o objetivo semelhantes ao proposto neste trabalho. As técnicas de Análise de Semântica Latente (Latent Semantic Analysis – LSA) (DEERWESTER, 1990) e Modelagem de Tópicos (Latent Dirichlet Allocation – LDA) (BLEI, 2003) , apresentadas no Capítulo 4, têm conseguido bons resultados no sentido de criar representações mais significativas para os documentos em uma coleção. Estas duas técnicas serão utilizadas como instrumentos para a análise do modelo proposto neste trabalho.

1.3 Hipótese

O modelo tradicional de documentos chegou ao seu limite. É difícil que consigamos melhorar a eficiência das ferramentas que lidam com dados do tipo texto utilizando o modelo baseado em *bag of words.* Também é difícil que consigamos criar novas abordagens para esta questão sem que tenhamos um modelo com maior representatividade. O modelo tradicional não é capaz de capturar características

fundamentais nos documentos que representa como, por exemplo, o relacionamento mantido entre as palavras. Neste sentido, vários modelos têm sido propostos, mas nenhuma das abordagens considera a possibilidade da captura do relacionamento das palavras através de uma abordagem que permita a variação do nível de granularidade (abstração) considerado. Um modelo com esta característica pode abrir novos caminhos para o desenvolvimento de ferramentas mais eficientes na manipulação da enorme quantidade de dados disponibilizados atualmente neste formato. Os princípios da computação granular envolvem este aspecto da construção do conhecimento, onde o conceito de granularidade da informação permite a navegação entre os seus vários níveis, entre outros aspectos. Com base nestes princípios podemos desenvolver ferramentas capazes de explorar esta característica, tornando o processo mais coerente com a capacidade humana em lidar com os vários níveis de conhecimento.

1.4 Objetivos do Trabalho

Este trabalho propõe um modelo de documento baseado em princípios da computação granular. O objetivo é apresentar um modelo, alternativo ao tradicional, com a capacidade de capturar as relações mantidas entre as palavras em diferentes níveis de granularidade. Esta característica irá permitir o desenvolvimento de outros tipos ferramentas para a manipulação da enorme quantidade de dados disponibilizada atualmente no formato textual. Também irá permitir que todos os aspectos relacionados a computação com grânulos, mencionados anteriormente, sejam explorados de forma efetiva. Os resultados alcançados demonstram que se trata de uma abordagem promissora para o desenvolvimento de ferramentas que auxiliem mais eficientemente a recuperação da informação e outras tarefas que envolvem texto.

1.5 Metodologia da Pesquisa

Trata o presente de um estudo de natureza binária: qualitativa e quantitativa, cujo caráter é exploratório, entendida como mais apropriada às questões aqui abordadas, sendo que ao final os resultados obtidos são mensurados. As informações coletadas foram analisadas e discutidas, sendo dispostas de acordo com as vertentes de ideias

apresentadas. A metodologia segue os princípios apresentados em (WAZLAWICK, 2008).

1.6 Contribuições e Originalidade

A principal contribuição deste trabalho é a proposta de um modelo de documentos, alternativo ao tradicional baseado em *bag of words* , que possui como característica principal a possibilidade de captura das relações mantidas entre as palavras contidas nos documentos considerando seus vários níveis de granularidade. Tal característica pode viabilizar a construção de ferramentas mais eficientes, capazes de explorar os diversos níveis de conhecimento, disponibilizados por estes níveis de granularidade. Apesar de existirem outras propostas de modelos alternativos para documentos, não foi encontrado na literatura uma abordagem que considere este aspecto. Outras contribuições incluem o teste do modelo e a comparação do seu desempenho em relação a abordagens semelhantes quanto ao objetivo de representação das relações entre as palavras. Também incluem a análise da aplicação do modelo às tarefas de classificação e agrupamento de documentos.

1.7 Organização do Trabalho

Além deste capítulo introdutório, este trabalho apresenta mais 6 capítulos organizados da seguinte forma: o Capítulo 2 introduz os principais conceitos de computação granular e apresenta as suas principais ferramentas. O Capítulo 3 apresenta os fundamentos e algoritmos da técnica de agrupamento espectral, utilizada para a construção dos grânulos de palavras, base do modelo proposto. O Capítulo 4 descreve o modelo de documentos proposto e apresenta alguns resultados comparativos com técnicas atuais e com objetivos semelhantes. Os capítulos 5 e 6 apresentam a aplicação do modelo proposto às tarefas de classificação e agrupamento de documentos, repectivamente. Finalmente, o Capítulo 7 apresenta algumas conclusões e sugestões para trabalhos futuros.

2. COMPUTAÇÃO GRANULAR

Embora o nome Computação Granular seja relativamente recente, a noção de granulação tem surgido sob diferentes nomes em muitos campos relacionados. Dividir e conquistar, teoria dos conjuntos fuzzy (fuzzy sets) e conjuntos aproximados (rough sets), espaços topológicos, computação de intervalos, quantização, compressão de dados, *chunking*, análise de agrupamentos, aprendizado de máquina e banco de dados são alguns exemplos. Muitas aplicações de computação granular têm surgido em várias áreas, tais como, medicina, economia, finanças, engenharias, etc.

A granulação mostra-se como uma metodologia de solução de problemas profundamente enraizada no pensamento humano. Muitas das coisas do dia a dia são rotineiramente granuladas em "sub coisas", como por exemplo, o tempo e o espaço. O processo é intrinsecamente fuzzy, vago e impreciso. Trata-se de uma abordagem simples e direta, para a computação em larga escala, que se divide em 3 etapas:

1. Dividir o problema e sub-tarefas, ponto a ponto ou nível a nível;
2. Abstrair o problema em espaços conceituais ou de conhecimento, em um ou vários níveis;
3. Integrar as soluções das sub-tarefas (espaços de conhecimento) e dos vários níveis.

Este capítulo destina-se a apresentação das principais questões inerentes a computação baseada em grânulos.

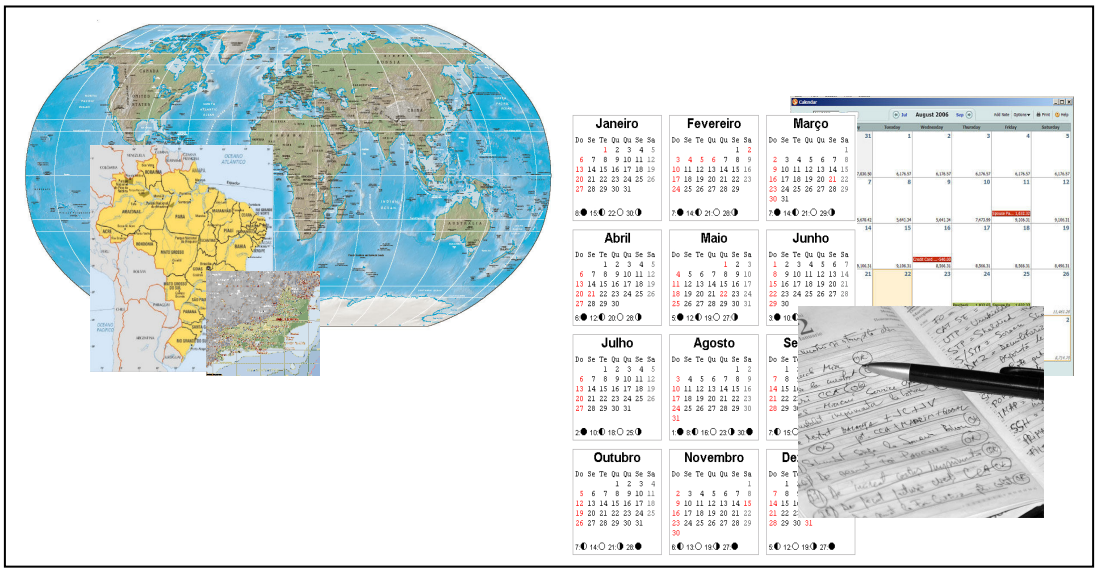


Figura 1. Exemplo de Granulação do Espaço e do Tempo

2.1 Granulação da Informação

O significado original da granulação nos dicionários é o ato ou processo de transformação de algo em grânulos (LIN, 2004) . É o processo para transformar um objeto em objetos menores. Zadeh adota esta idéia para decompor um universo em grânulos em um de seus artigos, no início da granular computação em 1996 (ZADEH, 1997). Pela definição de Zadeh, "Granulação envolve uma decomposição do todo em partes. Por outro lado, a organização envolve a integração das partes no todo". Com base nesta definição, são duas as operações na computação granular, a granulação e a organização.

Podemos adotar ainda, uma visão mais geral de granulação. Ou seja, granulação envolve os processos nas duas direções: construção e decomposição. A construção envolve o processo de formação de um grânulo maior e de nível superior a partir de sub-grânulos menores e de menor nível. É um processo *bottom-up*. A decomposição envolve o processo de dividir um grânulo em grânulos menores e de nível mais baixo, e é semelhante a definição do dicionário e a definição de Zadeh de granulação. Este é um processo *top-down*. A razão para consideramos uma visão mais geral e ampla de granulação é que a construção e a decomposição são estreitamente relacionadas. Quando se escolhe uma granulação especial, os benefícios e a eficácia de uma direção estão estreitamente relacionados a sua direção oposta. Se considerarmos o funcionamento da decomposição sem a contrapartida da construção, podemos acabar com uma operação de decomposição muito eficiente e uma construção muito ineficiente.

2.2 Espaços de Aproximação

Técnicas de computação granular extraem conhecimento ou padrões conceituais de contextos inexatos ou imprecisos (POLKOWSKI, 2002). Estes padrões são, geralmente, aproximados por um conjunto de certos e exatos componentes ou grânulos. O processo de mineração de dados, por exemplo, pode ser definido como um sistema de computação granular. Como já mencionado, a computação granular é um processo duplo de solução de problemas: após um processo de granulação, um processo de integração destes grânulos resulta na solução do problema. Estes grânulos formam um espaço, denominado espaço de aproximação, que é utilizado para aproximar conceitos e

padrões (SKOWRON, 2004). Modelos típicos de espaços de aproximação em computação granular incluem conjuntos fuzzy (*fuzzy sets*), conjuntos aproximados (*roughs sets*), conjuntos próximos (*near sets*) e sistemas de vizinhança (*neighborhood system*).

Conjuntos fuzzy (ZADEH, 1979) (ZADEH, 1996) (ZADEH, 1997) lidam com a informação vaga que está caracterizada nas funções de pertinência. A inferência, neste espaço de aproximação, é conduzida pela computação destas funções de pertinência.

Conjuntos aproximados (PAWLAK, 1982) (PAWLAK 1998) lidam com sistemas de informação inexata. Em um sistema de informação, um tabela de decisão consiste de um conjunto de objetos que são caracterizados por um conjunto de atributos de condição e atributos de decisão. Os objetos em uma tabela de decisão podem ser classificados em classes de equivalência, pela definição de uma relação de indiscernibilidade, e as classes de equivalência podem ser exploradas para aproximar conjuntos crisp de objetos (POLKOWSKI, 2002).

Conjuntos próximos (PETERS, 2006) (PETERS, 2007) estendem o conceito de conjuntos aproximados pela computação das r características de vizinhança que aproximam a informação sobre uma amostra do universo. Padrões são reconhecidos por uma família de características vizinhas em relação as características associadas as funções "de exploração".

Espaços de aproximação formados tanto por conjuntos aproximados quanto por conjuntos próximos são baseados em relações de equivalência de indiscernibilidade. Sistemas de vizinhança (LIN, 1997) (LIN, 1998) estendem as relações de equivalências para relações binárias e, assim, têm muito mais aplicações em campos como banco de dados e recuperação de informação (LIN, 1989) (LIN, 2004). Trata-se de uma estrutura matemática de computação granular para modelar grânulos, relacionamentos entre grânulos assim como a computação com os grânulos e/ou entre os grânulos. Uma vizinhança é imprecisamente definida como um subconjunto de objetos "próximos" e qualquer subconjunto de objetos pode ser aproximado por um conjunto de vizinhanças. Um sistema de vizinhança define um conjunto de relações binárias e um conjunto de relacionamentos binários pode ser usado para definir um sistema de vizinhança.

2.2.1 Conjuntos Fuzzy (Fuzzy Sets)

Conjuntos fuzzy e a granularidade da informação foram propostas por Zadeh em 1979 (ZADEH, 1979) (ZADEH, 1997) para fornecer uma base para a construção de teorias gerais que evidenciassem e permitissem a natureza fuzzy do mundo real. Na granularidade da informação, os grânulos de dados são vistos como um proposição na forma geral de

$$g: \text{“X é G” é } \lambda,$$

onde X é um variável que toma valores em um universo de discurso U, G é um conjunto fuzzy do universo e é caracterizado por suas funções de pertinência μ_G, e λ é uma probabilidade fuzzy caracterizada por uma distribuição de possibilidade sobre um intervalo unitário.

Um exemplo típico de tal proposição é dado no universo U de números reais, como segue:

$$g: \text{“X é pequeno” é provável},$$

onde o subconjunto fuzzy G é “pequeno” e a probabilidade fuzzy λ é um subconjunto fuzzy do intervalo unitário.

Cada subconjunto fuzzy G do universo pode ser geralmente entendido como um conceito, por exemplo, “pequeno”, “muito grande”, “velho”, “jovem”, etc. Todos os conceitos são colocados juntos para formar o espaço de conceitos, denotado por C,

$$C = \{G_1, G_2, ..., G_3\}.$$

Cada proposição g é considerada como uma evidência e todas as evidências comprometem uma coleção de proposições, $V = \{g_1, g_2, ..., g_N\}$, onde g_i: “X é G_i” é λ_i, i =1,2,...,N.

Grânulos condicionais podem ser representados como regras fuzzy tais como “Se X = u então Y é G”. Estes tipos de grânulos também podem ser representados como funções de pertinência dos conjuntos fuzzy ou expressões destas funções.

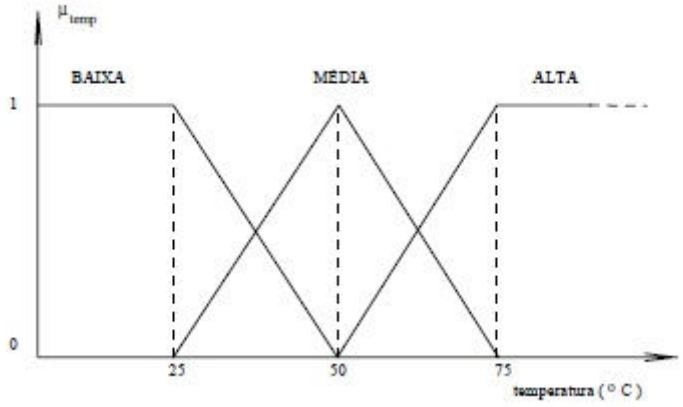


Figura 2. Conjuntos fuzzy representando a granulação do conceito temperatura

Em um sistema de conjuntos fuzzy, um subconjunto de um universo pode ser aproximado por todos os subconjuntos fuzzy deste universo. Suponha U um universo. O espaço de conceitos V (uma família de subcojuntos fuzzy de U) é chamada uma cobertura de U e cada conceito em V é chamado uma cobertura de U. Dizemos que os conjuntos fuzzy V_1, V_2, ..., V_n, identificados por suas funções de pertinência definidas em U, formam um partição fuzzy de U com as seguintes condições: cada função de pertinência é contínua e, para cada u em U, $\sum_{i=1} \mu vi(u) = 1$, onde μvi é a função de pertinência de V_i.

Uma aproximação crisp com um limiar α de um dado subconjunto A de U pode ser representado como a união ou a combinação de uma série de conceitos em um espaço de conceitos, tais como $A_\alpha = \{V_i \mid \mu v_i (a) >= \alpha$, para a em A$\}$; ou $A_\alpha = f(\mu v_i (a) >= \alpha$, para a em A), onde f é uma combinação específica de funções.

O melhor exemplo de aproximação em conjuntos fuzzy é a computação com palavras encontrada em detalhes em (ZADEH, 1996) (ZADEH, 1997).

2.2.2 Conjuntos Aproximados (Rough Sets)

Um sistema de informação SI é definido como: SI = <U,C,D, $\{V_u\}\, a \in C \cup D$, f >, onde U = $\{u_1, u_2, \ldots, u_3\}$ é um conjunto não vazio de objetos (tuplas), chamado tabela de decisão, C é um conjunto não vazio de atributos condicionais e D é um conjunto não vazio de atributos de decisão e C ∩ D = ø. V_a é o domínio de a com pelo menos dois

elementos. f é a função: U X (C $\cup$ D) $\rightarrow$ V = $\cup\, a \in C \cup D\, V_a$, mapeando cada par de objetos e atributos a um valor de atributo.

Suponha A $\subseteq$ C $\cup$ D, e t, s $\in$ U. Uma relação binária R_A, chamada relação de *indiscernibilidade* é definida como: R_A = [< t, s> $\in$ U x U | $\forall a \in$ A, t[a] = s[a]}, onde t[a] indica o valor do atributo a $\in$ A do objeto t. A relação de *indiscernibilidade*, denotada por IND(A), é uma relação de equivalência sobre U. Com esta relação de equivalência R_A pode-se construir um sistema de vizinhança de relações binárias <U, IND(A)>.

Suponha X um subconjunto de U em um sistema de informação e representando um conceito. È possível a determinação de suas aproximações inferior e superior usando-se subconjuntos elementares de U/IND(A). A aproximação inferior de X contém todos os objetos em U que estão *definitivamente* incluídas em X, enquanto a aproximação superior de X contém todos os objetos de U que estão *potencialmente* incluídos em X. Por outro lado, o complemento da aproximação superior de X contém todos os objetos em U que estão *definitivamente excluídos* de X. Como um conceito, X tem sua aproximação inferior como uma região positiva, enquanto o complemento de sua aproximação superior como sua região negativa. Da perspectiva do aprendizado de máquina ou da classificação, a região positiva contém todos os exemplos positivos de X, enquanto a região negativa possui todos os exemplos negativos de X. A região localizada entre as regiões positiva e negativa é chamada região *limite* ou *fronteira* de X

A principal aplicação da aproximação de conceitos usando grânulos na teoria dos conjuntos aproximados é a redução do tamanho de grandes tabelas de decisão pela redução dos seus atributos. Um atributo condicional a $\in$ C é um atributo central de C em U com respeito a D se

$$\forall X \in U / IND(D),\ \text{Low}_{IND(C)}(X) = \text{Low}_{IND(C-\{a\})}(X).$$

Um reduto de um conjunto de atributos condicionais é um subconjunto mínimo destes atributos com a mesma capacidade classificatória. Assim, um reduto de atributos condicionais pode ser usado para representar um conjunto inteiro de atributos condicionais.

Formalmente, um subconjunto A de C, $A \subseteq C$, está definido com um reduto de C em U em relação a D se

$$\forall X \in U / IND(D), \text{Low}_{IND(C)}(X) = \text{Low}_{IND(C)}(X)$$

e

$$\forall B \subset R, \text{Low}_{IND(B)}(X) \neq \text{Low}_{IND(C)}(X).$$

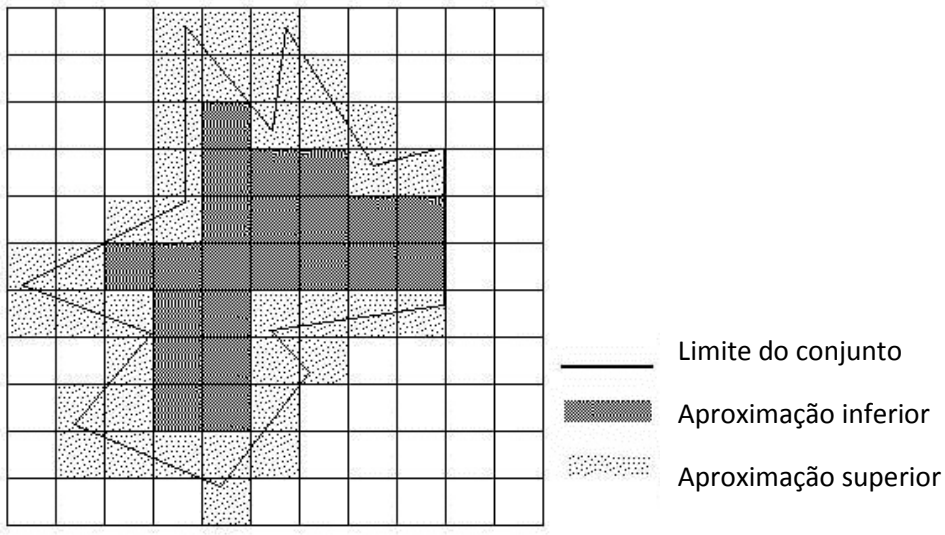


Figura 3. Conjunto aproximado com suas aproximações inferior e superior

Um atributo condicional $a \in C$ é dito um atributo de reduto se $\exists B \subseteq C$, B é um reduto de C e $a \in B$. Um reduto R de C é dito reduto mínimo de C se $\forall Q \subset R$, Q não é reduto de C.

Considere $P \subseteq C \cup D$ e $Q \subseteq C \cup D$, a região positiva de Q em relação a P, denotada por POS_P (Q), definida como $POS_P(Q) = \cup_{X \in U/IND(Q)} \text{Low}_{IND(P)}(X)$, que contém todos os objetos em U que podem ser classificados usando-se a informação contida em P. Com esta definição, o grau de dependência de Q a partir de P denotada por γ_P (Q), é definida como $\gamma_P(Q) = POS_P(Q)/|U|$, onde |X| representa a cardinalidade do conjunto X.

Investigando-se a granularidade do conhecimento, a partir do ponto de vista da teoria dos conjuntos aproximados, pode-se perceber que trata-se de uma técnica de computação granular baseada em partições (SKOWRON, 2004). Basicamente, suponha U com um universo finito e não vazio. Suponha $R \subseteq U \times U$ uma relação de

equivalência sobre U. O par <U,R> é denominado um espaço de aproximação. A relação R é um sistema de vizinhança binário especial; pode ser convenientemente representado por um mapeamento B_R de U no conjunto potência de U, e definido como $B_R : U \rightarrow 2^U$, $B_R(x) = [x]_R = \{ y \in U \mid xRy\}, \forall x \in U$.

$[x]_R$ é a classe de equivalência contendo x, denominada subconjunto elementar. A família de todas as classes de equivalência, denotada por $U/R = \{[x]_R \mid x \in U\}$, define uma partição do universo U, isto é, uma família de subconjuntos disjuntos cuja união cobre todo o universo.

	a1	**a2**	**a3**	**a4**
o1	1	1	1	1
o2	1	0	0	0
o3	1	1	0	1
o4	0	1	0	1
o5	1	0	1	0

Classes de equivalência para o atributo {a1}: {o1,o2,o3,o5}{o4}
Classes de equivalência para o atributo {a4} : {o1,o3,o4}{o2,o5}
Classes de equivalência para o atributo {a1,a2,a4}: {o1,o3}{o2,o5}{o4}

Figura 4 . Tabela de decisão e classes de equivalência

Do ponto de vista da teoria dos conjuntos aproximados, cada classe de equivalência é considerada um grânulo inteiro, ao invés de muitos elementos individuais. Com estes subconjuntos elementares, qualquer subconjunto de U pode ser aproximado.

Suponha X um subconjunto de U e R uma relação de equivalência sobre U. A aproximação inferior de X com base em R, denotada por $Low_R(X)$, é definida como

$$Low_R = (X) = \cup \{Y \in U/R \mid Y \subseteq X,$$

que contém todos os subconjunto elementares de U que estão completamente incluídos em X.

A aproximação superior de X com base em R, denotada por $Upp_R(X)$, é definida como

$$Upp_R = (X) = \cup \{Y \in U/R \mid R \cap X \neq \emptyset\},$$

que contém todos os subconjuntos elementares de U que possuem interseção não vazia com X.

Neste modelo de partição, estas formas de aproximações são equivalentes ao interior e ao exterior definidos em um sistema de vizinhança binário, respectivamente. Entretanto, não são apropriadas para generalizar o sistema de vizinhança. Por exemplo, em um espaço topológico, Upp_R (X) é sempre o universo todo, não importa o que se escolha.

2.2.2.1 Espaço Quociente (Quotient Space)

Existem múltiplas reduções em um sistema de informação. Cada redução pode ser usada para definir uma relação de equivalência. E pode ser usada para construir um espaço quociente. Da perspectiva matemática e topológica, um espaço quociente é um espaço onde pontos equivalentes são postos juntos e assim um novo espaço pode ser construído.

Suponha que o universo de discurso U é um espaço topológico e R é uma relação de equivalência sobre U. U/R, o conjunto de todas as classes de equivalência de R, é chamado um espaço quociente, onde a topologia sobre U/R é definida como um subconjunto $X \subseteq U/R$ é aberto se e somente se sua união é aberta.

As propriedades topológicas dos espaços quocientes, assim como os relacionamentos entre os espaços quocientes têm sido estudados para identificar e selecionar atributos para formar reduções de alta qualidade. A teoria dos espaços quocientes tem sido explorada em segurança de dados, segurança de redes, entre outros problemas (ZHANG, 2003).

2.2.3 Conjuntos Próximos (Near Sets)

Conjuntos próximos têm sido estudados e aplicados em reconhecimento de objetos pela classificação de amostra de objetos através de funções de exploração (probe functions) associadas as características dos objetos (PETERS, 2006) (PETERS, 2007).

Considere A um conjunto de características de objetos em um conjunto X. A cada $a \in A$ está associada uma função f_a, chamada "exploratória", para mapear X a algum conjunto V_{fa} (faixa de f_a). O valor de $f_a(x)$ é uma medida associada com uma característica a de um objeto $x \in X$. Para $B = \{b_1, b_2, ..., b_m\} \subseteq A$ e $x \in X$, a assinatura de x é denotada $Inf_B(x) = \{b, f_b(x) \mid b \in B\}$ e identificado com o vetor $(f_{b1}(x), f_{b2}(x), ..., f_{bm}(x))$ de valores de funções exploratórias em características em B.

De acordo com Peters (PETERS, 2006) (PETERS, 2007A), um espaço de aproximação generalizado A com relação a $B \subseteq A$ é a tupla $GAS = \{U, A, N_r, v_B\}$, onde U é o universo de objetos, A é um conjunto de funções exploratórias, N_r é uma família de funções de vizinhança e v_B é uma função de sobreposição definida como

$$V_B = 2^U \times 2^U \rightarrow [0,1]$$

V_B representa o grau de sobreposição entre os conjuntos de objetos com as características definidas por B. Para cada $B \subseteq A$ de funções exploratórias, uma relação binária é definida como $R(B) = \{ (x,y) \in U \times U \mid \forall f \in B, f(x) = f(y)\}$.

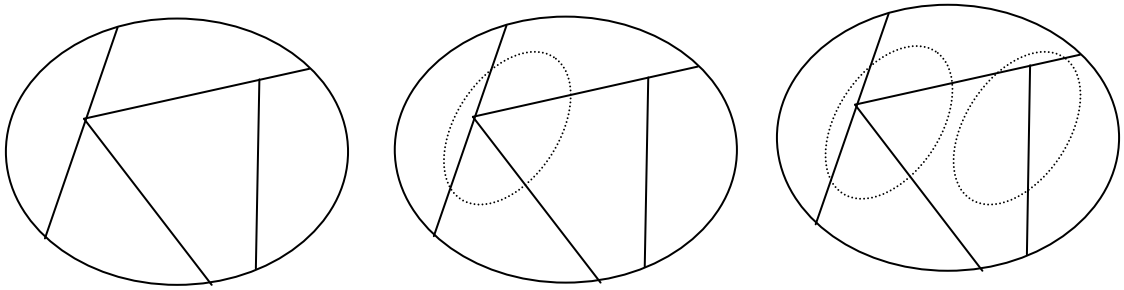
É possível se perceber, da perspectiva da teoria dos conjuntos aproximados, que R(B) é de fato a relação de indiscernibilidade. Entretanto, a classe de equivalência $[x]_B$, contendo x induzidos por B, pode ser representada como $[x]_B = y \in U \mid \forall f \in B, f(y) = f(x)\} \subseteq B$. Todos os objetos em $[x]_B$. é dito B-indiscernível (indiscernível em relação a B).

Para um inteiro $r \in [1, |A|]$, uma família de vizinhos $N_r(A)$ é definida como $N_r(A) = \cup \{[x]_B \mid B \subseteq Pr(A)\}$, onde $P_r(A) = \{B \subseteq A \mid |B| = r\}$. O inteiro r denota o número de características usadas para construir famílias de vizinhos.

Qualquer $X \subseteq U$ pode ser aproximado pelo uso de famílias de vizinhos. Considere $[x]_{Br}$ que denota a vizinhança formada pelas classes de equivalência induzidas pela r características de B. Uma aproximação $N_r(B)$-inferior de X é definida como $N_r(B)_* X = \cup \{[x]_{Br} \mid x : [x]_{Br} \subseteq X\}$. Uma aproximação $N_r(B)$-superior de X é definida como $N_r(B)^* X = \cup \{[x]_{Br} \mid x : [x]_{Br} \cap X \neq \emptyset\}$.

É fácil perceber que $N_r(B)_* X \subseteq N_r(B)^* X$. A diferença entre eles é denominada região limite e é definida como $BND_{Nr(B)}(X) = N_r(B)^* X - N_r(B)_* X = \{x \in N_r(B)^* X \mid x \notin N_r(B)_* X\}$.

Um conjunto X é definido com um conjunto próximo, relativo a família de vizinhos escolhida $N_r(B)$, se e somente se $BND_{Nr(B)}(X) \neq \emptyset$. Isso mostra que todo conjuntos aproximado é um conjunto próximo, mas nem todo conjunto próximo é um conjunto aproximado (PETERS, 2007A).



(a) (b) (c)

Figura 5. (a) Partição de um conjunto. (b) Conjunto Aproximado. (c) Conjuntos Próximos.

2.2.4 Sistemas de Vizinhança (Neighborhood Systems)

De acordo com (LIN, 1997) e (LIN, 1998), espaços de aproximação em computação granular podem ser matematicamente estruturados como sistemas de vizinhança, os quais são abstraídos da noção geométrica de "próximo" ou "distância desprezível". Aproximadamente, um sistema de vizinhança associa cada objeto no universo a uma (possivelmente vazia, finita ou infinita) família de subconjuntos não vazios, denominados vizinhança, para representar a semântica do "próximo". A vizinhança desempenha o papel fundamental em computação granular e pode ser considerada com grânulo. Uma vizinhança é precisamente um termo topológico para grânulo e um sistema de vizinhança é vagamente um espaço topológico.

Noções fundamentais sobre sistemas de vizinhança são apresentadas em (LIN, 1998A). Suponha que U é o universo de discurso e p é um objeto em U. Um objeto x é um vizinho de p se x está "próximo" a p. A noção de próximo é de julgamento subjetivo e pode ser semanticamente interpretada de diferentes formas (por exemplo, se satisfaz algum relacionamento ou condição). Para um caso específico, consideremos o relacionamento binário R em U, x é "próximo" a p, se xRp.

A vizinhança de p, denotada por N(p), em U é um subconjunto não vazio de U que pode conter ou não conter p. Um sistema de vizinhança de p, denotado por NS(p), em U é uma família de vizinhanças de p. Se p não tem vizinhança, então NS(p) é uma família vazia.

Um sistema de vizinhança de U, denotado NS(U), é a coleção de NS(p) $\forall p \in U$. Formalmente, NS(U) = NS(p) | $\forall p \in U$}.

Considere dois sistemas de vizinhança de U, digamos $NS_1(U)$ e $NS_2(U)$. $NS_1(U)$ é dito um refinamento de $NS_2(U)$, se para qualquer vizinhança N_1 em $NS_1(U)$, existe uma vizinhança N_2 em $NS_2(U)$ tal que $N_1 \subseteq N_2$. Da mesma forma, NS_2 é dito uma forma grosseira de $NS_1(U)$.

Um sistema de vizinhança é induzido a partir de vizinhos fundamentais de objetos, que são imprecisamente como "próximos". Se o "próximo" é explicitamente definido de acordo com uma relação binária, o resultado do sistema de vizinhança é chamado sistema de vizinhança binária.

Um sistema de vizinhança binária BNS é definido como BNS = <U, B, V> onde U e V são dois universos, e B é a função de mapeamento de U em V: U→V.

$\forall v \in V$, considere B, que denota o subconjunto de U, que é a fonte de v sob o mapeamento B, isto é, $B_v = \{u \mid u \in U \text{ e } B(u) = v\}$. Cada B é denominado vizinhança básica ou vizinhança binária.

Cada sistema de vizinhança BNS = <U, B, V> define uma relação binária $R \subseteq V \times U$: $\forall v \in V$ e $u \in U$, uRv se e somente se $u \in B(v) \subseteq U$. Assim a relação binária R é totalmente determinada por B. Reciprocamente B é totalmente determinada por uma relação binária R.

Se B é uma relação binária $R \subseteq V \times U$, então para cada objeto $v \in V$, a vizinhança binária em termos de v será $N_v = \{ u \mid u \in U \text{ e } uRv \}$. Este é um subconjunto de U, cujos elementos são relacionados a v por R.

Se U = V e a função de mapeamento B é considerada uma relação binária, então temos um sistema de vizinhança simples, representado por BNS = <U,R>. Por simplicidade, chamaremos este sistema de vizinhança binário com um relação binária de sistema de vizinhança com relação binária (BRNS).

Consideremos dois BRNS, BRNS1 e BRNS2, construídos sobre o mesmo universo, onde BRNS1 = <U, R1, V> e BRNS2 = <U, R2, V>. BRNS1 é dita um

refinamento de BRNS2, se R2 $\subseteq$ R1. Por outro lado, BRNS1 é dita uma forma grosseira de de BRNS2, se R1 $\subseteq$ R2.

Considere BNS = <U, B, V> um sistema de vizinhança binário em V e X $\subseteq$ U. X pode ser aproximado por vizinhanças de BNS. De acordo com a definição de espaços topológicos, a aproximação interior de X é definida como Interior(X) = $\cup_{x \in X}$ { N(x) | N(x) $\subseteq$ X}. De modo similar, a aproximação exterior de X pode ser definida como Exterior(X) = $\cup_{x \in X}$ { N(x) | N(x) $\cap$ X $\neq \emptyset$}.

O interior de um subconjunto do universo pode ser aproximado pela união de todas as vizinhanças do universo que estão completamente contidas no subconjunto, enquanto que o exterior de um subconjunto pode ser aproximado pela união de todas as vizinhanças que possuem interseção não vazia com o subconjunto.

O subconjunto X de U é aberto se $\forall$ p $\in$ X, existe uma vizinhança N(p) $\subseteq$ X. X é fechado se todos os seus complementos são abertos. Um sistema de vizinhança de p, NS(p), é aberto se todas as vizinhanças de p são abertas. Um sistema de vizinhança de U, NS(U). é aberto, se $\forall$ p $\in$ U, NS(p) é aberto.

Se o universo U é um espaço topológico e um sistema de vizinhança de U, NS(U), é aberto, então NS(U) é também um espaço topológico. Neste caso, tanto NS(U) quanto a coleção de conjuntos abertos são chamados topologia.

Um objeto p é um ponto limite de um conjunto E, se toda a vizinhança de p contém um ponto de E exceto p. O conjunto de todos os pontos limite de E é chamado conjunto derivado. E o seu conjunto derivado é um conjunto fechado.

É fácil verificar que, NS(U) é discreto se todo NS(p) é único. Uma cobertura de U é um sistema de vizinhança aberto NS(U).

Uma partição de U é uma cobertura de U, onde p e q $\in$ U cada NS(p) = NS(q) ou NS(p) $\cap$ NS(q) = $\emptyset$.

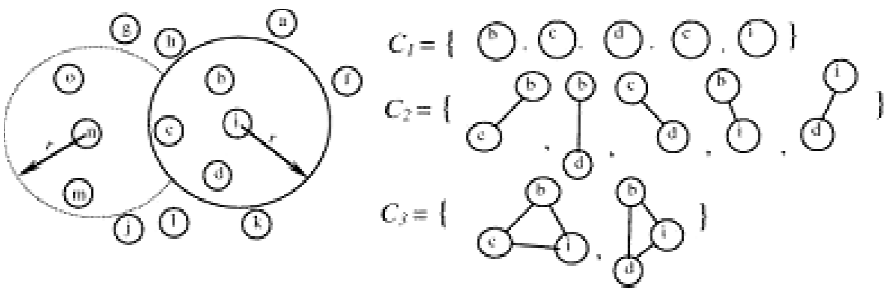


Figura 6. Sistema de Vizinhança

2.3 Estrutura Unificada de Espaços de Aproximação

Usando sistemas de vizinhança de subconjuntos e quocientes, podemos derivar o sistema de vizinhança de regras extraídas do sistema de informação de Pawlak. Vamos considerar um caso especial de sistema de vizinhança binário BNS = <U, R>, onde R é uma relação de equivalência binária em U x U. Para um elemento u $\in$ U, por definição, a vizinhança binária de U é definida como $N_u = \{x \mid x \in U \text{ e } x\ Ru\}$.

Como R é uma relação de equivalência, N_u é uma classe de equivalência [u] induzida por R. Então, o sistema de vizinhança (a coleção de todas as vizinhanças) será a família de classes de equivalência induzidas por R.

Pode-se perceber que o espaço de aproximação formado por near sets é também baseado em uma relação de equivalência, a relação de discernibilidade com restrições quanto ao número de características.

Intuitivamente, sistemas de vizinhança manipulam noções de proximidade, analogia e aproximação. Podemos usar uma relação binária fuzzy, uma função de distância, uma medida de dissimilaridade ou uma medida de similaridade para descrever o grau de similaridade ou proximidade. De fato, estas medidas quantitativas têm sido largamente utilizadas em uma vasta gama de áreas de pesquisa, tais como análise de dados e recuperação de informação (YAO, 2006). Os sistemas de vizinhança podem ser aplicados a situações onde o significado da distância ou a função de similaridade não é claro. Também pode ser aplicado quando a informação qualitativa (por exemplo, a ordem implícita nos valores numéricos) é mais útil do que os valores numéricos precisos. A noção de sistemas de vizinhança fornece uma ferramenta conveniente e flexível para representar similaridade e pode ser usada para descrever informações tanto quantitativamente quanto qualitativamente.

2.4 Espaços de Aproximação Generalizados

A teria clássica dos conjuntos aproximados (rough sets) está baseada em relações de equivalência, que dividem o universo de objetos em classes disjuntas. Por definição, uma relação R $\subseteq$ U x U é dita de equivalência se apresenta as propriedades reflexiva, simétrica e transitiva. Na prática, estas restrições se mostram muito rigorosas para algumas aplicações. A natureza dos conceitos em muitos domínios é imprecisa e pode ser sobreposta. Este é o caso, por exemplo, do universo de documentos e palavras.

Com base nesta constatação, Skowron (SKOWRON, 2004) introduziu o espaço de tolerância generalizado transformando a relação de equivalência R em uma relação de tolerância, onde a propriedade transitiva não é requerida. Formalmente, o espaço de aproximação generalizado é definido como pela quádrupla *A = (U, I, v, P)*, onde:

- U é um universo não vazio de objetos;

- I : U → P(U), onde P(U) é o conjunto potência de U, é uma função de incerteza satisfazendo as condições:

(1) $x \in I(x)$ para $x \in U$;

(2) $y \in I(x) \Leftrightarrow x \in I(y)$ para qualquer $x, y \in U$. Assim, a relação $xRy \Leftrightarrow y \in I(x)$ é relação de tolerância e I(x) é uma classe de tolerância de x.

- v: P(U) x P(U) → [0,1] é uma função de inclusão vaga. A inclusão vaga mede o grau de inclusão entre dois conjuntos. A inclusão vaga deve ser monótona (preserva uma relação de ordem) em relação ao segundo argumento, isto é, se $Y \subseteq Z$, então $v(X,Y) \leq v(X,Z)$ para $X, Y, Z \subseteq U$.

- P : I(U) → {0,1} é uma função de estruturalidade.

Junto com a função de incerteza I, a função de inclusão vaga define uma função de pertinência imprecisa para $x \in U$, $X \subseteq U$ por $\mu_{I,v}(x,X) = v(I(x), X)$. As aproximações superior e inferior para qualquer $X \subseteq U$ em A, denotada por $L_A(X)$ e $U_A(X)$, são respectivamente definidas como:

$$L_A(X) = \{ x \in U : P(I(x)) = 1 \wedge v(I(x), X) = 1\}$$
$$U_A(X) = \{ x \in U : P(I(x)) = 1 \wedge v(I(x), X) > 0\}$$

Com estas definições, espaços de aproximação generalizados podem ser usados em qualquer aplicação onde I, v e P sejam apropriadamente determinadas.

2.5 Integração da Informação

Para cada granulação de um problema, uma ação casada, denominada construção ou integração das subsoluções se faz necessária. (LIN, 2007) trata desta operação fazendo uso do conceito de functor para ilustrar a natureza dos problemas.

Functor, em teoria das categorias, é um mapeamento entre categorias que preserva suas estruturas. Os functores podem ser entendidos como homomorfismos na categoria de todas as categorias pequenas (ou seja, a categoria que tem como objetos todas as categorias compostas por objetos que são conjuntos).

Um *functor* (*covariante*) F da categoria C para a categoria D:

1. associa para cada objeto x em C um objeto F(x) em D;
2. associa para cada morfismo f: x → y um morfismo F(f) : F(x) → F(y);

tal que as seguintes propriedades valem:

1. $F(id_x) = id_{F(x)}$
2. F (g o f) = F(g) o F(f) para todos os morfimos f: x → y_e.

Material completo sobre functores e álgebra homológica pode ser encontrado em (ROTMAN, 2009).

2.5.1 Integração sem Estrutura de Informação Adicional

Considere o universo U com um conjunto Z = {..., 1, 0, 1, ...} de inteiros. Suponha que U tenha sido decomposto em dois subproblemas, denominados {...,-2, 0, 2, ...} e {..., -3, 1, 3, ...}. Tais conjuntos são denotados por E e O, respectivamente.

1. A estrutura granular está representada na seguinte coleção:
   GrS = {{...,-2, 0, 2, ...} , {..., -3, 1, 3, ...}}

Informalmente, GrS é uma coleção de caixas brancas, onde o conteúdo dos subconjuntos está visível. Observe que os dois grânulos são disjuntos, portanto a coleção de grânulos (subconjuntos) é um conjunto clássico. Caso estes subconjuntos

não fossem disjuntos, a estrutura matemática da coleção não poderia não ser um conjunto clássico.

2. A estrutura quociente é a coleção Q = {E,O}

A coleção Q é um conjunto crisp de dois elementos. Informalmente, está é uma coleção de caixa pretas, onde o conteúdo dos subconjuntos está invisível (escondida).

3. A estrutura interna dos dois grânulos {{...,-2, 0, 2, ...} , {..., -3, 1, 3, ...}} é, para ambos, o conjutno Z = {..., 1, 0, 1, ...} de inteiros. Eles serão denotados por Int(E) = Z e Int(O) = Z, respectivamente.

As evidências de que as duas cópias de inteiros estão incorporadas em Z como subconjuntos (grânulos) estão expressas pelos seguintes mapeamentos:

a. $Z \rightarrow E \subseteq Z$

b. $Z \rightarrow O \subseteq Z$

Um ser humano que observa a partir de E, pensa que o elemento $2 \in E$ é o $1 \in Z$.

Do ponto de vista de solução de problemas, a estrutura quociente representa a fórmula (instruções de alto nível) de como colocar as soluções dos subproblemas dentro da solução total. O conjunto quociente é o "programa principal" que consiste de chamadas a subprogramas. Neste caso, Q é o programa principal que representa as instruções de alto nível.

4. O conceito de integrações de informação será expresso pela equação $Int(E) = Z \cup Int(O) \rightarrow ? \rightarrow Q$ (estrutura quociente).

` O desconhecido (Unknown) universo ? é sabido como:

i. ter sido decomposto em dois subconjuntos desconhecidos mutuamente disjuntos, E e O, e

ii. a estrutura interna destes conjuntos são parcialmente conhecidas, denominadas Int(E) = Int(O) e são o conjunto Z de inteiros.

iii. duas cópias de inteiros Z, Int(E) e Int(O) são mapeados na estrutura quociente Q = {E,O}, onde Int(E) e Int(O) são mapeados na estrutura quociente Q = {E,O}, onde Int(E) e Int(O) são mapeados para E e O respectivamente.

5. Considere U a união disjunta de Int(E) e Int(O), uma vez que Q consiste de dois elementos ou de forma equivalente, $Z \times Q$.

Note que $U = Z \times Q$, equivalente a Z (em teoria dos conjuntos), é a integração final. Esquematicamente, temos a seguinte situação:

$$Z = \begin{cases} Int(E) \rightarrow E \\ Int(O) \rightarrow O \end{cases} \qquad (\subseteq)\ U \rightarrow Z_2$$

O produto cartesiano Z × Q representa U. Do ponto de vista teórico (em teoria dos conjuntos, o conjunto construído Z × Q e Z são equivalentes.

2.5.2 Integração com Estrutura de Informação Adicional

Considere uma segunda visão sobre o mesmo universo apresentado na seção 2.5.1. Desta vez, o universo carrega informação adicional, uma estrutura aditiva de inteiros, denominada grupo aditivo (Z, +). Este universo é denotado por (U, +). Então:

1. Int(E) é o grupo aditivo (Z,+) e Int(O) é um conjunto Z de inteiros.
2. A estrutura quociente (Q,+) = (E,O,+} é um grupo aditivo: E+E = O = O+E = O, O+O=E. Este (Q,+) é geralmente chamado de inteiro mod 2 e denotado por (Z_2, +). Novamente temos um situação similar.

$$\begin{cases} (Z,+) \rightarrow E \\ = Int(E) \quad homomofismo \\ Z = Int(O) \rightarrow (map)O \end{cases} \subseteq (U,+) \rightarrow (Z_2, +)$$

O conjunto (U,+) também pode ser reconstruído e através de duas soluções de reconstrução. Estas soluções são (Z_2 × Z, +) e (Z,+). Eles não são equivalentes como grupos aditivos. Em matemática pura, este fato pode ser expresso pela extensão functor, denominada, EXT (Z, Z_2) ≠ 0 em álgebra homológica.

2.6 Considerações Finais

A computação granular constitui-se de duas operações básicas. Em um primeiro momento a operação é a construção dos espaços de aproximação. Tais espaços desempenham um papel importante, em mineração de dados e sistemas de computação granular, na extração de padrões não triviais, desconhecidos e potencialmente úteis,

escondidos em grandes e incertos conjuntos de dados. Espaços de aproximação típicos incluem fuzzy sets, rough sets, near sets e sistemas de vizinhança. Este capítulo discutiu e comparou suas principais características e aplicações. Em um segundo momento, a operação é a de integração, ou seja a capacidade do processo de, a partir dos grânulos, reconstruir os dados na sua forma original. Este capítulo também mostrou que esta reconstrução também é matematicamente possível a partir de mais de uma abordagem. O capítulo seguinte apresenta os fundamentos, conceitos e técnicas necessários à compreensão do método de agrupamento espectral adotado neste trabalho como o método construtor dos grânulos de palavras.

3. AGRUPAMENTO ESPECTRAL

Nos últimos anos, o agrupamento espectral tornou-se um método de agrupamento bastante popular. É simples de implementar, pode ser resolvido de forma eficiente por softwares padrão de álgebra linear, e muitas vezes supera os algoritmos de agrupamento tradicionais, como o algoritmo k-means.

De modo geral, a partir de um conjunto de dados, o método procura resolver o problema de dividi-los em grupos, a partir das informações fornecidas pelo espectro da matriz associada ao grafo que representa as relações existentes entre estes dados.

Métodos espectrais têm sido utilizados com sucesso em uma grande variedade de aplicações nas mais variadas áreas do conhecimento (HAGEN, 1992) (CHUNG, 1997) (SHI & MALIK, 2000) (DING, 2001) (JORDAN & WEISS, 2001) (INDERJIT, 2004) (WHYTE & SMYTH, 2005) (SPIELMAN, 2007) (VON LUXBURG, 2007) (FILIPPONE, 2008) (HAMAD, 2008).

Este capítulo destina-se a apresentação dos fundamentos, conceitos e técnicas necessários à compreensão do método.

3. 1 Fundamentos de Teoria dos Grafos

Grafos são estruturas abstratas compostas por um par de conjuntos representando seus vértices e suas arestas. Um vértice representa a unidade fundamental da formação dos grafos. Uma aresta é um segmento que conecta dois vértices em um grafo. Um grafo pode ser expresso por $= (V,E)$, sendo $V = \{v_1, v_2, ..., v_n\}$ seu conjunto de vértices, com cardinalidade $|V| = V$, que representa sua ordem, e $E = \{e_1, e_2, ..., e_n\}$ seu conjunto de arestas, com cardinalidade $|E| = E$, que define o seu tamanho. Em um grafo não direcionado, cada aresta é um par não ordenado de vértices tal que $\{v_i, v_j\} = \{v_j, v_i\}$. Em um grafo direcionado, por sua vez, cada aresta é um par ordenado de vértices onde $\{v_i, v_j\} \neq \{v_j, v_i\}$. Os vértices v_i e v_j, chamados limites ou pontos finais da aresta, são ditos vizinhos. Em um grafo ponderado, a função peso $w: E \rightarrow \mathbb{R}$ é definida tal que um peso é atribuído para cada aresta do grafo. O que não acontece em grafos não ponderados, onde as arestas não possuem qualquer tipo de quantificação. O número de arestas incidentes em um dado v_i é o seu grau, denotado por k_i, ou seja, o grau de $v_i \in V$ é definido como

$$d_i = \sum_{j=1}^{n} w_{ij} \quad (3.1)$$

Observe que, de fato, esta soma só funciona sobre todos os vértices adjacentes a v_i, como para todos os outros vértices v_j, se o peso w_{ij} é igual a 0. Dado um subconjunto de vértices $A \subset V$, o seu complemento $V \backslash A$ é denotado por $\bar{A}$. O vetor $\mathbb{1}_A = (f_1, ..., f_n)' \in \mathbb{R}^n$ como um vetor com entradas $f_i = 1$ se $v_i \in A$ e $f_i = 0$, caso contrário. Para dois conjuntos, não necessariamente disjuntos, $A, B \subset V$ define-se

$$W(A,B) = \sum_{i \in A, j \in B} w_{ij} \quad (3.2)$$

Podemos considerar representações para a medida do tamanho de um subconjunto $A \subset V$:

$|A|$ = o número de vértices em A

$vol(A) = \sum_{i \in A} d_i$

Intuitivamente, $|A|$ mede o tamanho de A por seu número de vértices, enquanto vol (A) mede seu tamanho através da soma dos pesos de todas as arestas ligadas aos vértices em A. O subconjunto $A \subset V$ de um grafo é conectado se quaisquer dois vértices em A podem ser conectados por um caminho tal que os pontos intermediários também estão em A. Um subconjunto de A é chamado um componente conectado se ele é conectado e se não existem conexões entre os vértices A e $\bar{A}$. Os conjuntos não vazios $A_1, ..., A_k$ formam uma partição do grafo se $A_i \cap A_j = \emptyset$ e $A_1 \cup ... \cup A_k = V$. Um grafo é dito regular se todos os seus vértices possuem o mesmo grau. Se um grafo regular possui arestas ligando um vértice a todos os outros, é dito completo.

3. 2 Grafos de Similaridade

Dado um conjunto de pontos $x_i, ..., x_n$ e uma medida de similaridade $s_{i,j} \geq 0$ entre todos os pontos de dados, o objetivo do agrupamento é dividir os pontos de dados em grupos, de forma que os pontos que pertencem ao mesmo grupo sejam similares e os pontos em diferentes grupos sejam diferentes uns dos outros. Se toda a informação que temos sobre estes pontos de dados é a similaridade que estes pontos mantêm entre si,

uma forma simples de representá-los é através de um grafo de similaridade G = (V,E). Cada vértice v_i no grafo representa um dado x_i. Dois vértices estão conectados se a similaridade s_{ij} entre os dados x_i e x_j é positiva e maior que um determinado limite. O vértice tem, então, o peso $s_{i,j}$. Sendo assim, o problema de agrupamento pode ser reformulado com base no grafo de similaridade. O objetivo passa a ser encontrar uma partição do grafo cujas as arestas entre diferentes grupos tenha pesos muito baixos (o que diferencia os grupos) e as arestas dentro dos elementos do grupo tenha pesos altos (o que representa a similaridade entre os elementos do grupo).

Existem várias formas de se associar pesos as arestas para a representação da similaridade.

Grafo k-vizinhança. O objetivo é conectar o vértice v_i com o vértice v_j se v_j está entre os k vizinhos mais próximos de v_i. Esta forma resulta em um grafo orientado pois a relação de vizinhança não é simétrica, como ilustra a figura:

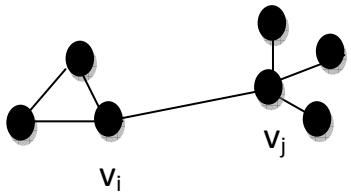
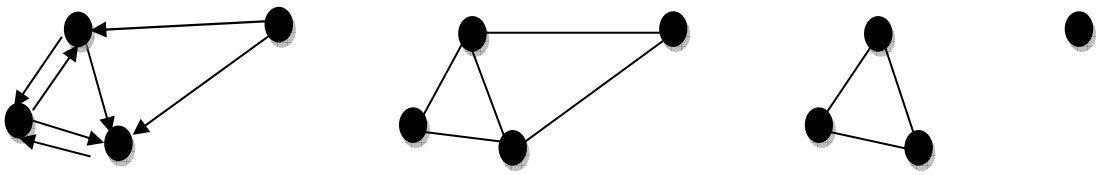


Figura 7. Exemplo da não simetria da relação de vizinhança. O vértice v_j é um dos 3 vizinhos mais próximos de v_i, mas não o contrário.

Existem duas formas de tornar o grafo não direcionado. A primeira é simplesmente ignorar a direção das arestas e conectar v_i e v_j através de uma aresta não direcionada se v_i está entre os k vizinhos mais próximos de v_j ou se v_j está entre os k vizinhos mais próximos de v_i. O grafo resultante é chamado grafo dos k vizinhos mais próximos. A segunda, é conectar os vértices v_i e v_j se tanto v_i está entre os k vizinhos mais próximos de v_j quanto v_j está entre os k vizinhos mais próximos de v_i. O grafo resultante é chamado grafo dos k-vizinhos mútuos mais próximos. Em ambos os casos, após a conexão apropriada dos vértices, as arestas são ponderadas pela similaridade das suas extremidades.



(a) (b) (c)

Figura 8. Exemplos de (a) k-vizinhança direcionado, (b) k-vizinhança simétrica e (c) k –vizinhança mútua.

Grafo completamente conectado. Neste caso, todos os pontos com similaridade positiva são conectados uns aos outros e o peso de todas as arestas são identificadas por $s_{i,j}$. Como o grafo deve representar as relações da vizinhança local, esta construção é útil apenas se a própria função de similaridade modela esta vizinhança.

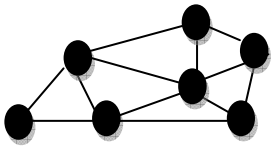


Figura 9. Exemplo de grafo completamente conectado.

Grafo ε-vizinhança. Todos os pares de pontos cuja distância é menor que ε são conectados. Como as distâncias entre todos os pontos estão aproximadamente na mesma escala (no máximo ε), os pesos das arestas podem não incorporar mais informação sobre os dados. Por esta razão, os grafos ε-vizinhança são geralmente considerados como grafos não ponderados.

Todos estes grafos de similaridade são regularmente usados para o agrupamento espectral e, segundo (VON LUXBURG, 2007) não existem resultados teóricos que demonstrem a influência da escolha do grafo nos resultados do agrupamento.

3.3 Representação de Grafos

A teoria espectral dos grafos, que fundamenta os algoritmos de agrupamento espectral, é um ramo da matemática discreta e da álgebra linear que estuda as propriedades de um grafo a partir das informações fornecidas pelo espectro da matriz associada a este grafo. As matrizes mais comuns são a matriz de adjacência e a matriz Laplaciana (HOGBEN, 2009).

3.3.1 Matriz de Adjacência

A representação mais comum de um grafo é através da sua matriz de adjacências que é uma matriz quadrada da ordem do grafo. Os elementos fora da diagonal principal representam o número de arestas do vértice v_i ao vértice v_j e os elementos da sua diagonal principal são duas vezes o número de arestas que tem por pontos finais os mesmos vértices $\{v_i, v_i\}$.

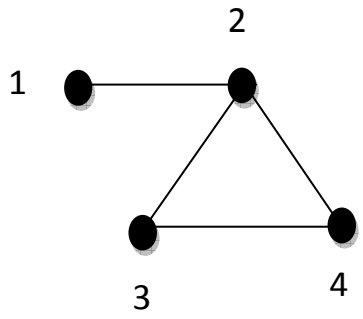


Figura 10. Grafo G_1 simples não ponderado

$$A(G_1) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Figura 11. Matriz de Adjacências do grafo da Figura 10

No caso de grafos simples e não ponderados, tem-se a matriz de adjacências $A \in (0,1)^{VxV}$, definida da seguinte forma:

$$A_{i,j} = \begin{cases} 1 & \text{se } (v_i, v_j) \in E \\ 0 & \text{caso contrário} \end{cases} \quad (3.3)$$

No caso de grafos ponderados, a matriz de adjacências $W \in \mathbb{R}^{VxV}$ é definida como:

$$W_{i,j} = \begin{cases} w(v_i, v_j) & \text{se } (v_i, v_j) \in E \\ 0 & \text{caso contrário} \end{cases} \quad (3.4)$$

Onde $w(v_i, v_j)$ é uma função w: e → $\mathbb{R}$ que pondera as arestas do grafo.

O polinômio característico da matriz de adjacência A(G) de um grafo G é chamado polinômio característico de G e denotado por $p_G(\lambda)$. Assim, $p_G(\lambda) = det(\lambda I - A(G))$, onde λ é uma raiz deste polinômio e dito ser um autovalor de G. Como o grafo tem n vértices, ele possui n autovalores, sendo o maior deles o raio espectral de G, denominado índice do grafo.

O espectro de G, indicado por spect(G), é definido como uma matriz 2 × d, tendo na primeira linha os d autovalores distintos de G dispostos em ordem decrescente e na segunda linha as suas respectivas multiplicidades algébricas. Como a matriz de adjacência de G é simétrica, todos os seus autovalores são reais.

Como exemplo, considere o grafo G1 da Figura 10. O seu polinômio característico é $p_{G1} = \lambda^4 - 4\lambda^2 - 2\lambda + 1$, tendo como espectro

$$\text{Spect(G1)} = \begin{bmatrix} 2{,}1701 & 0{,}3111 & -1 & -1{,}4812 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

3.3.2 Matriz Laplaciana

Algumas das principais ferramentas para o agrupamento espectral são as matrizes de grafos Laplacianos ou matrizes Laplacianas. Considere G um grafo

ponderado não direcionado com uma matriz de pesos W, onde $w_{i,j} = w_{j,i} \geq 0$. Quando usamos os autovetores de uma matriz não assumimos, necessariamente, que estes estão normalizados. Autovalores são sempre ordenados de forma decrescente em relação a sua multiplicidade. Por "os primeiros k autovetores" vamos assumir os k menores autovetores.

A matriz Laplaciana do grafo G é definida como:

$$L(G) = D(G) - A(G)$$

Onde D(G) é a matriz diagonal composta pelos graus dos vértices de G e A(G) é a matriz de adjacência de G. Vejamos o grafo G2 da Figura 12.

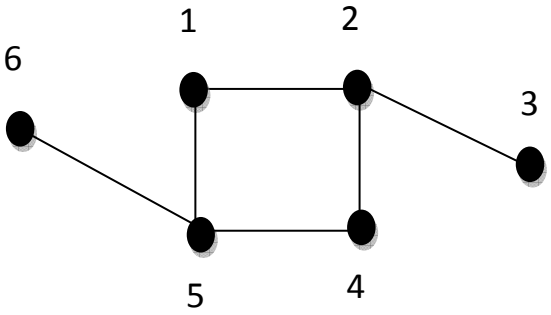


Figura 12. Grafo G2

A matriz diagonal dos graus dos vértices de G2 e sua matriz adjacência são, respectivamente:

$$D(G2) = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

e

$$A(G2) = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$$

Logo, a matriz Laplaciana de G2 é:

$$A(G2) = \begin{bmatrix} 2 & -1 & 0 & 0 & -1 & 0 \\ -1 & 3 & -1 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 2 & -1 & 0 \\ -1 & 0 & 0 & -1 & 3 & -1 \\ 0 & 0 & 0 & 0 & -1 & -1 \end{bmatrix}$$

Assim como na matriz de adjacência, é possível encontrar o espectro da matriz Laplaciana de um grafo, chamado espectro do Laplaciano.

O espectro do Laplaciano de G, denotado por ζ(G), é uma matriz 1 × n na qual as entradas são os autovalores de L(G) ordenados de maneira não crescente. Então,

$$\zeta(G) = (\mu_1, \mu_2, ..., \mu_n)$$

onde $\mu_1 \geq \mu_2 \geq .... \geq \mu_{n-1} \geq \mu_n$ são os autovalores de L(G). O maior autovalor, μ_1, é chamado índice do Laplaciano de G.

Assim para o grafo G2 da Figura 12, temos o espectro Laplaciano:

$$\zeta(G5) = (4{,}7321, 3{,}4142, 2, 1{,}2679, 0{,}5858, 0)$$

3.3.2.1 Matriz Laplaciana não Normalizada

Uma matriz Laplaciana não normalizada é definida como:

$$L = D - W$$

Onde D é a matriz diagonal composta pelos graus dos vértices de G e W uma matriz de pesos W, onde $w_{i,j} = w_{j,i} \geq 0$.

Tal matriz apresenta uma série de propriedades importantes e uma boa visão geral sobre estas propriedades pode ser encontrada em (MOHAR, 1992) e (MOHAR, 1997). As proposições que seguem resumem as propriedades mais importantes quando o objetivo é o agrupamento espectral.

Proposição 1 (Propriedades de L) A matriz L satisfaz as seguintes propriedades:

1. Para todo vetor $f \in \mathbb{R}^n$ temos que $f' L f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij}(f_i - f_j)^2$.
2. L é simétrica e positiva semi-definida. Ou seja, em muitos aspectos a matriz é análoga a um número real positivo.
3. O menor autovalor de L é 0 e o autovetor correspondente é composto por constantes de valor 1.
4. L tem n não negativo e autovalores reais.

A prova para cada uma destas propriedades pode ser encontrada em (VON LUXBURG, 2007).

A matriz Laplaciana não normalizada não depende dos elementos da diagonal da matriz de adjacências W. Cada matriz de adjacência que coincide com W em todas as posições fora da diagonal levam ao mesmo grafo Laplaciano não normalizado.

A matriz Laplaciana e seus autovalores e autovetores podem ser usados para descrever muitas das propriedades se seus grafos, vide (MOHAR, 1992) e (MOHAR, 1997).

Proposição 2 (Número de componentes conectadas e o espectro de L) Considere G um grafo não direcionado com pesos não negativos. A multiplicidade k do autovalore 0 de L é igual a número de componentes conectados A_1, A_2, ..., A_k no grafo. O autoespaço de autovalor 0 é medido pelos vetores compostos por 1 destes componentes. A prova pode ser encontrada em (VON LUXBURG,2007).

Como os pesos $w_{i,j}$ são não negativos, sua soma só pode desaparecer se todos os termos $w_{i,j}(f_i - f_j)^2$ desaparecerem. Então, se dois vértices v_i e v_j estão conectados (isto é, se $w_{i,j} > 0$), então f_i ser igual a f_j. Com este argumento, podemos perceber que f precisa ser constante para todos os vértices que podem ser conectados por um caminho no grafo.Além disso, como todos os vértices de um componente conectado em um grafo não direcionado podem ser conectados por um caminho, f precisa ser constante sobre todo o componente conectado. Em uma grafo que consiste de apenas um componente

conectado nós temos autovetores formados por constantes 1 com autovalores 0, o que é indicativo de um componente conectado.

Agora, considere o caso de k componentes conectados. Sem perda de generalidade, podemos assumir que os vértices são ordenados de acordo com os componentes conectados aos quais eles pertencem. Neste caso, a matriz de adjacências W tem um bloco na forma diagonal e o mesmo é verdadeiro para a matriz L:

$$L = \begin{bmatrix} L_1 & & & \\ & L_2 & & \\ & & \ddots & \\ & & & L_k \end{bmatrix}$$

Note que cada um dos blocos L_i é um grafo Laplaciano sobre ele próprio, correspondendo a um subgrafo Laplaciano. É este o caso em todas as matrizes de bloco diagonal. Sabemos que o espectro de L é dado pela união dos espectros de Li, e o autovetores correspondentes de L são os autovetores de Li, preenchidos com 0 nas posições dos outros blocos. Como cada Li é um grafo Laplaciano de um grafo conexo, sabemos que cada Li tem autovalor 0 com multiplicidade 1, e o vector próprio correspondente é o vector de uma constante sobre o i-ésimo componente ligado. Assim, a matriz L tem tantos autovalores 0 quantas são os componentes conectados, e os autovetores correspondentes são os vetores indicadores dos componentes conectados.

3.3.2.2 Matriz Laplaciana Normalizada

Existem, na literatura, duas matrizes denominadas como matrizes Laplacianas normalizadas. As duas estão fortemente relacionadas e são definidas como:

$$L_{sym}: D^{-1/2} LD^{-1/2} = I - D^{-1/2} WD^{-1/2} \quad (3.5)$$

$$L_{rw}: D^{-1} L = I - D^{-1} W \quad (3.6)$$

A primeira matriz é denotada por L_{sym} por se tratar de uma matriz simétrica e a segunda por L_{rw} por estar fortemente relacionada a um caminho randômico ou aleatório.

A seguir estão relacionadas algumas das propriedades fundamentais deste tipo de matriz. A principal referência para o assunto é (CHUNG, 1997).

Proposição 3 (Propriedades de L_{sym} e L_{rw}) As matrizes Laplacianas normalizadas satisfazem às seguintes propriedades:

1. Para todo vetor $f \in \mathbb{R}^n$ temos que $f' L_{sym} f = \frac{1}{2} \sum_{i,j=1}^{n} w_{ij} (\frac{f_i}{\sqrt{d_1}} - \frac{f_j}{\sqrt{d_j}})^2$.
2. λ é um autovalor de L_{rw} com autovetor u se e somente se λ é um autovalor de L_{sym} com autovetor $w = D^{1/2}u$.
3. λ é um autovalor de L_{rw} com autovetor u se e somente se λ e u o problema do valor generalizado $Lu = \lambda Du$.
4. 0 é um autovalor de L_{rw} com um vetor de constantes 1 como autovetor. O é auto valor de L_{sym} com autovetor $D^{1/2}$ **1**.
5. L_{sym} e L_{rw} são positivas e semi definidas. E têm n autovetores reais não negativos $0 = \lambda_1 \leq \ldots \leq \lambda_n$.

A prova destas propriedades encontra-se em (VON LUXBURG, 2007). Como no caso da matriz Laplaciana não normalizada, a multiplicidade do autovalor 0 da matriz Laplaciana normalizada está relacionada aos componentes concetados.

Proposição 4 (Número de componentes conectadas e o espectro de L_{sym} e L_{rw}) Suponha G um grafo não direcionado com pesos não negativos. A multiplicidade k do autovalor 0 é, tanto para L_{sym} quanto para L_{rw} igual ao número de componentes conectadas A_1, ..., A_k no grafo. Para L_{rw} o autoespaço de 0 é medido por vetores completo de 1s destas componentes. Para L_{sym} o autoespaço de 0 é medido por vetores $D^{1/2}$ 1^a_i.

A prova para esta proposição também pode ser encontrada em (VON LUXBURG, 2007).

3.4 Particionamento de Grafos

No contexto da teoria dos grafos, a tarefa de agrupamento é uma tarefa de particionamento de grafos (SCHAEFFER, 2007). O objetivo é particionar o grafo em grupos distintos de vértices, com base nas suas similaridades e de acordo com a estrutura de suas arestas. Cada grupo deve conter um número grande de arestas e poucas arestas devem conectar estes grupos.

Dado um grafo G = (V,E) e um número k de grupos que se deseja obter, o objetivo do particionamento é gerar k sub-grafos $G'_i = (V'_i, E'_i)$, i = 1,2, ..., k, tal que $V'_i \subseteq V$, $\bigcup_i^k V'_i = V$ e $\bigcap_i^k V'_i = \emptyset$. Embora nem todos os grafos possuam uma estrutura com grupos naturais, ou seja, um número maior de vértices ou um peso maior nas arestas internas aos grupos do que entre eles, sempre será possível partcionar um grafo em grupos disjuntos. Se a estrutura do grafo é completamente uniforme ou aleatória, o resultado será um particionamento arbitrário.

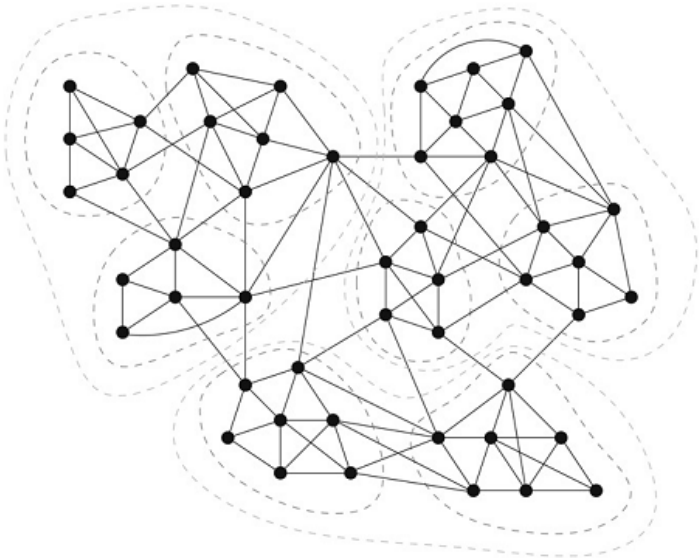


Figura 13. Agrupamento como uma tarefa de particionamento de grafos.

Existem duas abordagens distintas para o problema do particionamento de grafos: multi-particionamento ou bi-particionamento. A abordagem do multi-particionamento divide o conjunto de vértices V do G = (V,E) em k partições distintas $P_1, P_2, ..., P_n$, tal que $|P_i| = p$, i=1, 2, ..., k e $V = p * k$, satisfazendo $P_1 \cup P_2 \cup ... \cup P_k = V$ e $P_1 \cap P_2 \cap ... \cap P_k = \emptyset$. A abordagem do bi-particionamento divide o conjunto de vértices V do G = (V,E) em apenas 2 partições distintas P_1 e P_2 , satisfazendo $P_1 \cup P_2 =$

V e $P_1 \cap P_2 = \emptyset$. Podemos usar a abordagem do bi-particionamenro para particionar grafos em k partições distintas, aplicando-a de forma recursiva.

3.4 .1 Avaliação do Particionamento

Existem várias medidas de avaliação da qualidade do particionamento de um grafo. Estas medidas podem ser usadas na identificação de partições, na escolha de um melhor esquema de particionamento ou na comparação entre diferentes abordagens de particionamento.

3.4.1.1 Corte Mínimo

Uma das funções mais conhecidas para otimização do particionamento de grafos é a do Corte Mínimo (*Minimum Cut*) (CHUNG, 1997). Ou seja, dado um grafo de similaridades com a sua matriz de adjacência W, o caminho mais simples e mais direto para a sua partição é a solução do problema do corte mínimo. O objetivo é definir grupos de vértices cuja quantidade de arestas que os conectam seja mínima, ou seja, para uma dada quantidade k de subconjuntos, a abordagem do corte mínimo consiste na escolha de partições $A_1, ..., A_k$ que minimiza

$$\text{cut}(A_1, ..., A_k) = \frac{1}{2}\sum_{i=1}^{k}(A_i, \bar{A}_i) \qquad (3.7)$$

Para o caso particular de k = 2, a abordagem pelo corte mínimo é relativamente de fácil solução e pode ser resolvido de forma eficiente (STOER & WAGNER, 1997). Entretanto, na prática, não cria partições satisfatórias. Em muitos casos, a solução do corte mínimo simplesmente separa um vértice individual do resto de grafo. Um caminho para contornar este problema é definir explicitamente que $A_1, ..., A_k$ tenham um tamanho considerável. As funções que codificam esta possibilidade de definição do tamanho das partições são o corte médio – *RatioCut* – (HAGEN & KAHNG, 1992), o corte normalizado – *Ncut* – (SHI & MALIK, 2000) e corte Mínimo-Máximo – *Minimum-Maximum Cut* – (DING, 2001).

3.4.1.2 Corte Médio

A abordagem pelo corte médio define o tamanho das partições através da sua quantidade de vértices $|A_i|$:

$$\text{RatioCut}\ (A_1, ..., A_k) = \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \bar{A}_i)}{|A_i|} = \sum_{i=1}^{k} \frac{cut(A_i, \bar{A}_i)}{|A_i|} \quad (3.8)$$

O objetivo é gerar partições cuja quantidade de arestas intergrupos seja normalizada pelo tamanho de $|A_i|$. Esta abordagem, apesar melhorar a qualidade das partições geradas, não considera um aspecto essencial para a análise de agrupamento: as conexões intragrupos. Um método realmente robusto deve considerar tanto as conexões intergrupos quanto as intragrupo. È isto que ocorre com os dois métodos que seguem.

3.4.1.3 Corte Normalizado

A abordagem pelo corte normalizado define o tamanho das partições através dos pesos de suas arestas, impondo uma restrição ao volume, vol(A), destas partições.

$$\text{NCut}\ (A_1, ..., A_k) = \frac{1}{2} \sum_{i=1}^{k} \frac{W(A_i, \bar{A}_i)}{vol(A_i)} = \sum_{i=1}^{k} \frac{cut(A_i, \bar{A}_i)}{vol(A_i)} \quad (3.9)$$

Enquanto a restrição imposta por Ratiocut atua sobre a quantidade de vértices presentes nas partições, a restrição Ncut utiliza a quantidade de arestas, $vol(A_i)$, para balancear as partições. Para que as conexões intergrupos sejam consideradas, o objetivo é minimizar o valor de $vol(A_i)$ enquanto maximiza $cut(A_i, \bar{A}_i)$. O fator de normalização representa os graus dos vértices internos a partição.

3.4.1.4 Corte Mínimo-Máximo

A abordagem pelo corte Mínimo-Máximo busca minimizar as similaridades intergrupos e maximizar as similaridades intragrupos simultanemente.

$$\text{MinMaxCut}\ (A_1, ..., A_k) = \sum_{i=1}^{k} \frac{cut(A_i, \bar{A}_i)}{cut(A_i, A_i)} \quad (3.10)$$

De acordo com (DING, 2004), se as partições são bem pronunciadas, as 3 abordagens conseguem resultados similares e precisos. Se as partições são marginalmente separados apenas os cortes normalizado e mínimo-máximo conseguem bons resultados. No caso onde há significantes sobreposição entre as partições, apenas o corte mínimo-máximo apresenta partições balanceadas e compactas.

O problema de minimizar estas funções pertence a categoria de problemas NP-Complexos, dada a sua natureza combinatória (SHI & MALIK, 2000). Uma alternativa viável para a sua solução é a abordagem espectral, pois representa um relaxamento de suas restrições. Através da abordagem espectral, o problema de particionamento é convertido em um problema de autovalor e a solução aproximada e dada através da utilização de ferramentas de álgebra linear (GOLUB & VAN LOAN, 1996).

3.5 Algoritmos

Os algoritmos espectrais não fazem suposições acerca da estrutura dos grupos. Ao invés disso, evidências locais da probabilidade de dois objetos pertencerem ao mesmo grupo são coletadas e uma decisão global é tomada para particionar todos os dados em grupos disjuntos, segundo algum critério. Geralmente, tais critérios são interpretados com um arcabouço que preserva os relacionamentos entre os dados, tanto quanto possível, em uma representação de menor dimensionalidade.

O que torna os métodos de agrupamento espectrais atrativos é o fato de que o seu ótimo global, em um domínio contínuo relaxado, é obtido através de autodecomposição. Ou seja, através da solução de um problema de autovalor. Obter uma solução discreta a partir de autovetores requer, geralmente, a solução de um outro problema de agrupamento, só que em um espaço de menor dimensionalidade. Desta forma, os autovetores são tratados como coordenadas geométricas do conjunto de dados. E vários métodos de agrupamento podem ser aplicados sobre esta representação, como, por exemplo, o algoritmo k-means (MACQUEEN, 1967).

Os algoritmos de agrupamento espectral são, em geral, constituídos de três etapas: pré-processamento, representação espectral e agrupamento (HAMAD, 2008). A etapa de pré-processamento é responsável por construir a matriz de similaridades e pela geração de seu respectivo grafo. Nesta etapa podem ser criados qualquer um dos grafos apresentados na seção 3.2. Na etapa de representação espectral, é gerada a matriz

Laplaciana associada ao grafo e são calculados seus autovalores e autovetores. Um método bastante empregado é o de Lanczos (GOLUB & VAN LOAN, 2008). Em seguida, os elementos do conjunto de dados são mapeados uma representação de menor dimensionalidade, baseada em um ou mais autovetores. Por fim, a etapa de agrupamento realiza a divisão dos elementos em grupos distintos.

Existem diversos argumentos a favor da utilização da versão normalizada da matriz Laplaciana ao invés da sua versão não normalizada. Um destes argumentos é a sua capacidade de implementar a minimização da similaridade intergrupos quanto a sua maximização intragrupos (VON LUXBURG, 2007). De qualquer forma estão apresentados as abordagens mais comumente usadas, baseadas tanto na versão normalizada quanto na versão não normalizada da matriz Laplaciana.

Algoritmo 3.1 – Agrupamento Espectral não normalizado

Entrada: Uma matriz de similaridade $S \in \mathbb{R}^{nxn}$, quantidade k de grupos a construir.

1. Construir um grafo de similaridade de acordo com um daqueles descritos na seção 3.2. Considere W como sua matriz de adjacência ponderada.
2. Calcular a matriz Laplaciana não normalizada.
3. Calcular os primeiros k autovetores u_i, ..., u_k de L.
4. Considere $U \in \mathbb{R}^{nxk}$ como a matriz que contém os vetores u_i, ..., u_k como colunas.
5. Para i=1, ..., n, considere $y_i \in \mathbb{R}^k$ como o vetor correspondente a i-ésima linha de U.
6. Agrupar os pontos $(y_i)_{i=1,...,n}$ em $\mathbb{R}^k$ com o uso do algoritmo k-means nos grupos C_1, ..., C_k.

Saída: Grupos A_1, ..., A_k com $A_i = \{j | y_j \in C_i\}$

Existem 2 versões diferentes para o agrupamento espectral normalizado, dependendo da matriz Laplaciana utilizada.

Algoritmo 3.2 – Agrupamento Espectral Normalizado 1 (SHI & MALIK, 2000)

Entrada: Uma matriz de similaridade $S \in \mathbb{R}^{nxn}$, quantidade k de grupos a construir.

1. Construir um grafo de similaridade de acordo como um daqueles descritos na seção 3.2. Considere W como sua matriz de adjacência ponderada.
2. Calcular a matriz Laplaciana não normalizada L.
3. Calcular os primeiros k autovetores do problema de autovetor generalizado $L_u = \lambda D_u$.
4. Considere $U \in \mathbb{R}^{nxk}$ como a matriz que contém os vetores u_i, ..., u_k como colunas.
5. Para i=1, ..., n, considere $y_i \in \mathbb{R}^k$ como o vetor correspondente a i-ésima linha de U.
6. Agrupar os pontos $(y_i)_{i=1,...,n}$ em $\mathbb{R}^k$ com o uso do algoritmo k-means nos grupos C_1, ..., C_k.

Saída: Grupos A_1, ..., A_k com $A_i = \{j | y_j \in C_i\}$

Este algoritmo usa os vetores generalizados da matriz Laplaciana L. Tais vetores correspondem aos autovetores da matriz L_{rw}, de acordo com a Proposição 3. De fato, o algoritmo trabalha com autovetores de matriz Laplaciana normalizada e por esta razão é chamado agrupamento espectral normalizado. O próximo algoritmo também usa a matriz Laplaciana, mas a matriz L_{sym} ao invés de L_{rw}. Este algoritmo precisa introduzir um passo adicional de normalização de colunas que não é necessário em outros algoritmos.

Algoritmo 3.3 – Agrupamento Espectral Normalizado 2 (JORDAN & WEISS, 2000)

Entrada: Uma matriz de similaridade $S \in \mathbb{R}^{nxn}$, quantidade k de grupos a construir.

1. Construir um grafo de similaridade de acordo como um daqueles descritos na seção 3.2. Considere W como sua matriz de adjacência ponderada.
2. Calcular a matriz Laplaciana normalizada L_{sym}..

3. Calcular os primeiros k autovetores u_1, ..., u_k de $L_{sym.}$
4. Considere $U \in \mathbb{R}^{nxk}$ como a matriz que contém os vetores u_i, ..., u_k como colunas.
5. Construir a matriz $T \in \mathbb{R}^{nxk}$ de U pela normalização das linhas para norma 1, definido por $t_{i,j} = u_{i,j} / (\sum_k u_{ik}^2)^{1/2}$.
6. Para i=1, ..., n, considere $y_i \in \mathbb{R}^k$ como o vetor correspondente a i-ésima linha de T.
7. Agrupar os pontos $(y_i)_{i=1,...,n}$ em $\mathbb{R}^k$ com o uso do algoritmo k-means nos grupos C_1, ..., C_k.

Saída: Grupos A_1, ..., A_k com $A_i = \{j | y_j \in C_i\}$

Os três algoritmos parecem bastante similares, exceto pelo fato de que usam três diferentes grafos Laplacianos. Nos três algoritmos, o artifício principal é mudar a representação abstrata dos pontos de dados x_i para os pontos $y_i \in \mathbb{R}^k$. Esta troca é útil devido as propriedades dos grafos Laplacianos. Esta troca de representação aumenta as propriedades de agrupamento nos dados, de forma que os grupos podem ser detectados facilmente na nova representação (VON LUXBURG, 2007).

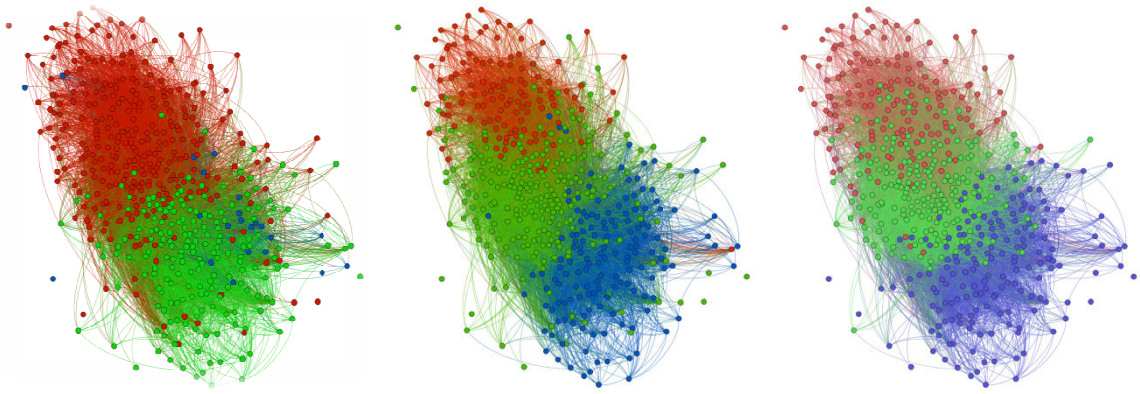


Figura 14. Exemplos de agrupamentos formados a partir da aplicação de diferentes algoritmos de agrupamento espectral.

3.6 Considerações Finais

Métodos de agrupamento espectral mostram-se interessantes por sua eficiência e simplicidade de implementação. Além disso, têm se mostrado bastante útil as mais

variadas áreas de aplicação. Este capítulo apresentou os fundamentos, conceitos e técnicas necessários à sua compreensão. O próximo capítulo descreve os conceitos chave e as etapas necessárias a construção dos grânulos de palavras. Também apresenta os resultados alcançados e compara tais resultados a duas técnicas com objetivos semelhantes.

4 GRÂNULOS DE PALAVRAS

O processo de granulação se baseia na agregação de objetos indiscerníveis. A indiscernibilidade entre estes objetos pode ser tratada por uma função de similaridade. Este trabalho apresenta a construção de grânulos cujos objetos são as palavras contidas em coleções de documentos. Uma abordagem fuzzy é utilizada na avaliação da similaridade entre estes objetos e a agregação se dá através da adoção de algoritmos de agrupamento espectral, capazes de computar tal agregação com base nesta similaridade. Este capítulo descreve os conceitos chave e as etapas necessárias a construção destes grânulos. Os resultados alcançados são apresentados e comparados a duas técnicas bem conhecidas e que possuem objetivos semelhantes.

4.1 Similaridade entre Palavras

Existem, na literatura, dois tipos principais de similaridade entre palavras (KOZIMA, 1993) (RAPP, 2002):

- Paradigmática (ou de substituição): duas palavras são paradigmaticamente similares se podem se substituir em um contexto particular. Por exemplo, na frase e no contexto de "Eu comprei um carro", a palavra carro pode ser substituída por automóvel sem nenhuma perda semântica.

- Sintagmática: duas palavras são sintagmaticamente similares se ocorrem de maneira significativa em um mesmo contexto. Por exemplo, as palavras "carro" e "trânsito" são sintagmaticamente similares, pois ocorrem tipicamente juntas dentro de determinados contextos.

Apesar de esta distinção ser relativamente rara na literatura, alguns trabalhos (RAPP, 2002) referem-se ao primeiro tipo como similaridade semântica ("semantic similarity") e ao segundo (WASHTELL, 2009) como semântica de relacionamento ou afinidade ("semantic relatedness"). Mesmo entendendo que estes dois aspectos estão fortemente relacionados, o foco deste trabalho está essencialmente neste segundo tipo.

Os dois tipos são computados através de diferentes métodos e utilizados em uma grande variedade de aplicações. Tipicamente, a similaridade semântica é computada tanto através dos relacionamentos taxionômicos, como hiperonímia (palavras que dão idéia de conjunto, por exemplo, animal ou vestimenta) e sinonímia (palavras de mesmo significado), quanto através de evidências de distribuição probabilísticas. No primeiro caso, as abordagens, geralmente, pressupõem uma estrutura independente que representa informações léxico-semânticas sobre as palavras, tais como WordNet®.

Wordnet® é um sistema de referência léxico (FELLBAUM, 1998), para a língua inglesa, criado por lingüistas e psicolinguistas da Universidade Princeton. O sistema se distingue dos sistemas convencionais pelo fato que a informação léxica é organizada de acordo com o significado das palavras e não de acordo com a sua forma. A unidade central do Wordnet® é chamada *synset* e é representada por conjuntos de palavras que contém o mesmo significado, isto é, sinônimos. Um *synset* representa um conceito referenciado por diferentes palavras. Por exemplo, o conjunto {*car, auto, automobile*, *machine, motorcar*} é um *synset* e forma uma unidade básica do léxico Wordnet®. As diferenças sutis existentes entre os significados dos sinônimos são ignoradas pelo sistema. Os *synsets* também registram definições curtas e registram relações semânticas entre os conjuntos de sinônimos.

A similaridade semântica é então, calculada com base na sobreposição dos seus níveis (contextos). A avaliação de similaridade semântica baseada em taxionomia depende crucialmente da inserção de palavras em classes semânticas mais gerais. Em contrapartida, a avaliação baseada em distribuição probabilística somente pode assumir que palavras com distribuições semelhantes, pertencem a mesma relação sintagmática (estrutural, hierárquica). Portanto, podem ser entendidas como semanticamente similares, embora este procedimento possa encobrir outras propriedades.

Métodos baseados em relacionamentos taxionômicos se baseiam em contagem de arestas e definem a similaridade entre duas palavras como uma função do tamanho do caminho de ligação entre estas palavras. O trabalho de Rada et al. (RADA, 1989) representa a proposta inicial dos métodos que se baseiam em contagem de arestas. A semântica por relacionamento ou afinidade é computada com base do número de arestas entre as palavras na taxionomia. Leacock e Chodorow (LEACOCK, 1998) consideram em sua medida o nível taxionômico no qual as palavras são encontradas: lch(c1,c2) = -log (lenght(c1,c2)/2D, onde lenght(c1,c2) é o número de nós do caminho mais curto entre as duas palavras. D é a quantidade máxima de níveis da taxionomia. A métrica de

similaridade descritas em Wu e (PALMER, 1994) considera os níveis de duas palavras na taxionomia juntamente com os níveis do menor subordinado comum (least commom subsume – LCS): sim wup = (2 * depth (LCS)/depths(words1)+(word2)).

Métodos baseados em distribuição probabilísticas, também conhecidos como métodos baseados no conteúdo ou métodos baseados em corpus (SALAHLI, 2009), medem a diferença de conteúdo de informação entre duas palavras em função da sua probabilidade de ocorrência no documento ou na coleção. O método, inicialmente proposto por (RESNIK, 1999) , considera que a similaridade entre duas palavras é igual ao conteúdo de informação (information content ou IC) do menor subordinado comum (LCS): simrez = IC (LCS(c1,c2). Como muitas palavras podem compartilhar o mesmo LCSr e, com isso apresentarem os mesmos valores de similaridade, a medida de Resnik pode não ser capaz de obter medidas finas de distinção. (JIANG, 1997) e (LI, 2003) desenvolveram medidas que avaliam o conteúdo de informação de um conceito subordinado pelo conteúdo de informação dos conceitos individuais. Lin utiliza razão e proporção, Jiang e Conrath utilizam a diferença. Métodos baseados em glossários definem a relação entre duas palavras com uma função de sobreposição de glossários. Banerjee e Pedersen (BANERJEE, 2002), propuseram um método que calcula o escore de sobreposição pela extensão dos glossários de forma a englobar hierarquicamente todos os documentos relacionados. Muitos destes conceitos também foram, inicialmente, desenvolvidos no contexto Wordnet®.

Alguns trabalhos definem a semântica de relacionamento entre duas palavras fazendo uso da Web. (BOLLEGARA, 2007) propõe um método que explora a contagem de páginas e trechos de texto retornados pelos mecanismos de busca da Web para medir a similaridade semântica entre as palavras. (CILIBRASI, 2007) desenvolveu um método que define a relação entre as palavras através da medida de similaridade Google. O trabalho utiliza a Web como banco de dados e o Google como mecanismo de busca. Uma abordagem que propõe uma medida de similaridade de relacionamento usando a Wikipedia é proposta por Gabrilovich em (GABRILOVICH, 2007). Strube e Ponzetto também investigam a utilização da Wikipedia no cálculo de medidas de similaridade de relacionamento (STRUBE, 2006) . (LI, 2003) determina a similaridade pela junção das informações estruturais contidas em uma taxionomia e informações do conteúdo do documento.

Apesar da criação de um agrupamento de palavras ser um modo direto para se encontrar palavras relacionadas, técnicas de agrupamento de documentos têm sido

efetivamente utilizadas com este objetivo. O agrupamento de palavras pode se dar a partir de um agrupamento prévio dos documentos onde os agrupamentos de palavras se baseiam nos agrupamentos de documentos (CROUCH, 1989) ou pela vinculação de cada tópico a um agrupamento de documentos (CROFT, 1980). A ideia básica por trás desta abordagem é que, tanto a similaridade entre os documentos quanto a similaridade entre as palavras, refletem conexões semânticas entre as palavras, apesar de fornecerem informações diferentes.

4.2 Similaridade Fuzzy entre Palavras

A maioria dos modelos de representação de documentos adota um esquema de pesagem de termos. Este peso expressa o grau de importância de cada termo como um descritor da informação contida no documento. A computação destes pesos se baseia na contagem das ocorrências de cada termo no documento. Neste caso, uma função F computa, para cada documento d e cada termo t, um valor numérico. Um exemplo de definição da função F é, como expresso na equação X, considerar o peso de cada termo t como proporcional a sua frequência no documento d e inversamente proporcional a frequência do termo na coleção de documentos.

$$F(d,t) = tf_{dt} \text{ X } IDF_t \qquad (4.1)$$

onde:

- tf_{dt} representa a frequência normalizada do termo que pode ser definida como $tf_{dt} = OCC_{dt}/MAXOCC_d$; onde OCC_{dt} representa a quantidade de ocorrências de t em d e $MAXOCC_d$ representa a quantidade de ocorrências do termo mais frequente em d;
- IDF_t representa um frequência inversa que pode ser definida como $IDF_t = \log (N/NDOC_t)$, onde N representa a quantidade de documentos na coleção e $NDOC_t$ a quantidade de documentos que contém t.

A definição da função F se baseia em uma análise quantitativa do texto que torna possível a modelagem qualitativa da importância de cada termo.

Os modelos fuzzy para a representação de documentos se baseiam em uma representação normalizada de tais pesos que permite interpretá-los como um conjunto fuzzy de termos (ZADROZNY, 2009). Do ponto de vista matemático, trata-se de uma extensão natural: o conceito de importância dos termos na descrição do conteúdo da informação pode ser naturalmente descrito pela adoção da função F (tal qual como obtida em (4.1), mas normalizada de forma a a produzir valores na faixa [0,1]) como uma função de pertinência ao conjunto fuzzy que representa o documento.

4.2.1 Relações Fuzzy

A teoria dos conjuntos fuzzy (ZADEH, 1965) trata da representação de conjuntos cujos limites não são bem definidos. A idéia básica é a associação de uma função de pertinência a cada um dos elementos destes conjuntos. Esta função produz valores no intervalo [0,1] com 0 correspondendo a nenhuma pertinência do elemento ao conjunto e 1 correspondendo a pertinência total. Valores entre 0 e 1 indicam a pertinência parcial destes elementos. Assim, a pertinência dos elementos a um conjunto fuzzy é definida de forma gradual ao invés de abrupta ou crisp como na teoria clássica.

Matematicamente, o conceito de relação é formalizado a partir da teoria dos conjuntos. Desta forma, intuitivamente pode-se dizer que a relação é fuzzy quando optamos pela teoria dos conjuntos fuzzy e, é crisp quando optamos pela teoria clássica para conceituar a relação em estudo. A principal consequência na opção pela relação fuzzy é que uma relação clássica indica se há ou não relação entre os dois elementos enquanto, uma relação fuzzy além de indicar se há ou não a relação, indica também o grau desta relação.

O conceito matemático de uma relação é formalizado utilizando-se o produto cartesiano clássico entre conjuntos como dado a seguir:

Definição 4.1 Uma relação R, sobre $U_1 \times U_2 \times ... \times U_n$ é qualquer subconjunto do produto cartesiano $U_1 \times U_2 \times ... \times U_n$. Se o produto cartesiano for formado apenas por dois conjuntos $U_1 \times U_2$ a relação é chamada binária sobre $U_1 \times U_2$. Se $U_1 = U_2 = ... = U_n = U$, diz-se que R é uma relação sobre U e, se o produto cartesiano for composto por dois conjuntos iguais, $U \times U$, R é chamada de relação binária sobre U.

Como a relação R é um subconjunto do produto cartesiano, pode ser representada por uma função característica C_R. Assim,

$$C_R(x_1, x_2, \ldots, x_n) = \begin{cases} 1 \; se \; (x_1, x_2, \ldots, x_n) \in R \\ 0 \; se \; (x_1, x_2, \ldots, x_n) \notin R \end{cases} \quad (4.2)$$

Definição 4.2 Uma relação fuzzy R, sobre U_1 X U_2 X ... x U_n é qualquer subconjunto fuzzy do produto cartesiano U_1 X U_2 X ... x U_n. Se o produto cartesiano for formado apenas por dois conjuntos U_1 X U_2 a relação é chamada fuzzy binária sobre U_1 X U_2. Se os conjuntos U_i forem todos iguais a U, R é uma relação fuzzy sobre U e, sobre U x U, R é chamada apenas relação fuzzy binária.

Se a função de pertinência da relação fuzzy R for também indicada por R, então o número R (x_1, x_2, ..., x_n) ∈ [0,1] indica o grau com que os elementos x_i que compõem a n-tupla (x_1, x_2, ..., x_n) estão relacionados segundo a relação R.

4.2.2 Relação de Similaridade Fuzzy

Formalmente, um documento é representado como um conjunto fuzzy de termos: $R_d = \sum_{t \in T} \frac{\mu_d(t)}{t}$, onde a função de pertinência é definida como μ_d = D × T → [0,1]. Neste caso, μ_d (t) = F (d,t). Neste contexto, podemos definir a relação de similaridade fuzzy entre os termos dos documentos de uma coleção como:

Definição 1. Uma relação fuzzy entre dois conjuntos finitos X = {x_1, ..., x_u} e Y = {y_1, ..., y_v} é formalmente definida como uma relação binária fuzzy f: X × Y →[0,1], onde u e v representam a quantidade de elementos em X e Y, respectivamente.

Definição 2. Dado um conjunto de termos, T = {t_1, ..., t_i} e um conjunto de documentos, D = {d_1, ..., d_j}, cada t_i é representado por um conjunto fuzzy h(t_i) de documentos; h(t_i) = {F(t_i,d_j) | ∀ d_j ∈ D}, onde F(t_i,d_j) e o grau de pertinência de t_i em d_j.

Definição 3. A relação fuzzy de similaridade entre dois termos se baseia na avaliação da coocorrência de t_i e t_j no conjunto D e pode ser definida como:

$$\mathrm{RT}\ (t_i,t_j) = \frac{\sum_k \min\ (F(t_i,d_k\ ),F(t_j,d_k)}{\sum_k \max\ (F(t_i,d_k\ ),F(t_j,d_k)} \tag{4.3}$$

Uma simplificação da relação fuzzy RT baseada na coocorrência de termos pode ser definida como:

$$r_{i,j} = \frac{n_{i,j}}{n_i + n_j - n_{i,j}} \tag{4.4}$$

onde

- $r_{i,j,}$ representa a relação fuzzy RT entre os termos i e j;
- $n_{i,j}$, representa a quantidade de documentos que contém os termos i e j;
- n_i, representa a quantidade de documentos que contém o termo i;
- n_j, representa a quantidade de documentos que contém o termo j;

4.3 Construção dos Grânulos de Palavras

A construção dos grânulos se baseia nas etapas básicas do processo de mineração de textos. Cada um dos textos de cada uma das coleções, apresentadas em detalhes nas seções 4.3.1.1 a 4.3.1.4, são submetidos a etapa de pré-processamento, incluindo limpeza e seleção de atributos. Após a determinação da similaridade fuzzy entre as palavras de cada coleção, tais palavras são agregadas por algoritmos de agrupamento espectral.

4.3.1 Coleções de Documentos

Foram utilizadas 4 coleções de documentos distintas. Com o objetivo de manter um controle mais próximo dos resultados do experimento, duas coleções foram formadas a partir de escolha pessoal. Com o objetivo de estabelecer parâmetros

comparativos nos resultados, duas coleções conhecidas e amplamente empregadas em experimentos que envolvem dado do tipo texto foram selecionadas.

4.3.1.1 Coleção 1 – Artigos Científicos

Esta coleção é formada por 200 artigos científicos, com assuntos relacionados a área de inteligência computacional, selecionados aleatoriamente no Google Acadêmico. Tais artigos tratam de 10 assuntos distintos: cognição, lógica fuzzy, algoritmos genéticos, redes neurais, mineração de dados, representação do conhecimento, aprendizado de máquina, reconhecimento de padrões, otimização e lógica.

4.3.1.2 Coleção 2 – Artigos Científicos

Esta coleção é formada por 160 artigos científicos, com assuntos relacionados a área de mineração de textos/recuperação de informação, selecionados aleatoriamente no Google Acadêmico. Tais artigos tratam de 8 assuntos distintos: agrupamento (clustering), análise de semântica latente, recuperação de informação, ontologia, semântica, relações fuzzy, extração de conceitos, modelo de tópicos.

4.3.1.3 Coleção 3 – Reuters 21578

Esta coleção vem sendo amplamente utilizada em experimentos que envolvem documentos, em especial experimentos de classificação. Tal coleção é composta de 21.578 notícias publicadas na rede de notícias Reuters em 1987. É classificada em 135 categorias, sendo a maioria sobre economia e negócios. Na realidade, apenas 12.902 documentos são utilizados em experimentos de classificação de texto, pois 8.676 documentos não foram considerados na classificação da base. Com a finalidade de tornar os resultados exeprimentais comparáveis, partições compostas por conjuntos de treinamento e teste foram definidas pelos criadores de base composta pelos 12.902 documentos. A partição mais utilizada é denoninada partição ModApté, onde 9.603 documentos foram selecionados para compor o conjunto de treinamento e 3.299 foram selecionados para compor o conjunto de teste. Apenas 115 categorias da partição ModApté possuem pelo menos um documento de treinamento e a maioria dos experimentos fazem uso de alguns subconjuntos destas categorias. Dentre estes

subconjuntos, um dos mais populares é o subconjunto das 8 categorias que possuem a maior quantidade de documentos de treinamento, denominado R8. Este subconjunto foi o escolhido para compor a terceira coleção de documentos utilizada neste trabalho.

4.3.1.4 Coleção 4 – Reuters 50-50

Esta coleção é um subconjunto da coleção RCV1 (Reuters Corpus Versão 1). Tem sido utilizada em experimentos de identificação de autores. Para cada um dos 50 autores com os maiores textos produzidos (em relação ao tamanho dos artigos) foram selecionados 50 textos relacionados com pelo menos um subtópico CCAT (corporativo/industrial). Dessa forma, tenta-se minimizar o fator tópico na distinção dos textos. O conjunto de treinamento consiste de 2.500 textos (50 por autor) e o conjunto de teste inclui outros textos 2500 (50 por autor) não coincidentes com os textos de treinamento.

4.3.2 Pré-Processamento

Cada uma das coleções foi submetida ao mesmo processo de pré-processamento dos documentos. Inicialmente, na etapa de limpeza, foram removidas todas as stopwords, utilizando a ferramenta WVTools, disponível em http://sourceforge.net/projects/wvtool. Foi considerado o conjunto básico de stopwords para a língua inglesa. Foram removidas, também, todas as palavras não classificadas como substantivos. Para tal, foi utilizado um tagger com a implementação disponível em http://dragon.ischool.drexel.edu/. As palavras restantes foram reduzidas aos seus radicais. Em um passo seguinte, a similaridade fuzzy foi calculada pela aplicação da equação (4.4).

4.3.3 Agrupamento de Palavras

Finalmente, as palavras e suas respectivas similaridades foram submetidas aos algoritmos de agrupamento espectral cuja implementação encontra-se em http://www.mathworks.com/matlabcentral/fileexchange/34412. A ferramenta implementa os três principais algoritmos de agrupamento espectral conforme apresentado no capítulo 2. Tais algoritmos requerem a informação da quantidade de grupos que será formada com os dados. Para cada uma das bases, a escolha da

quantidade se baseou na quantidade de categorias contidas em cada uma das coleções de texto, a saber, 10 para a base da apresentada na seção 4.3.1.1, 8 para a base da apresentada na seção 4.3.1.2 , 8 para a base da apresentada na seção 4.3.1.3 e 50 para a base da apresentada na seção 4.3.1.4. A justificativa para esta escolha é a possibilidade de controle dos grupos gerados com base nas categorias previamente conhecidas. Para uma segunda avaliação, as quantidades de grupos foram reduzidas as suas respectivas metades. Nestas condições, o objetivo foi avaliar a capacidade de generalização do algoritmo ao agrupar palavras em uma quantidade menor de grupos. Todos os parâmetros dos algoritmos foram mantidos em seus valores default.

4.4 Grânulos Produzidos

As Tabelas 1, 2, 3 e 4 a seguir apresentam parcial ou integralmente os grânulos produzidos a partir da aplicação dos procedimentos descritos na seção 4.3. Todas as tabelas apresentam uma seleção das palavras mais significativas.

Tabela 1. Grânulos da Coleção 1

GRÂNULO	ASSUNTO	PALAVRAS
01	machine learning	computer, aspect, behavior, intelligence, paradigm
02	----	exploration, benchmark, architecture, variation, characteristic, interaction, fact
03	neural network	extension, importance, neuron, goal, stability, property, choice
04	knowledge management	storage, knowledge, capability, management, path, business
05	cognition	representation, theory, life, language, cognition
06	pattern recognition	conclusion, input, classification, region, element, application
07	genetic algorithm	population, fitness, optimum, member, algorithm, convergence, solution
08	----	importance, performance, definition, statistics, measurement

09	data mining	attention, data, concept, generalization, addition, relationship
10	----	extraction, example, relations, variable, analysis, satisfaction

Tabela 2. Grânulos da Coleção 2

GRÂNULO	ASSUNTO	PALAVRAS
01	semantic	evolution, entity, library, management, language, technology, ontology, domain, description, semantics
02	latent semantic analysis	subspace, combination, detection, decomposition, association, retrieval, matrix, effectiveness, vector, collection
03	clustering	example, prototype, constraint, tendency, algorithm, objective, possibility, principle, data, problem,
04	information retrieval	period, kind, property, relations, decomposition, retrieval, information, expansion, criterion, construction
05	concept extraction	extension, representation, evaluation, concept, strategy, selection, explanation, logic, interpretation, identification, text, baseline
06	ontology	mechanism, classifier, correlation, thesaurus, creation, ontology, context, integration, recognition, source, module.
07	fuzzy relations	membership, co-occurrence, set, binary
08	topic models	probability, language, processing, mixture, model, generator

Tabela 3. Grânulos da Coleção 3

GRÂNULO	PALAVRAS
01	act , agreed , announc, bank , capit , cash , commis, common, deal , disclos , exchang, firm , fin , gener, group, invest, industr , interest, investm , offic prev, talk , tender , told , shareholder , stock , undisclos.
02	barrel , compan ,countr , corp , energ , expect , export , minister , opec , offic , report , pric , dlr , week , industr , mark , minister , petrol, corp , rais , increas.
03	split , board , shar , record , compan , annu , increas , payout , ros , bas , revenu , account , cash , asses , declar , distribut , month , nam , pretac.
04	agricultur , markes , month , produc , report , fal , farm , govern , depart.
05	week , markes , lower , gov , term , major , mone , half , point , lens , discount , billion , fund , cut.
06	intern , fin , exchang , compar , trad , level , rat , cur, dol , dlr , markes , mees , econ , expect , feder , com , agreed , countr .
07	port , ton , year , comp , work , report , south , vessel , spokesman , st , offic , strik , union , offer , worker
08	expo , act , countr , intern , unit , st, minister, offic , econom , week , reduc , surplus , mak , gener , agre , tariff , inclus , iss , hos , deficit

Tabela 4. Grânulos da Coleção 4

GRÂNULO	PALAVRAS
01	sen , secur , compan , fin , issu , propos , regl , republican , technolog , softwar , inform , limis , commerc , bil , st , inclus , corp , clint , administer , hous , offic , group.
02	markes , remain , analyst , exchang , pragu , ris , strong , show , centr , cur , indic , point , said , czech , investor, begin , strength , increas , issu , mark , crown , record , dol , baskes.

03	britain , markes , bank , offic , said , investm , valu , execut , direct , mak , ad , tim , major , futur , year , group , compan , week , secur , result.
04	chin , beij , author , polic , offic , governm , polit , inclus , nat , countr , allow , foreign , whes.
05	group , new , corp , analyst , expect , perc , profit , year , result , sydne , austral , earl , million , competit , increas , major , look , fin , governm , tak , own , cur.
06	fin , compan , said , million , debt , stock , perc , valu , shar , tim , bas , markes , earn , result , rat st , oper , inclus , look , pric , expect.
07	miner , said , wednesda , busang , gold , centur , partner , chief , execut , announc , indones , test , insignif , amount , sampl , prelimin , releas , strathcon.
08	motor , offer , car , vehicl , truck , compan , unit , st , markes , automaker , inclus , gener , corp , year , annu , millio , billion , expect , chrysler , intern , industr .
09	compan , execut , reuter , expect , group , sal , major , year , oper , seen , mill , ad , valu , london , britain , month , part , number , cur , want , analyst.
10	technolog , expect , revenu , quarter , million , compan , communic , cal , growth , pric , york , plan , system , trad , network , deal , chair , internet , consumer , offer , acces.

As Tabelas 5 e 6 apresentam parcial ou integralmente os grânulos produzidos a partir da redução das quantidades de grupos das coleções 1 e 2 para as suas respectivas metades, conforme definido na seção 4.3.

Tabela 5. Grânulos da Coleção 1 (Reduzido)

GRÂNULO	ASSUNTO	PALAVRAS
01	genetic algorithm / optimization	exploration, performance, fitness, operator, member, algorithm, convergence, solution, population, optimum, crossover
02	neural networks	extension, input, example, property, regression, analysis, neuron, procedure, realization, synthesis, vector, coefficient, manner, applicability

03	data mining / knowledge management	user, technique, topic, storage, knowledge, management, capability, information, methodology, data, business, database
04	cognition / logic	behavior, theory, life, paradigm, language, computer, principle, aspect, manipulation, intelligence
05	cognition	protocol, difference, relations, complexity, analysis, problem, role, system, cognition, method, application

Tabela 6. Grânulos da Coleção 2 (Reduzido)

GRÂNULO	ASSUNTO	PALAVRAS
01	semantic/ ontology	development, evolution, entity, library, management, language, version, technology, ontology, methodology, domain, description, semantics, input, mechanism, classifier, correlation, thesaurus, creation, ontology, context, integration, identification, recognition, source, module.
02	latent semantic analysis/ concept extraction	item, user, basis, subspace, combination, detection, decomposition, association, retrieval, matrix, effectiveness, vector, collection, method, extension, representation, evaluation, concept, strategy, selection, explanation, addition, logic, interpretation, identification, text, baseline
03	clustering/ information retrieval	example, prototype, constraint, tendency, algorithm, objective, possibility, finding, principle, data, problem, difficulty, period, user, minimum, kind,

		property, relations, decomposition, retrieval, information, expansion, criterion, method, construction
04	topic models	probabilistic, language, processing, mixture, model, generative

4.5 Avaliação dos Grânulos

Comumente, coleções de textos têm sido representadas como matrizes de co-ocorrência de palavras e documentos. Implícita neste tratamento está a hipótese de que a informação contida em um documento pode ser representada pela soma das informações contidas nas palavras que o compõe (MANNING & RAGHAVAN & SCHÜTZE, 2008). Empiricamente, esta abordagem mostra que o ganho advindo da incorporação de mais informações na representação dos documentos não supera o prejuízo causado pelo aumento da dimensionalidade dos dados. Os modelos de análise semântica (HOFMANN, 1999) (MANNING, 2008) (STEYVERS & GRIFFITHS, 2006) (MANNING & RAGHAVAN & SCHÜTZE, 2008) foram desenvolvidos como solução para o impasse entre a incorporação de informação e o aumento de dimensionalidade e têm demonstrado eficácia na identificação dos conceitos contidos nos documentos em diversos tipos de experimentos.

Esta seção apresenta resultados comparativos da aplicação de duas das principais abordagens desenvolvidas com este propósito e a abordagem apresentada neste trabalho.

4.5.1 Grânulos X Conceitos LSA

A técnica de Análise de Semântica Latente (Latent Semantic Analysis – LSA) explora a relação existente entre as palavras e os textos nos quais elas aparecem para construir um espaço vetorial onde similaridades de significado podem ser estabelecidas. Este novo espaço é conhecido como Espaço Conceito ou Espaço Semântico (DEERWESTER, 1990), sendo que a proximidade entre significados é

proporcional ao ângulo entre vetores neste espaço. Basicamente, LSA consiste na construção de uma matriz A que informa a co-ocorrência de palavras e documentos (palavras X documentos). Após, a matriz A é decomposta algebricamente segundo a decomposição em valores singulares (SVD) de forma a aproximar a matriz A por combinações lineares.

Formalmente, seja A uma matriz onde o elemento (i; j) descreve a ocorrência do termo i no documento j (por exemplo, a frequência de i em j). Se A tem dimensões m n, onde m é o número de termos e n é a quantidade de documentos, a SVD de A é definida como:

$$A = UDV^T \tag{4.1}$$

Onde $U = [u_{ij}]$ é uma matriz ortonormal m x m cujas colunas são chamadas de vetores singulares a esquerda; $D = diag(\sigma_1; \sigma_2; ... ; \sigma_n)$ é uma matriz diagonal m x n cujos elementos são chamados de valores singulares não negativos, os quais aparecem ordenados de forma decrescente; e $V = [v_{ij}]$ é uma matriz ortonormal n x n cujas colunas são chamadas de vetores singulares a direita. Se posto(A) = k a SVD pode ser interpretada como o mapeamento do espaço de A em um espaço conceito (reduzido) de k dimensões, as quais são linearmente independentes. Ainda que posto(A) seja maior que k, quando seleciona-se apenas os k maiores valores singulares ($\sigma_1; \sigma_2; ... ; \sigma_n$) e seus correspondentes vetores singulares em U e V, pode-se obter uma aproximação de posto k para a matriz A. Deste modo, pode-se tratar vetores de termos e documentos como um espaço conceito. Neste novo espaço, os vetores de termos em U tem k entradas, cada um dando a ocorrência do termo i em um dos k conceitos. Da mesma forma, os vetores de documentos em V revelam a relação entre o documento j com cada conceito k. Usualmente formaliza-se o espaço conceito como

$$A_k = U_k D_k V_k^T \tag{4.2}$$

Deste modo, pode-se então verificar em V_k o quão relacionados estão dois dados documentos j e l, possibilitando a aplicação de técnicas de agrupamento de

documentos no espaço conceito. A implementação do método utilizada neste trabalho pode ser encontrada em http://www.mathworks.com/matlabcentral/fileexchange/22795.

Cada uma das células nas Tabelas 9, 10, 11 e 12 a seguir, representa o percentual de similaridade (ou de interseção) entre os grânulos e os conceitos LSA. Para facilitar a análise destes resultados, as células com as maiores medidas de similaridade estão destacadas.

Tabela 7. Equivalência entre grânulos e conceitos LSA para a Coleção 1.

		LSA												
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
GRANULO	**1**	0,39	0,53	0,50	0,49	0,45	0,42	**0,92**	0,42	0,42	0,45	0,51	0,32	0,56
	2	**0,96**	0,41	0,55	0,43	0,45	0,60	0,42	0,56	0,38	0,44	0,32	0,45	0,34
	3	0,46	0,67	0,43	0,40	0,45	0,56	0,50	**0,89**	0,42	0,34	0,23	0,56	0,42
	4	0,58	0,67	0,76	0,40	0,40	0,54	0,45	0,78	0,23	0,34	0,56	0,56	**0,92**
	5	0,34	0,45	0,76	0,23	0,40	0,54	0,45	0,78	**0,95**	0,34	0,56	0,56	0,23
	6	0,39	0,34	0,50	**0,87**	0,45	0,42	0,67	0,42	0,42	0,67	0,51	0,68	0,45
	7	0,46	0,24	0,43	0,46	0,45	0,56	**0,78**	0,45	0,42	0,34	0,25	0,56	0,42
	8	0,78	**0,85**	0,36	0,32	0,45	0,60	0,42	0,15	0,38	0,44	0,47	0,45	0,39
	9	0,39	0,34	0,36	**0,87**	0,45	0,42	0,67	0,47	0,68	0,67	0,51	0,68	0,76
	10	0,45	0,48	0,50	0,35	0,47	0,42	0,67	0,42	0,65	0,67	**0,90**	0,68	0,45

Tabela 8. Equivalência entre grânulos e conceitos LSA para a Coleção 2.

GRANULO	LSA									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,39	0,53	**0,96**	0,49	0,32	0,42	0,51	0,42	0,42	0,45
2	0,43	0,41	0,32	0,43	0,45	0,60	0,42	**0,96**	0,38	0,44
3	0,46	0,67	0,23	**0,88**	0,68	0,56	0,50	0,47	0,42	0,34
4	0,58	0,67	0,56	0,40	0,56	0,57	0,45	0,78	0,23	**0,87**
5	0,34	0,45	0,56	0,23	0,56	0,54	0,45	0,78	**0,95**	0,34
6	0,39	0,34	0,51	0,33	0,68	**0,82**	0,67	0,42	0,42	0,67
7	0,46	0,24	0,25	0,46	0,56	0,56	**0,89**	0,45	0,42	0,34
8	0,78	**0,92**	0,47	0,32	0,45	0,60	0,42	0,15	0,38	0,44

Tabela 9. Equivalência entre grânulos e conceitos LSA para a Coleção 3.

GRANULO	LSA									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,79	0,86	**0,91**	0,77	0,83	0,89	0,89	0,75	0,63	0,89
2	0,98	0,61	0,68	0,69	0,87	0,82	0,88	**0,99**	0,65	0,52
3	0,76	0,36	0,88	**0,92**	0,71	0,58	0,73	0,75	0,86	0,89
4	0,52	0,73	0,66	0,89	0,88	0,67	0,71	0,86	0,72	**0,90**
5	0,57	0,76	0,79	0,78	0,76	0,77	0,80	0,70	**0,91**	0,70
6	0,62	0,63	0,60	0,89	0,88	**0,93**	0,85	0,90	0,81	0,65
7	0,83	0,89	0,63	0,66	0,65	0,77	**0,91**	0,66	0,51	0,86
8	0,76	**0,95**	0,89	0,65	0,88	0,79	0,87	0,86	0,86	0,84

Tabela 10. Equivalência entre grânulos e conceitos LSA para a Coleção 4.

GRANULO	LSA												
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
1	0.79	0.53	0.53	0.45	0.45	0.42	**0.96**	0.42	0.42	0.45	0.51	0.32	0.56
2	**0.92**	0.41	0.55	0.44	0.45	0.60	0.42	0.686	0.37	0.44	0.32	0.45	0.34
3	0.46	0.67	0.27	0.40	0.45	0.56	0.50	**0.92**	0.42	0.39	0.43	0.56	0.41
4	0.58	0.67	0.76	0.43	0.67	0.55	0.45	0.77	0.23	0.34	0.57	0.50	**0.89**
5	0.34	0.45	0.56	034	0.40	0.54	0.45	0.78	**0.87**	0.34	0.56	0.58	0.23
6	0.30	0.38	0.50	**0.89**	0.67	0.42	0.67	0.42	0.32	0.36	0.69	0.44	0.43
7	0.48	0.67	0.47	0.44	0.23	0.32	**0.83**	0.45	0.42	0.34	0.25	0.56	0.42
8	0.56	**0.87**	0.46	0.22	0.56	0.68	0.43	0.67	0.38	0.46	0.54	0.36	0.42
9	0.43	0.37	0.65	**0.84**	0.54	0.41	0.68	0.56	0.69	0.68	0.44	0.64	0.73
10	0.43	0,45	0.53	0.56	0.89	0.55	0.68	0.43	0.54	0.76	**0.85**	0.55	0.55

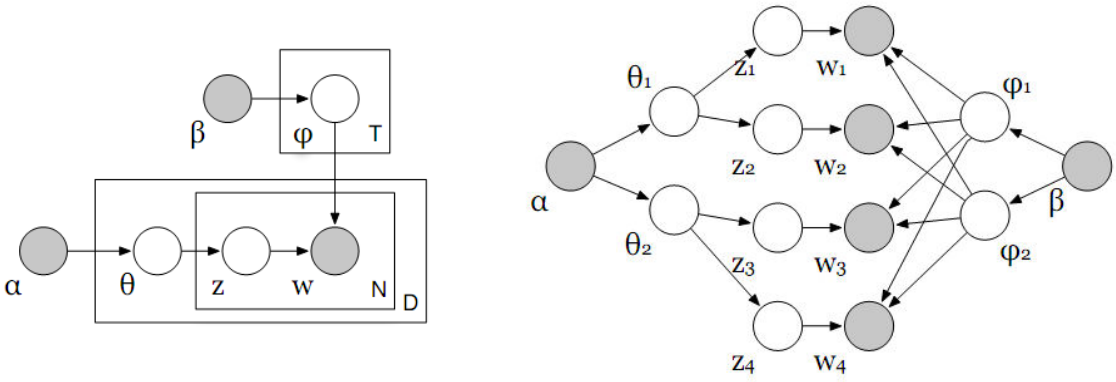
4.5.2 Grânulos X Tópicos LDA

Latent Dirichlet Allocation é um poderoso arcabouço para a modelagem de coleções de dados que recentemente tem sido aplicado a diversas tarefas, especialmente nas áreas de processamento de linguagem natural e visão computacional (XING, 2007).

De forma sucinta, pode-se descrever o processo generativo de um documento descrito pelo LDA da seguinte forma: Primeiro, amostre T vetores positivos φ_j de uma distribuição Dirichlet com parâmetros β. Os vetores φ_j são parâmetros de distribuições discretas de palavras denominadas tópicos, que representam os diferentes possíveis assuntos que um autor pode utilizar ao redigir um documento. Então, amostre um vetor positivo θ de uma distribuição Dirichlet com parâmetro α. O vetor θ_d é o parâmetro de uma distribuição discreta que indica a proporção de cada tópico φ_j no documento. Para cada palavra no documento, primeiro amostre uma variável de rótulo de tópico z de uma distribuição discreta com parâmetros θ_d, então amostre uma palavra w do tópico respectivo φ_z. O modelo é representado graficamente na Figura 15(a). No grafo

direcionado da figura, cada vértice indica uma variável, e arcos indicam dependências diretas. As caixas indicam repetição de uma variável pelo número no canto inferior direito da caixa. Vértices sombreados e não sombreados representam, respectivamente, variáveis observadas (ou definidas pelo usuário) e latentes. Por exemplo, a Figura 15(b) ilustra o modelo LDA para uma coleção de dois documentos, cada qual com duas palavras, e dois tópicos. O processo generativo pode ser concisamente descrito pela notação abaixo, onde o símbolo ~ significa *amostrado de*.

$$\varphi_j \sim \text{Dirichlet}(\beta)$$
$$\theta_d \sim \text{Dirichlet}(\alpha)$$
$$z_i \sim \text{Discreta}(\theta d_i)$$
$$w_i|z_i = j \sim \text{Discreta}(\varphi_j)$$



(a) (b)

Figura 15. Modelo LDA

A implementação do método utilizada neste trabalho pode ser encontrada em http://psiexp.ss.uci.edu/research/programs_data/toolbox.htm#Matlab_Functions.

Da mesma forma como apresentado na seção anterior, cada uma das células nas Tabelas 13, 14, 15 e 16 a seguir, representa o percentual de similaridade (ou de interseção) entre os grânulos e os conceitos LDA. Para facilitar a análise destes resultados, as células com as maiores medidas de similaridade estão destacadas.

Tabela 11. Equivalência entre grânulos e conceitos LDA para a Coleção 1.

GRANULO	LDA									
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
1	0,68	0,85	0,45	**0,98**	0,89	0,83	0,94	0,71	0,4	0,85
2	0,72	**0,99**	0,82	0,91	0,51	0,85	0,47	0,75	0,42	0,95
3	0,98	0,65	0,88	0,44	0,51	0,57	0,90	**0,99**	0,68	0,58
4	**0,98**	0,92	0,41	0,95	0,52	0,42	0,55	0,43	0,47	0,92
5	0,77	0,87	0,93	0,8	0,78	0,44	0,83	0,4	**0,99**	0,94
6	0,85	0,64	0,45	0,73	0,41	**0,97**	0,51	0,92	0,96	0,78
7	0,75	0,61	0,89	0,51	0,84	0,57	**0,98**	0,72	0,54	0,76
8	0,79	0,77	0,58	0,53	0,96	0,77	0,42	0,95	0,89	**0,98**

Tabela 12. Equivalência entre grânulos e conceitos LDA para a Coleção 2.

GRANULO	LDA												
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
1	0,88	0,95	0,59	0,82	0,47	0,89	0,72	0,94	0,71	0,81	**0,99**	0,84	0,76
2	0,86	0,48	0,57	0,73	0,8	**0,98**	0,95	0,73	0,96	0,97	0,59	0,77	0,85
3	0,91	0,93	**0,97**	0,76	0,42	0,90	0,78	0,73	0,74	0,45	0,57	0,68	0,91
4	0,47	0,98	0,91	0,42	0,76	0,93	0,66	0,48	0,45	**0,99**	0,56	0,68	0,91
5	0,91	0,76	0,41	**0,99**	0,47	0,98	0,94	0,6	0,55	0,94	0,45	0,77	0,41
6	0,69	0,82	0,47	0,82	0,84	0,72	0,7	0,49	0,85	0,93	0,88	0,49	**0,93**
7	0,89	**0,98**	0,91	0,58	0,61	0,67	0,85	0,95	0,77	0,56	0,98	0,75	0,43
8	0,47	0,54	0,9	0,79	0,72	0,90	**0,98**	0,95	0,56	0,64	0,95	0,54	0,77
9	0,64	0,85	0,74	0,71	**0,93**	0,64	0,4	0,97	0,72	0,88	0,71	0,86	0,55
10	0,73	0,79	0,78	0,72	0,41	0,92	0,43	**0,95**	0,86	0,91	0,72	0,89	0,55

Tabela 13. Equivalência entre grânulos e conceitos LDA para a Coleção 3.

GRANULO	LDA												
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
1	0,68	0,56	0,79	0,80	0,67	0,69	0,42	**0,84**	0,73	0,81	0,57	0,68	0,77
2	0,66	0,48	0,57	0,73	0,8	0,88	0,95	0,73	0,96	**0,98**	0,59	0,77	0,85
3	0,78	0,93	0,44	0,76	0,42	**0,90**	0,78	0,73	0,74	0,45	0,57	0,45	0,82
4	0,48	0,96	0,91	0,42	0,76	0,93	0,66	0,48	0,45	0,59	0,56	0,34	**0,97**
5	0,70	**0,96**	0,43	0,78	0,47	0,83	0,94	0,6	0,55	0,84	0,47	0,77	0,41
6	0,58	0,86	0,47	0,84	0,67	0,70	0,7	0,49	0,67	0,93	**0,98**	0,49	0,93
7	0,79	0,90	**0,96**	0,58	0,61	0,67	0,85	0,95	0,77	0,57	0,68	0,75	0,43
8	0,77	0,55	0,9	0,79	0,72	0,90	0,78	0,95	0,56	0,58	0,85	**0,95**	0,77
9	0,64	0,84	0,74	**0,91**	0,72	0,66	0,37	0,87	0,74	0,88	0,71	0,86	0,55
10	0,73	0,70	0,78	0,72	0,41	0,92	0,43	0,85	**0,96**	0,91	0,72	0,89	0,78

Tabela 14. Equivalência entre grânulos e conceitos LDA para a Coleção 4

GRANULO	LDA												
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
1	0,64	0,86	0,74	0,37	0,86	0,66	0,74	0,87	0,76	**0,88**	0,71	0,86	0,72
2	0,45	0,77	**0,97**	0,95	0,77	0,88	0,57	0,73	0,96	0,78	0,59	0,77	0,8
3	0,55	0,68	0,44	0,78	0,68	**0,94**	0,44	0,73	0,74	0,45	0,57	0,68	0,42
4	**0,93**	0,68	0,73	0,66	0,68	0,93	0,71	0,48	0,45	0,59	0,56	0,68	0,76
5	0,77	0,77	0,43	0,94	0,77	0,83	0,43	**0,95**	0,55	0,84	0,57	0,77	0,47
6	0,56	**0,94**	0,47	0,7	0,49	0,70	0,47	0,49	0,67	0,93	0,56	0,49	0,67
7	0,45	0,75	0,76	0,85	0,75	0,67	0,86	0,85	0,77	0,57	0,47	0,75	**0,91**
8	0,77	0,85	0,9	**0,97**	0,95	0,90	0,9	0,95	0,56	0,58	0,76	0,85	0,72
9	0,64	0,86	0,74	0,37	0,86	0,66	0,74	0,87	0,74	0,88	**0,94**	0,86	0,72
10	0,68	0,68	0,79	0,42	**0,97**	0,69	0,79	0,84	0,73	0,81	0,85	0,68	0,67

4.5.3 Avaliação dos Resultados

Observando os resultados apresentados nas Tabelas 1 e 2 pode-se perceber que a técnica combina palavras suficientemente significativas para representar cada um dos tópicos das coleções de teste. Para a Coleção 1, que contém texto sobre inteligência computacional, sete tópicos foram facilmente identificados a partir das palavras que se associaram nos grânulos. Para a Coleção 2, contendo textos sobre mineração de texto e recuperação de informação, a técnica alcança resultados ainda melhores, pois os oito assuntos que compõem a coleção foram facilmente identificados.

Os resultados apresentados na Tabelas 5 e 6 mostram que a técnica mostra boa capacidade em generalizar os grânulos contidos nas coleções. Podemos ressaltar, por exemplo, os agrupamentos de palavras que descrevem os tópicos algoritmos genéticos/otimização (genetic algorithms/optimization) e mineração de dados/gestão do conhecimento (data mining/ knowledge management). Os dois tópicos são fortemente relacionados. Esta constatação mostra que a técnica apresenta consistência pois é capaz de capturar estes relacionamentos agrupando as palavras contidas em seus respectivos documentos.

4.6 Considerações Finais

O processo de granulação de palavras é fundamentalmente baseado no agrupamento de palavras que mantém uma relação de indiscernibilidade entre si. Tal indiscernibilidade é medida através de uma relação de similaridade que descreve, segundo uma abordagem fuzzy, a coocorrência de palavras em documento. Este capítulo apresentou o processo de construção destes grânulos, bem como os resultados alcançados. O capítulo também apresentou um estudo comparativo dos resultados obtidos com abordagem proposta com os resultados alcançados por duas técnicas bem conhecidas e de objetivos semelhantes. O capítulo seguinte apresenta um estudo da apliacação dos grânulos de palavras a tarefa de classificação de documentos.

5 CLASSIFICAÇÃO BASEADA EM GRÂNULOS

Segundo (SEBASTIANI, 2005), classificação ou categorização de textos é a tarefa de dividir, automaticamente, um conjunto predefinido de documentos em classes ou categorias. Trata-se de uma tarefa de aprendizado supervisionado que possui diversas aplicações práticas, tais como, criação automática de diretórios Web, filtragem de spam, identificação de autoria, marketing direto, entre outras. A representação dos documentos, usados como exemplos de treinamento e teste, e o esquema de redução de características são determinantes na eficiência dos algoritmos de classificação. Este capítulo apresenta a aplicação dos grânulos de palavras na representação destes documentos, a sua influência sobre os resultados na classificação destes documentos e a comparação destes resultados com a técnicas Análise de Semântica Latente (LSA) e Latent Dirichlet Allocation (LDA), comumente usadas com o objetivo de criar modelos mais adequados a tarefa de classificação.

5.1 Introdução e Trabalhos Relacionados

A representação de documentos mais comum em tarefas de classificação de textos é a representação por Bag of Words (BOW). Tal representação considera algum tipo de contagem de cada palavra ou termo ignorando a ordem com estes ocorrem ou a relação que estes mantém entre si. Desta forma, mescla eficiência computacional com a retenção do conteúdo do documento. O resultado é um vetor que pode ser analisado ou utilizado em diversas técnicas de aprendizado de máquina e, em especial, em algoritmos de classificação. Tal representação, apesar de eficaz, apresenta dois problemas graves: a alta dimensionalidade do espaço de características e a perda da informação contida nos textos originais.

Muitas técnicas de redução de características têm sido propostas para minimizar os efeitos do problema da alta dimensionalidade (DASH & LIU, 1997) (YANG & PEDERSEN, 1997). Para lidar com a incapacidade do modelo em representar as relações entre as palavras, muitos trabalhos têm sugerido o emprego de LSA e LDA como boas soluções para a improvisação dos resultados na tarefa de classificação de textos. (WANG, 2003) e (BARAK, 2009) usam LSA com este objetivo e descrevem

bons resultados. (SRIURAI, 2011) e (BIRÓ, 2009) empregam LDA nesta tarefa e também apresentam bons resultados. O trabalho descrito em (ANAYA, 2011) compara os resultados alcançados com o emprego da LSA e LDA com classificadores humanos. O experimento demonstrou que ambas as técnicas apresentam desempenho semelhante quando comparadas ao desempenho do processo executado por um humano.

5.2 Classificadores

Existem vários tipos de técnicas de aprendizado para a classificação. Tais técnicas incluem métodos probabilísticos, métodos de regressão, árvores de decisão, redes neurais, algoritmos genéticos, modelo oculto de Markov, máquinas de vetores suporte, comitês de classificadores, entre outros. O modelo de documento proposto neste trabalho foi submetido a quatro destes classificadores. Tais classificadores são K-Vizinhos mais Próximo (K-Nearest Neighbors - K-NN), classificador Naive Bayes, máquinas de vetores suporte e comitês de classificadores (Boosting). Os classificadores K-NN e Naive Bayes foram escolhidos devido a simplicidade na clasificação. Os demais foram escolhidos por se tratarem das mais eficientes técnicas de classificação da atualidade de acordo com (SEBASTIANI, 2005).

5.2.1 K-Vizinhos mais Próximos (K -Nearest Neighbors - K-NN)

Trata-se de classificador onde o aprendizado se baseia em similaridade. Os exemplos de treinamento são representados por vetores n-dimensionais que representam pontos no espaço n-dimensional. Para determinar a classe de um elemento que não pertença ao conjunto de treinamento, o classificador procura os k elementos do conjunto de treinamento que estejam mais próximos deste elemento desconhecido, ou seja, aqueles elementos que apresentam a menor distância em relação a este elemento desconhecido. Estes k elementos são chamados k-vizinhos mais próximos. Após a verificação das classes destes k-vizinhos, aquela mais frequente é atribuída ao novo elemento.

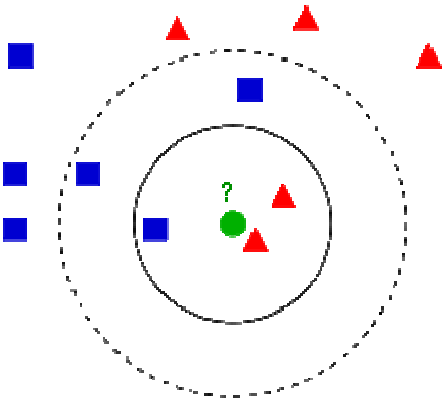


Figura 16. K-Vizinho mais Próximo.

Várias métricas podem ser usadas na avaliação desta distância, sendo a mais comum a distância euclidiana, definida como

$$d(x,y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + ... + (x_n - y_n)^2} \qquad (5.1)$$

onde X = (x_1 , x_2 , ... x_n) e Y= (y_1 , y_2 , ... y_n) representam dois pontos do $\mathbb{R}^n$.

O classificador possui apenas um parâmetro livre, a quantidade de k-vizinhos, que pode ser controlado para a obtenção de uma melhor classificação. Informações mais detalhadas podem ser obtidas em (MITCHELL, 1997). A implementação escolhida é a encontrada na ferramenta Weka, disponível em http://www.cs.waikato.ac.nz/ml/weka.

5.2.2. Classificador Naive Bayes

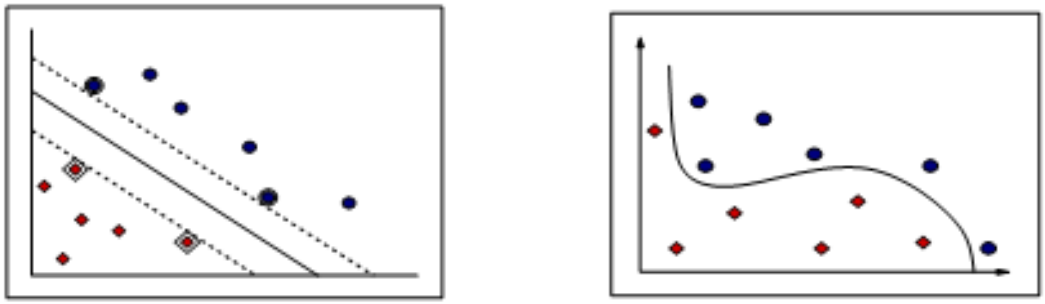
Um classificador bayesiano é um método probabilístico para a tarefa de classificação. Pode ser usado para determinar a probabilidade de um documento pertencer a classe Ci dados valores de atributos como exemplos, ou seja, P (Ci | A1 = V1, A2 = V2, ..., An = Vn) Como os valores dos atributos são considerados independentes, esta probabilidade é proporcional a $P(C_i) = \prod_1^k P(A_k = V_k \mid C_i)$. Onde, P (Ci) que significa a probabilidade da classe Ci ocorrer e P (Ak = Vk | Ci), corresponde a probabilidade de um documento ocorrer na classe Ci contendo os atributos A1, A2, ..., Ak e podem ser estimadas a partir dos dados de treinamento. Para determinar a classe

de um documento, a probabilidade para cada classe é computada. O documento é, então, associado aquela que apresentar maior probabilidade.

5.2.3 Máquinas de Vetores Suporte (Support Vector Machine – SVM)

O método foi introduzido em classificação de textos por (JOACHIMS, 1997). Fundamentada na Teoria de Aprendizado Estatístico, foi desenvolvida por (VAPNIK, 1995), com o intuito de resolver problemas de classificações de padrões. A técnica, originalmente desenvolvida para classificação binária, busca a construção de um hiperplano como superfície de decisão, de tal forma que a separação entre os exemplos seja máxima. Isso considerando padrões linearmente separáveis. Já para padrões não-linearmente separáveis, busca-se uma função de mapeamento Φ apropriada para tornar o conjunto mapeado linearmente separável. Devido a sua eficiência em trabalhar com dados em alta dimensionalidade é definida na literatura como uma técnica altamente robusta, muitas vezes comparada às Redes Neurais (SUNG & MUKKAMALA, 2003).

O modelo mais simples de SVM, chamado Classificador de Margem Máxima, trabalha apenas com dados linearmente separáveis, ficando restrito, portanto, à poucas aplicações práticas. Apesar desta limitação, este modelo apresenta propriedades importantes e foi o ponto de partida para a formulação de SVMs mais sofisticadas. A Figura 17(a) mostra um espaço linearmente separável para um conjunto de treinamento bidimensional e a Figura 17(b) ilustra um espaço linearmente inseparável. A linha escura presente em ambas as figuras, é chamada de superfície de decisão (ou separação) e separa os vetores de entrada em classes distintas. Em particular, na Figura primeira, devido a linearidade da superfície de decisão, a mesma é conhecida com hiperplano de separação.



(a) (b)

Figura 17. Exemplos de (a)espaço linearmente separável e (b) espaço linearmente inseparável.

O Classificador de Margem Máxima otimiza limites no erro de generalização das máquinas lineares em termos da margem de separação entre as classes determinadas pelo hiperplano de separação. Essa estratégia envolve separar os dados com um tipo especial de hiperplano, o hiperplano de margem máxima ou de separação ótima.

Sendo $f(x) = (w \cdot x) + b$ um hiperplano, podemos definir margem como a menor distância entre os exemplos do conjunto de treinamento e o hiperplano utilizado para a separação destas classes. A margem determina quão bem duas classes podem ser separadas (SMOLA, 1999).

Definição 5.1 A margem ρ de um classificador f é definida por

$$\rho = \min_{i} y_i f(x_i) \qquad (5.2)$$

A margem é obtida pela distância entre o hiperplano e os vetores que estão mais próximos a ele, denominados vetores suporte. De acordo com (SMOLA, 1999) os vetores suporte são padrões críticos, que sozinhos determinam o hiperplano ótimo, sendo os outros padrões (não críticos) irrelevantes, podendo ser removidos do conjunto de treinamento sem afetar os resultados. Na Figura 18 os vetores suporte são destacados por círculos externos nos padrões.

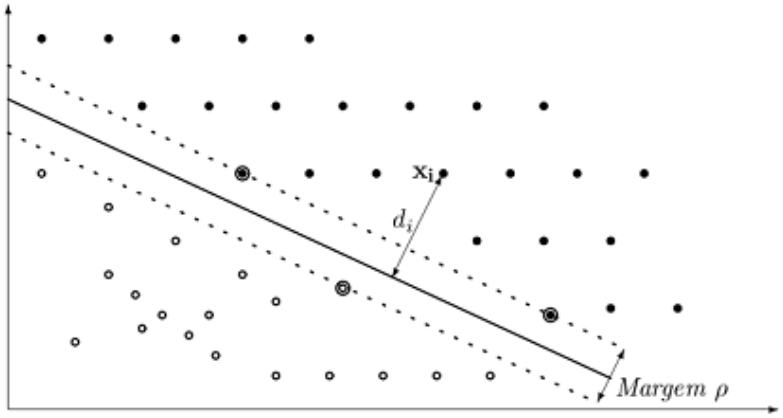


Figura 18. Margem e Vetores Suporte.

Uma classificação linear consiste em determinar uma função f: X⊆ $\mathbb{R}^N$ que atribui rótulo +1 se f(x) ≥ 0 e -1, caso contrário. Considerando uma função linear, podemos representá-la como

$$f(x) = < w \,.\, x > + b = \sum_{i=1}^{n} w_i x_i + b \qquad (5.3)$$

onde w e b ∈ $\mathbb{R}^N$ x $\mathbb{R}^N$, são conhecidos como vetor peso e bias e são responsáveis por controlar a função e a regra de decisão. Os valores de w e b são obtidos por aprendizado a partir dos dados de entrada. O vetor peso (w) o bias (b) podem ser interpretados geometricamente sobre um hiperplano. Um hiperplano é um subespaço afim, que divide um espaço em duas partes, correspondendo a dados de duas classes distintas. O vetor peso (w) define um direção perpendicular ao hiperplano, como mostra a Figura 19 e com a variação do bias o hiperplano é movido paralelamente a ele mesmo. Assim, um SVM linear busca encontrar um hiperplano que separe perfeitamente os dados da cada classe e cuja margem de separação seja máxima. Tal hiperplano é denominado hiperplano ótimo.

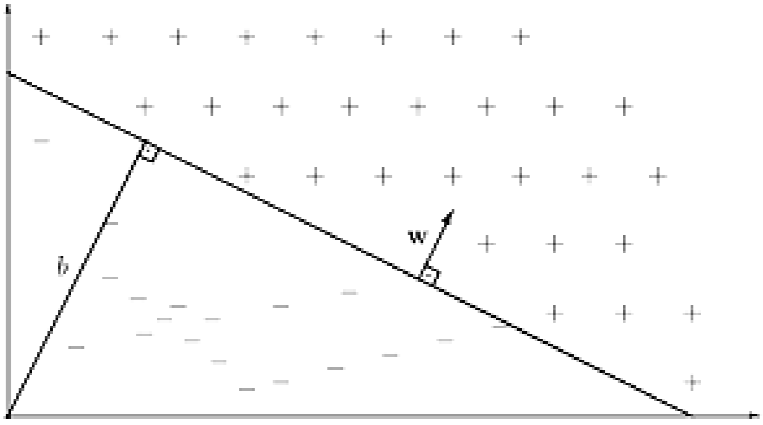


Figura 19. Interpretação geométrica de w e b sobre um hiperplano.

Para estender a SVM linear para a solução de problemas não lineares foram introduzidas funções reais, que mapeiam o conjunto de treinamento em um espaço linearmente separável, o espaço de características. Tais funções, definidas como funções Kernel, permitem que a SVM não linear realize uma mudança de dimensionalidade, caindo, assim, em um problema de classificação linear e podendo fazer uso do hiperplano ótimo.

As SVM foram propostas, inicialmente, como ferramentas de classificação binária. Para possibilitar a sua utilização na solução de problemas de classificação em múltiplas classes foram propostas algumas extensões a SVM binária. Em termos formais, em um sistema multiclasses o conjunto de treinamento é composto por pares (x_i,y_i) tal que $y_i \in \{1,2, ..., k\}$, com $k > 2$, onde k é o número de classes. As principais abordagens utilizam como base a decomposição de um problema multiclasse com k classes, $k > 2$, em k problemas binários. Informações mais detalhadas podem ser obtidas em (CRISTIANI, 2000). A implementação escolhida é a encontrada na ferramenta Weka, disponível em http://www.cs.waikato.ac.nz/ml/weka com todos os parâmetros default.

5.2.4 Comitês de Classificadores (Boosting)

O método de comitês de classificadores está baseado na idéia que k diferentes classificadores $\Phi_1, ... \Phi_k$ podem classificar melhor do que um único classificador se os julgamentos individuais de cada classificador forem apropriadamente combinados. No

método Boosting os k classificadores são obtidos pelo mesmo método de aprendizado e não são treinados de forma independente, mas sim de forma sequencial. Desta forma, durante o treinamento do classificador Φ_t, são levados em consideração os comportamentos dos demais classificadores e o treinamento é concentrado sobre os exemplos que levam estes outros classificadores aos piores resultados. A cada iteração, um algoritmo base é chamado para gerar um classificador simples, utilizando uma versão diferente do conjunto de treinamento. As diferentes versões do conjunto de treinamento são obtidas através da variação do peso associado a cada um dos exemplos. Assim, são geradas diferentes versões ponderadas do conjunto de dados. Após um determinado número de iterações, os diversos classificadores parciais são combinados, gerando um classificador único que, possivelmente, possui um desempenho melhor do que o do melhor classificador parcial. Todos os algoritmos derivados do Boosting adotam esse mesmo esquema geral. A diferença está em dois aspectos fundamentais: a forma de atualização da ponderação do conjunto de dados em cada iteração e a forma de combinação dos classificadores parciais. Informações mais detalhadas podem ser obtidas em (SCHAPIRE, 2012). A implementação escolhida é a encontrada na ferramenta Weka, disponível em http://www.cs.waikato.ac.nz/ml/weka com todos os parâmetros default.

5.3 Avaliação de Classificadores

Segundo (SEBASTIANI, 2005) , a eficiência do treinamento, isto é, a medida do tempo necessário para se construir um classificador Φi dado um corpus Ω, assim como a eficiência da classificação, isto é, a medida do tempo necessário para classificar um documento de acordo com Φ_i e a eficácia, isto é, a medida da correção de Φ_i na classificação, são todas medidas legítimas do sucesso para um classificador.

Em geral, nos trabalhos publicados nesta área, a eficácia é considerada o critério mais importante pois é a mais confiável quando o objetivo é comparar diferentes classificadores e porque a eficiência depende de parâmetros muito voláteis, como por exemplo, a plataforma de hardware e/ou software adotada.

Em tarefas de classificação de texto simples, com uma única classe, a eficácia é geralmente medida pela correção (accuracy), isto é, o percentual de decisões de classificação corretas (o erro é o inverso da correção, isto é, Erro = 1 - Correção).

Entretanto, para tarefas de classificação binária e, sobretudo, para tarefas de classificação em múltiplas classes, a correção não é a medida mais adequada.

Em tarefas de classificação binária, a eficácia em relação a uma categoria ou classe c_i é medida pela combinação da precisão em relação a esta classe c_i (π_i) , que representa o percentual de documentos classificados em c_i e que realmente pertencem a c_i e pela abrangência com relação a c_i (ρ_i) que representa o percentual de documentos que pertencem a c_i e que foram de fato classificados em em c_i. Em tarefas de classificação em múltiplas clássicas, quando a eficácia é computada para muitas categorias, os resultados de precisão e a abrangência das categorias individuais devem ser combinados de alguma forma. Existem duas formas para avaliação destes resultados: a micro média, onde as categorias contam proporcionalmente a quantidade de exemplos positivos de treinamento, e a macro média, onde todos as categorias contam da mesma forma. Estas formas cálculo estão representadas na Tabela 15, a seguir:

Tabela 15. Fórmulas Micro-Média e Macro-Média.

	Micro-Média	**Macro-Média**
Precisão (π)	$\boldsymbol{\pi = \frac{\sum_{i=1}^{\lvert C\rvert} TP_i}{\sum_{i=1}^{\lvert C\rvert} TP_i + FP_i}}$	$\boldsymbol{\pi = \frac{\sum_{i=1}^{\lvert C\rvert} \pi_i}{\lvert C\rvert} = \frac{\sum_{i=1}^{\lvert C\rvert} \frac{TP_i}{TP_i + FP_i}}{\lvert C\rvert}}$
Abrangência (ρ)	$\boldsymbol{\rho = \frac{\sum_{i=1}^{\lvert C\rvert} TP_i}{\sum_{i=1}^{\lvert C\rvert} TP_i + FN_i}}$	$\boldsymbol{\pi = \frac{\sum_{i=1}^{\lvert C\rvert} \rho_i}{\lvert C\rvert} = \frac{\sum_{i=1}^{\lvert C\rvert} \frac{TP_i}{TP_i + FN_i}}{\lvert C\rvert}}$

Uma vez que a maioria dos classificadores pode ser arbitrariamente ajustada para enfatizar tanto a precisão quanto a abrangência, apenas a combinação entre estas medidas pode ser significativa. O caminho mais popular para esta combinação é a função $F\beta = \frac{(\beta^2+1)\pi\rho}{\beta^2\pi+\rho}$, para $0 \leq \beta \leq \infty$. Em geral, β é mantido com o valor 1, o que transforma a função $F\beta$ em $F_1 = \frac{2\,\pi\rho}{\pi+\rho}$, isto é, a média harmônica entre a precisão e a abrangência.

5.4 Representação dos Documentos

Nas abordagens convencionais, documentos são representados como vetores de palavras, denominados "*bag-of-words*", como representado na Figura 20.

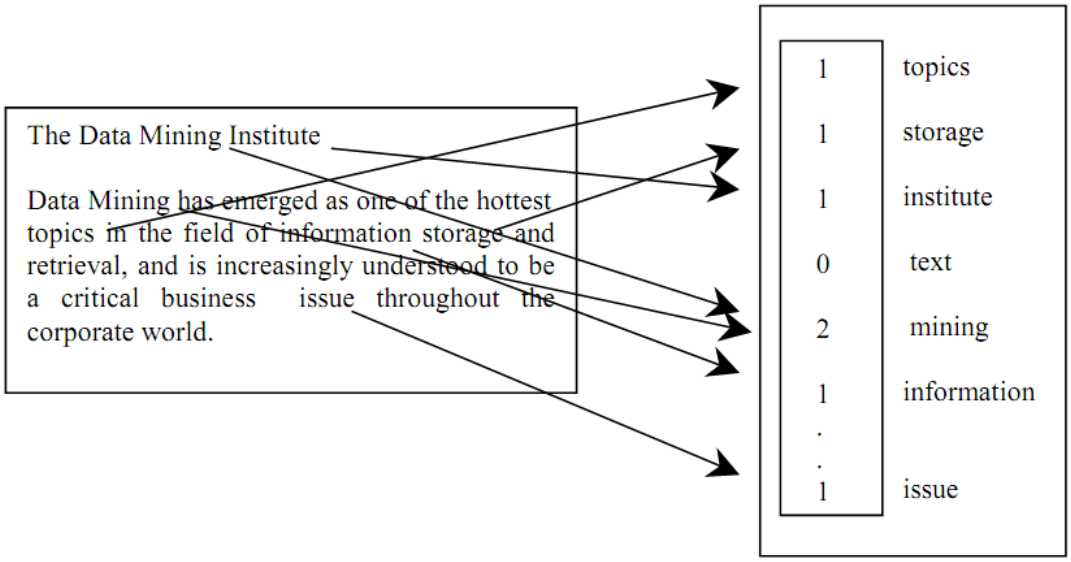


Figura 20. Vetor de Palavras.

A partir de um dicionário previamente confeccionado, os documentos são transformados e representados como vetores, contendo valores binários indicando a presença ou ausência de cada palavra ou números, onde cada número corresponde a frequência, normalizada ou não, da ocorrência de cada palavra no documento.

Como já mencionado, tal representação apresenta dois problemas graves: a alta dimensionalidade de características, já que o tamanho do vetor é proporcional ao tamanho do dicionário, e a perda da informação contida nos textos originais, pois as relações mantidas entre as palavras não são capturadas.

A representação utilizada neste experimento é uma adaptação desta abordagem. O dicionário é composto pelos grânulos da coleção. Os vetores são construídos com base na proporção de palavras de cada grânulo em cada um dos documentos. Tal proporção é definida pela divisão da quantidade de palavras no documento e no grânulo pela quantidade de palavras no grânulo. A Figura 21 apresenta a ideia.

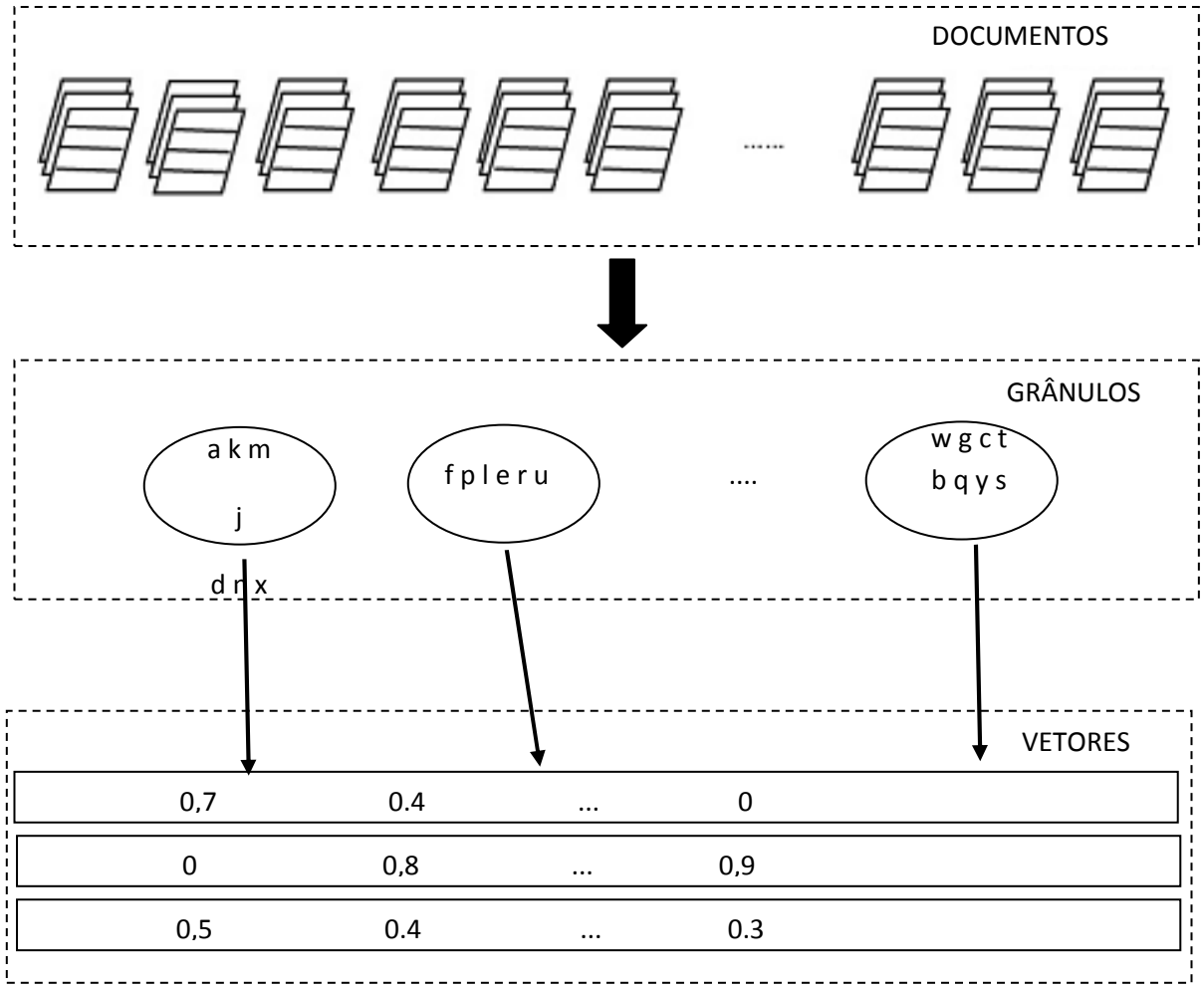


Figura 21. Vetor de Grânulos.

5.5 Experimento

Para o experimento foram utilizadas as mesmas coleções de documentos descritas no Capítulo 4. Cada uma destas bases foi submetida aos procedimentos de pré-processamento, descritos na seção 4.3.2 e ao algoritmo de agrupamento espectral apresentado na seção 4.3.3. De cada coleção, 70% dos documentos foram, aleatoriamente, selecionados para treinamento e 30% foram selecionados para teste. Em uma etapa inicial, a quantidade de grânulos gerada foi definida de forma proporcional a quantidade de documentos em cada coleção. De modo geral, foi possível observar que os valores máximos para micro-média F1 e macro-média F1 eram atingidos quando esta

quantidade girava em torno de 10% da quantidade de documentos em cada coleção. Sendo assim, foi escolhido o percentual de 10% para a proporção de grânulos em relação quantidade de documentos na coleção para análise final. A Tabela 16 apresenta estes números.

Tabela 16. Quantidade de Grânulos por Coleção.

Coleção	Quantidade de Documentos	Quantidade de Grânulos
1	200	20
2	160	16
3	5.485	548
4	2.500	250

Ao final da construção dos dicionários de grânulos, os vetores representantes de cada coleção foram construídos conforme definido na seção 5.5. Cada um dos conjuntos de vetores foi submetido aos algoritmos de classificação apresentados na seção 5.3 e os resultados avaliados conforme os critérios de avaliação apresentados na seção 5.4. Procedimentos semelhantes foram adotados para criação dos vetores construídos a partir das técnicas LSA e LDA, também submetidos aos mesmos algoritmos. Os resultados alcançados estão organizados em tabelas na seção 5.7.

5.6 Resultados

Para cada uma das coleções, representadas a partir de cada um dos modelos de documentos e submetidas a cada um dos classificadores usados no experimento, foram calculados os valores micro-média F1 e macro-média F1 como definidos na seção 5.4.

5.6.1 Micro F1 – Algoritmo KNN

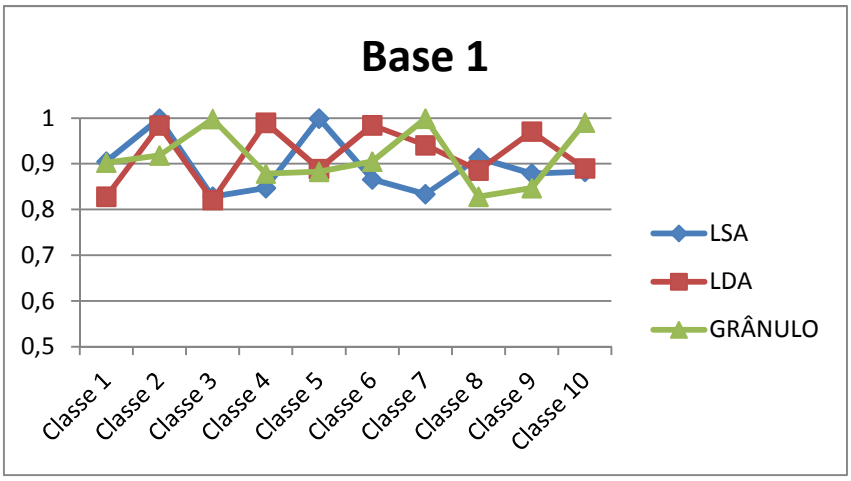


Figura 22. Resultados Micro-F1 para Algoritmo K-NN sobre Base 1.

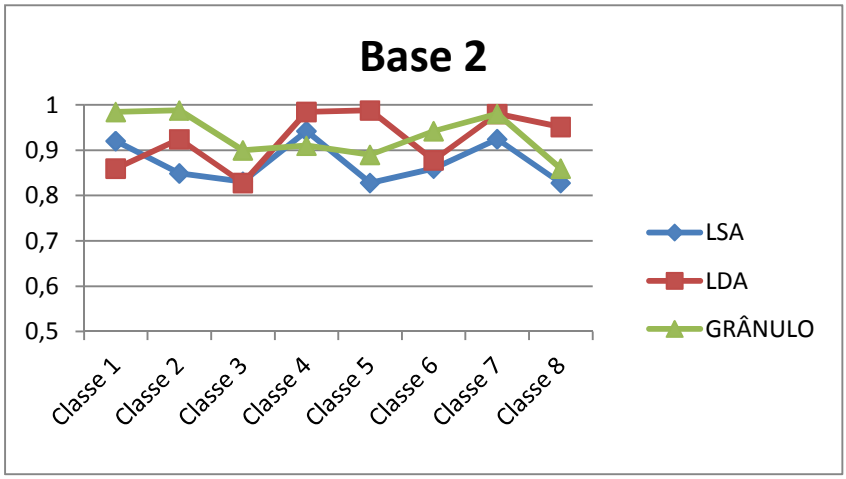


Figura 23. Resultados Micro-F1 para Algoritmo K-NN sobre Base 2.

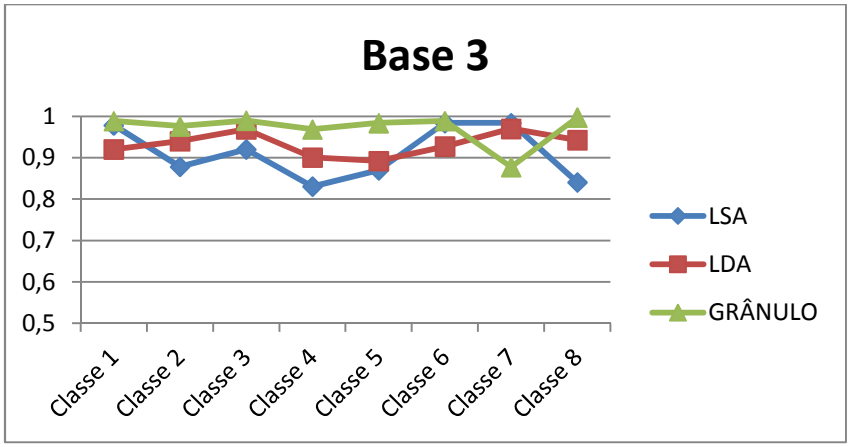


Figura 24. Resultados Micro-F1 para Algoritmo K-NN sobre Base 3.

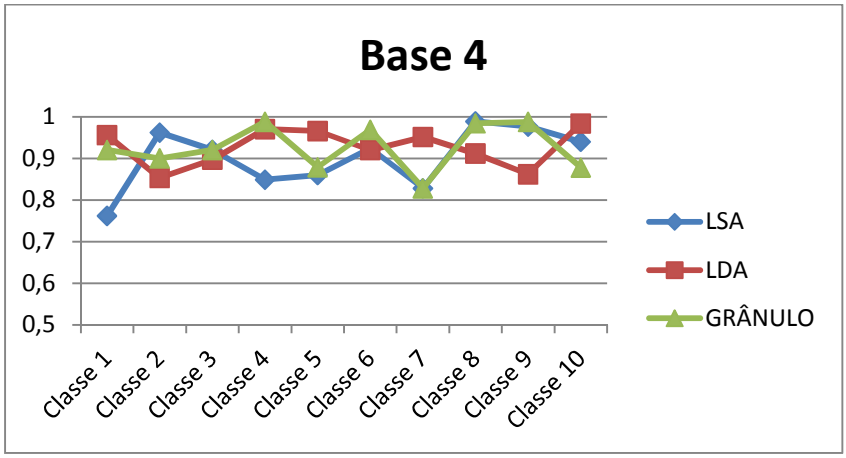


Figura 25. Resultados Micro-F1 para Algoritmo K-NN sobre Base 4.

5.6.2 Macro F1 – Algoritmo KNN

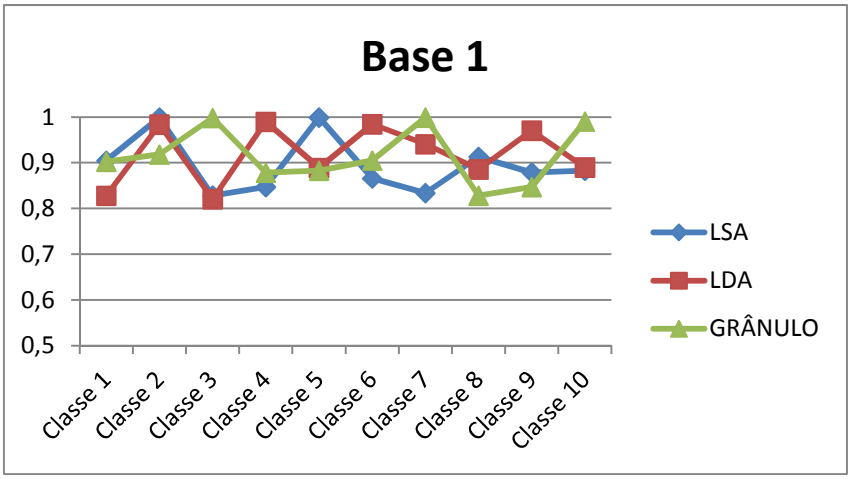


Figura 26. Resultados Macro-F1 para Algoritmo K-NN sobre Base 1.

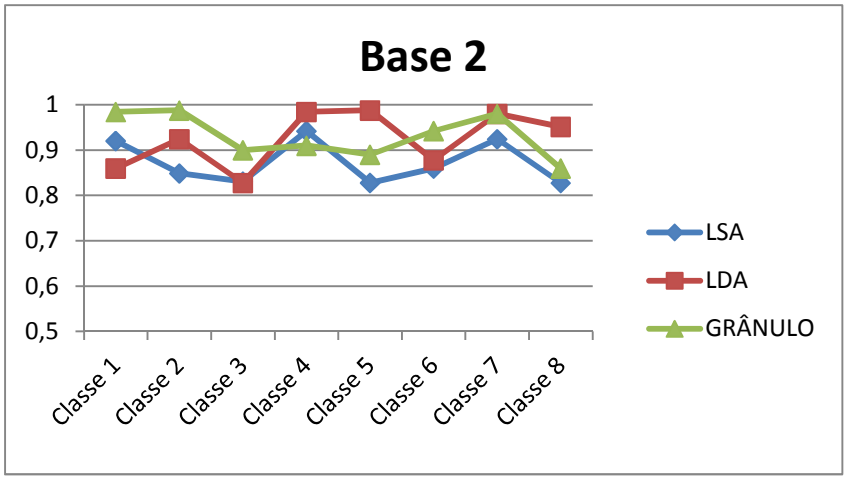


Figura 27. Resultados Macro-F1 para Algoritmo K-NN sobre Base 2.

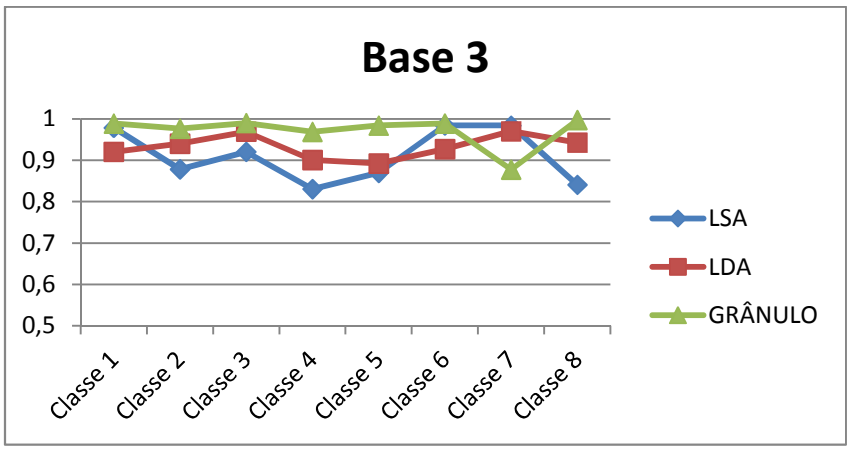


Figura 28. Resultados Macro-F1 para Algoritmo K-NN sobre Base 3.

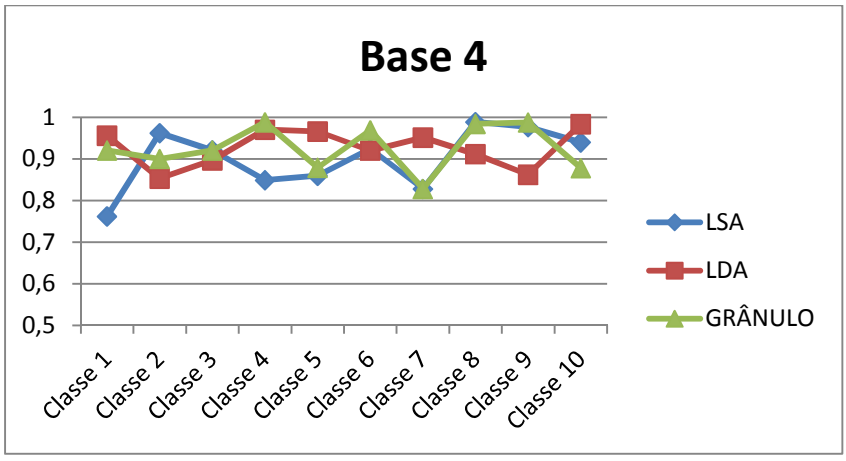


Figura 29. Resultados Macro-F1 para Algoritmo K-NN sobre Base 4.

5.6.3 Micro F1 – Algoritmo Naive Bayes

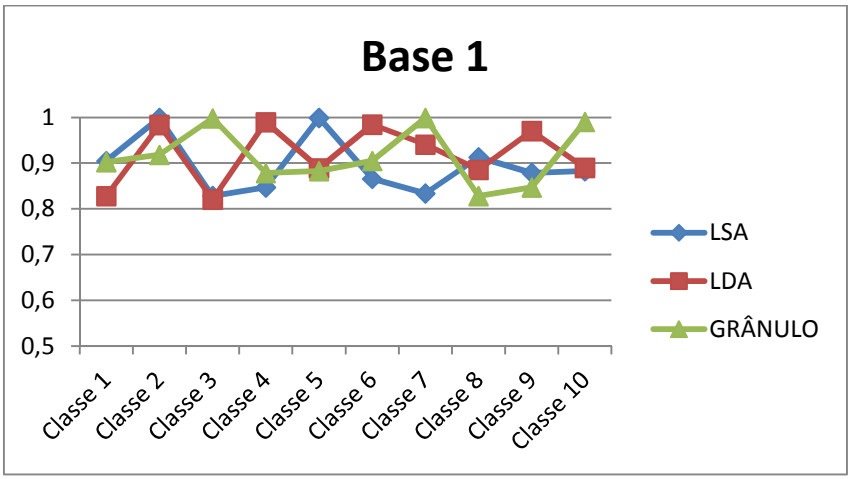


Figura 30. Resultados Micro-F1 para Algoritmo Naive Bayes sobre Base 1.

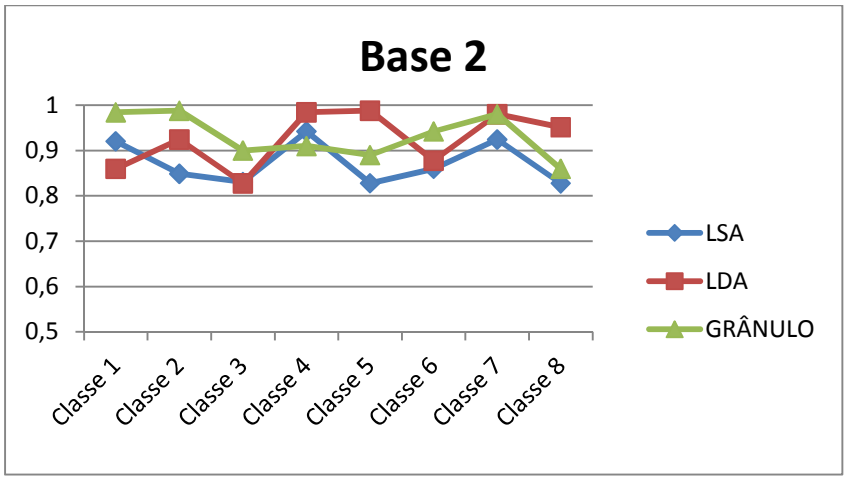


Figura 31. Resultados Micro-F1 para Algoritmo Naive Bayes sobre Base 2.

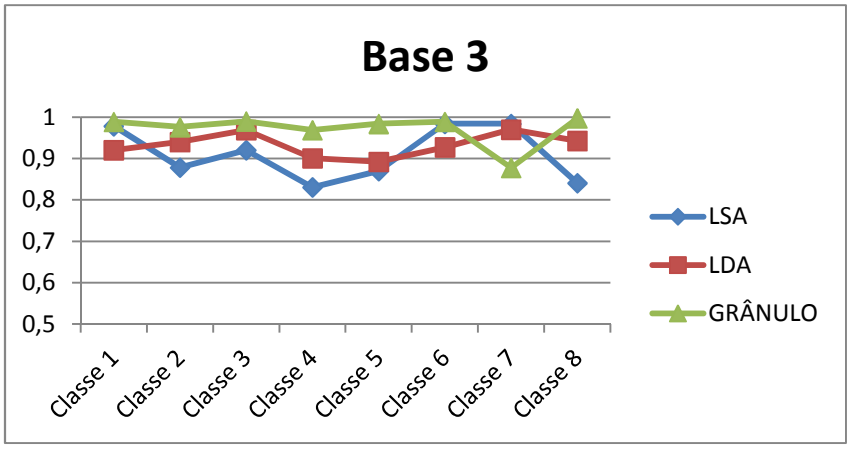


Figura 32. Resultados Micro-F1 para Algoritmo Naive Bayes sobre Base 3.

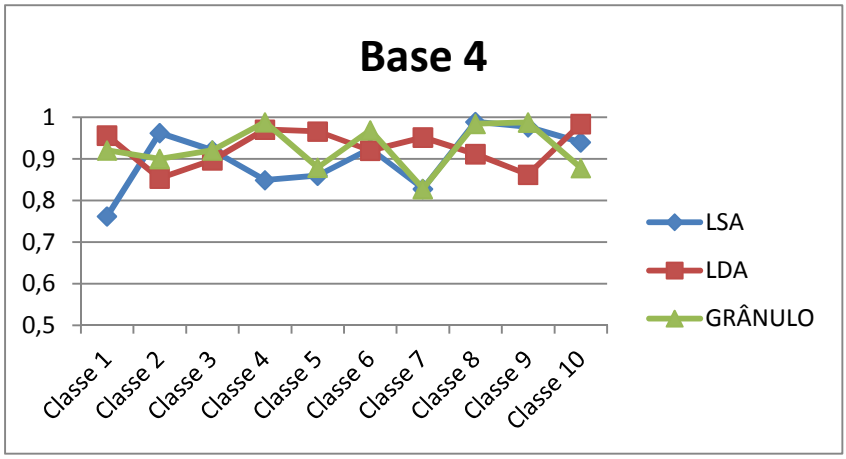


Figura 33. Resultados Micro-F1 para Algoritmo Naive Bayes sobre Base 4.

5.6.4 Macro F1 – Algoritmo Naive Bayes

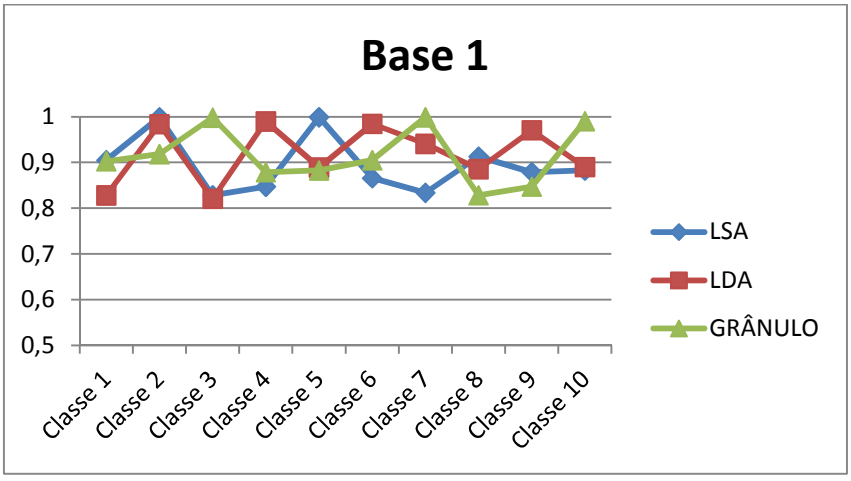


Figura 34. Resultados Macro-F1 para Algoritmo Naive Bayes sobre Base 1.

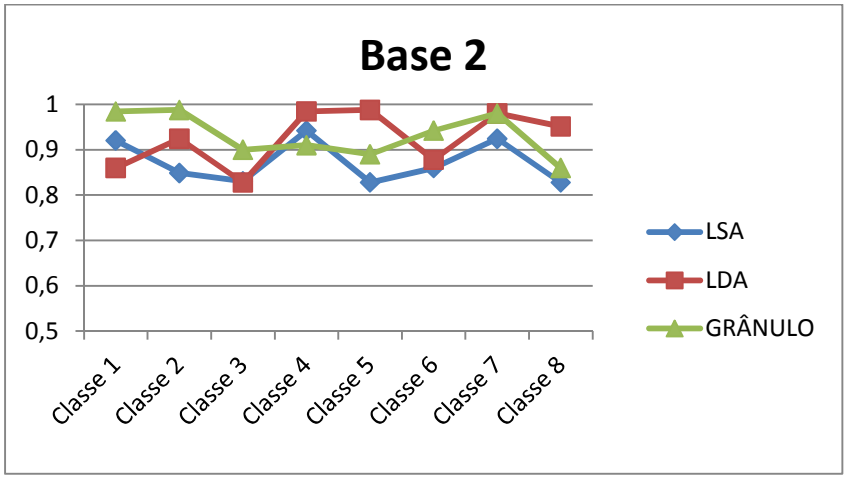


Figura 35. Resultados Macro-F1 para Algoritmo Naive Bayes sobre Base 2.

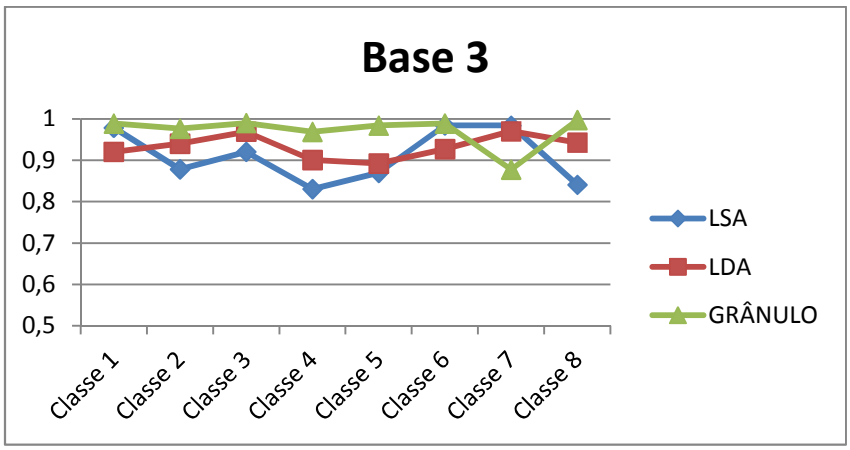


Figura 36. Resultados Macro-F1 para Algoritmo Naive Bayes sobre Base 3.

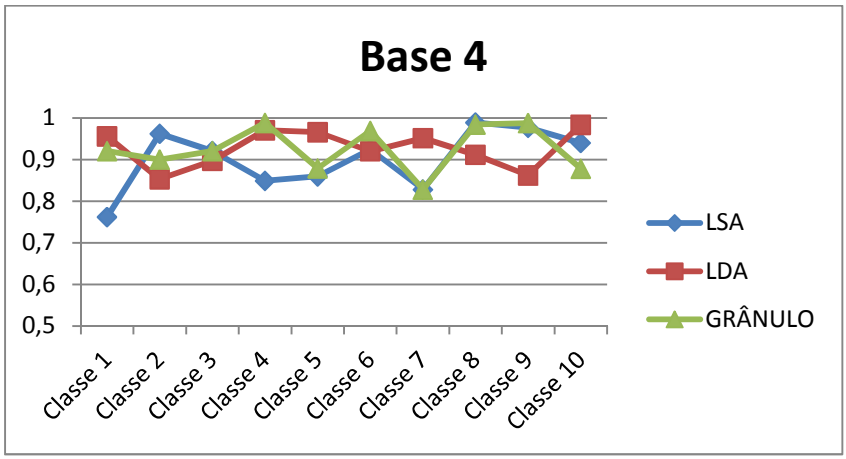


Figura 37. Resultados Macro-F1 para Algoritmo Naive Bayes sobre Base 4.

5.6.5 Micro F1 – Máquinas de Vetores Suporte

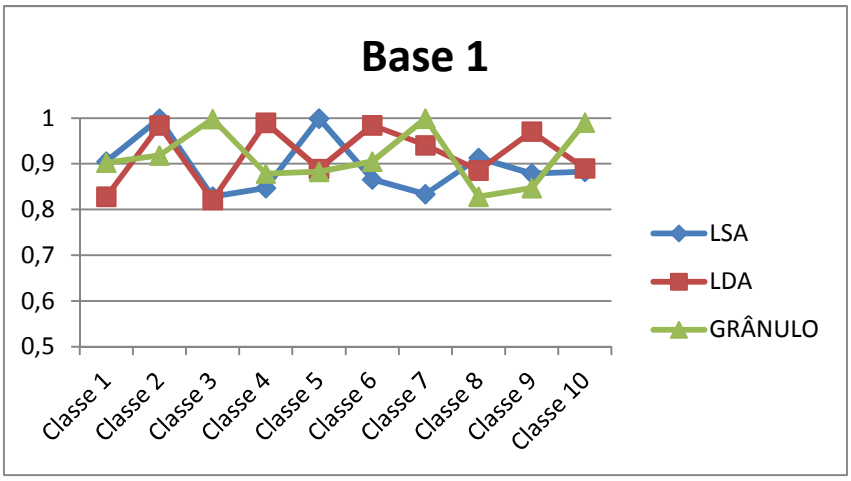


Figura 38. Resultados Micro-F1 para Algoritmo SVM sobre Base 1.

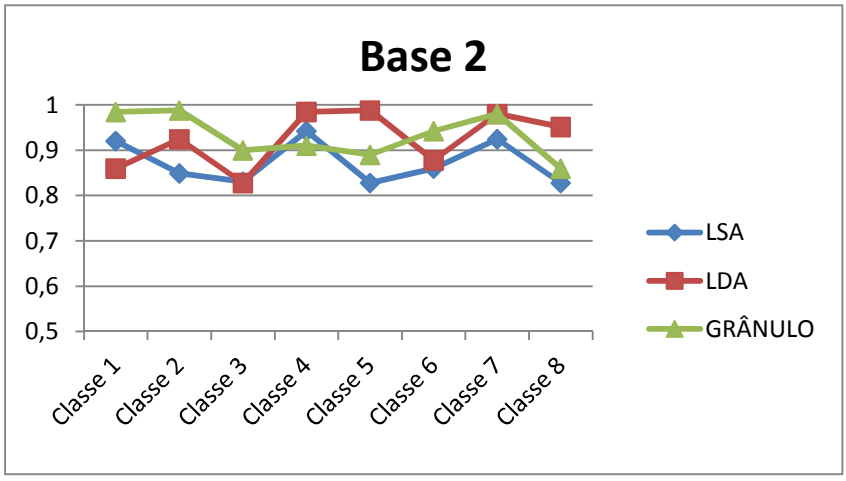


Figura 39. Resultados Micro-F1 para Algoritmo SVM sobre Base 2.

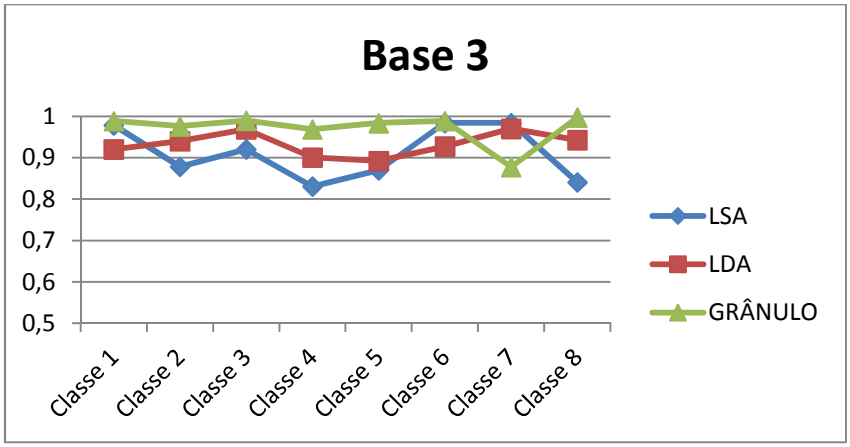


Figura 40. Resultados Micro-F1 para Algoritmo SVM sobre Base 3.

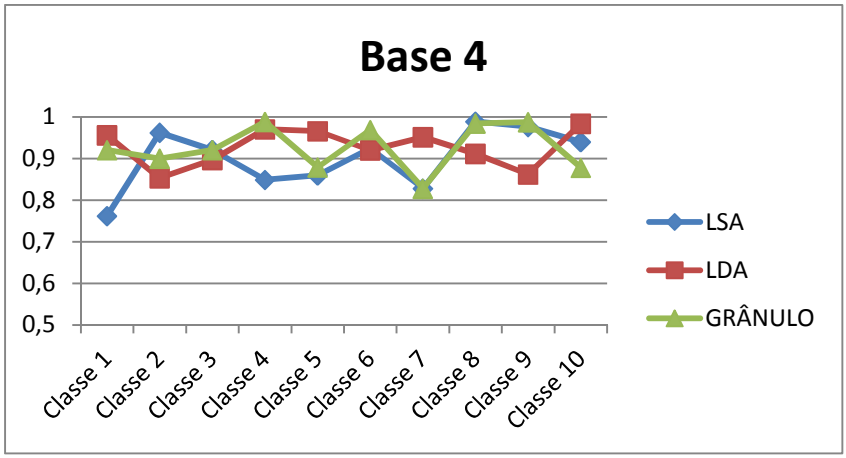


Figura 41. Resultados Micro-F1 para Algoritmo SVM sobre Base 4.

5.6.6 Macro F1 – Máquinas de Vetores Suporte

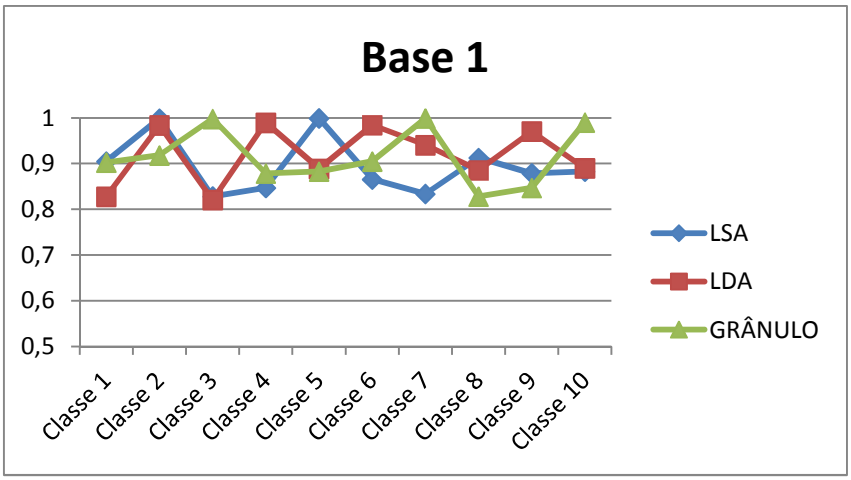


Figura 42. Resultados Macro-F1 para Algoritmo SVM sobre Base 1.

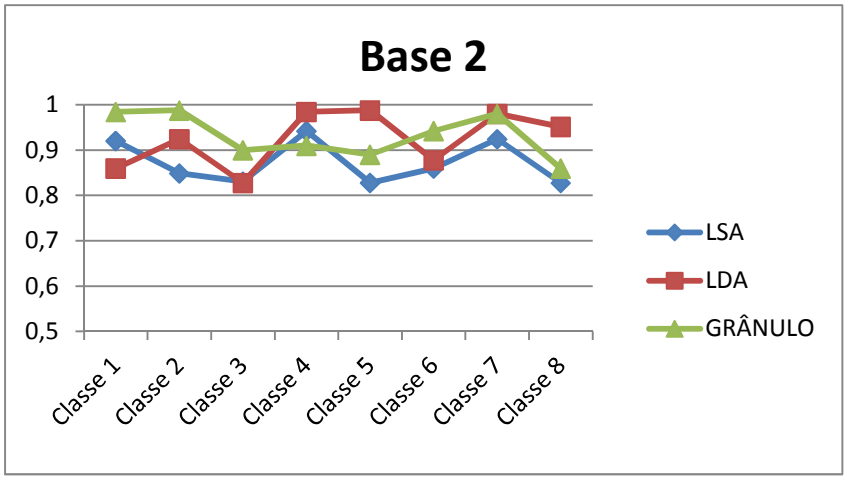


Figura 43. Resultados Macro-F1 para Algoritmo SVM sobre Base 2.

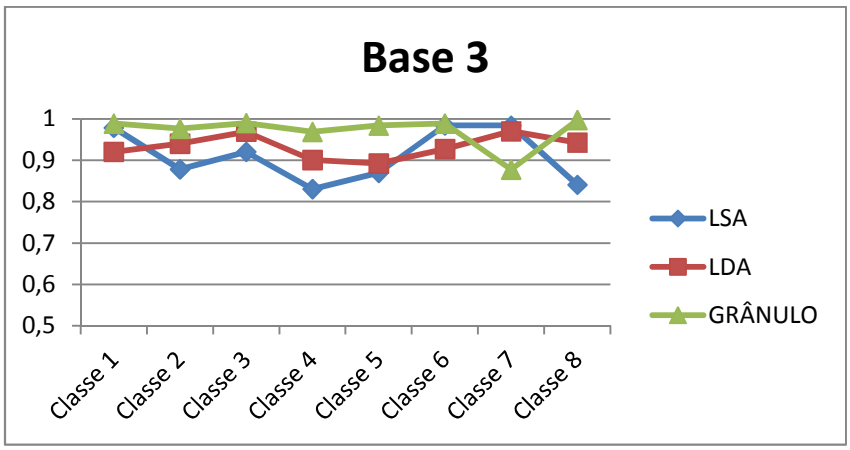


Figura 44. Resultados Macro-F1 para Algoritmo SVM sobre Base 3.

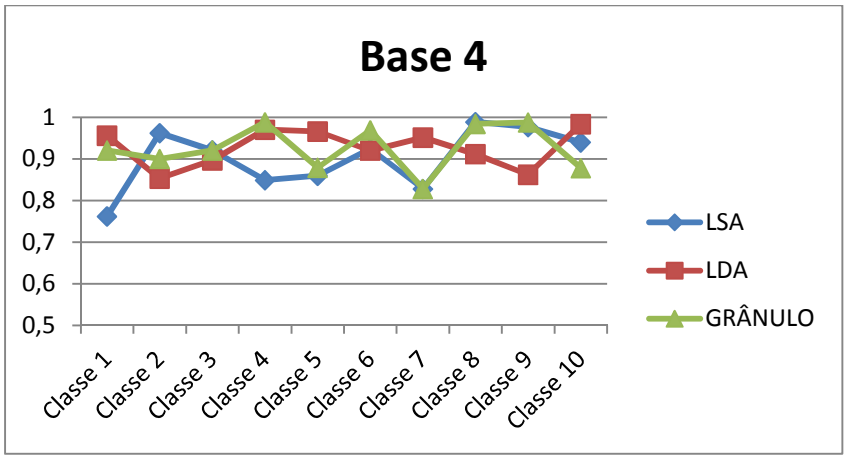


Figura 45. Resultados Macro-F1 para Algoritmo SVM sobre Base 4.

5.6.7 Micro F1 – Comitês de Classificadores

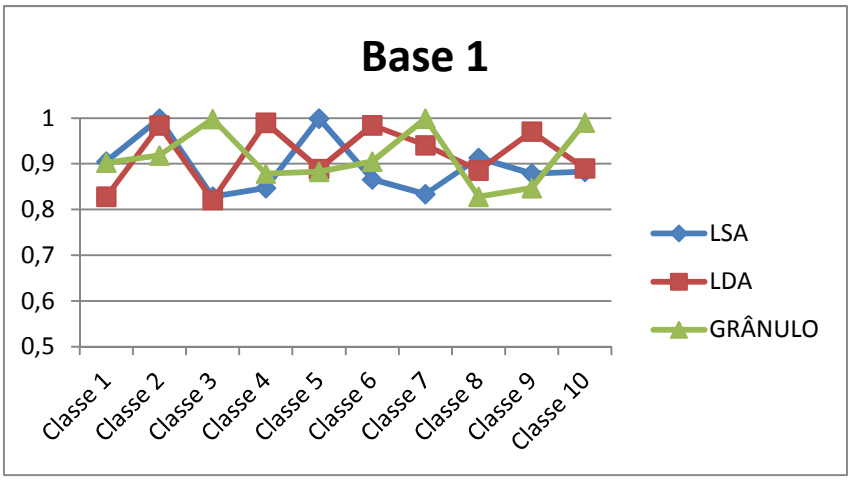


Figura 46. Resultados Micro-F1 para Algoritmo Boosting sobre Base 1.

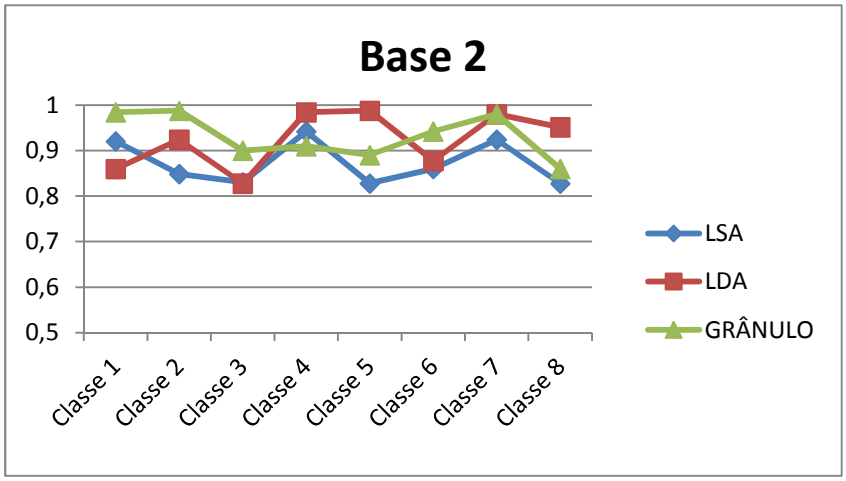


Figura 47. Resultados Micro-F1 para Algoritmo Boosting sobre Base 2.

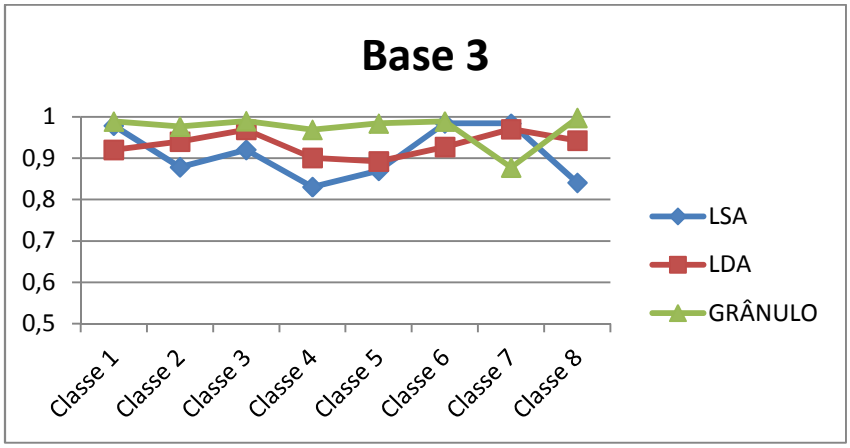


Figura 48. Resultados Micro-F1 para Algoritmo Boosting sobre Base 3.

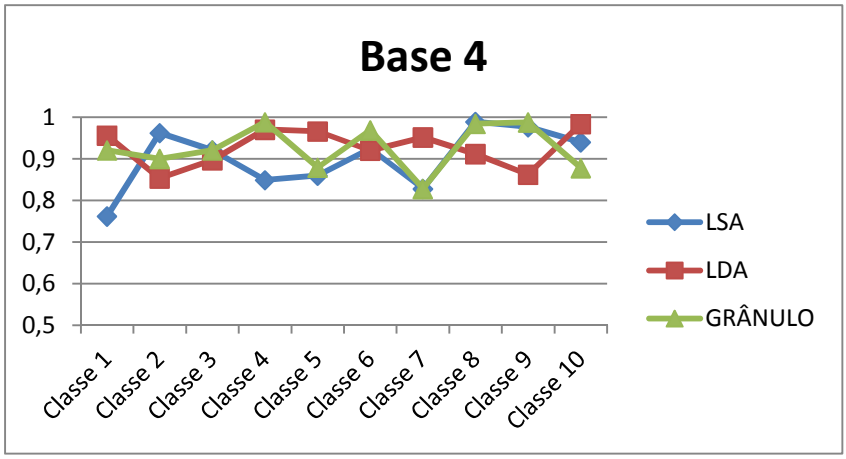


Figura 49. Resultados Micro-F1 para Algoritmo Boosting sobre Base 4.

5.6.8 Macro F1 – Comitês de Classificadores

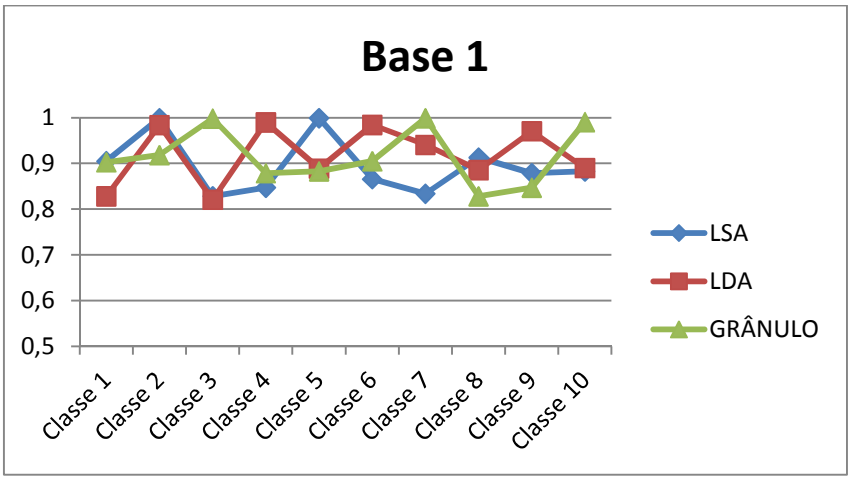


Figura 50. Resultados Macro-F1 para Algoritmo Boosting sobre Base 1.

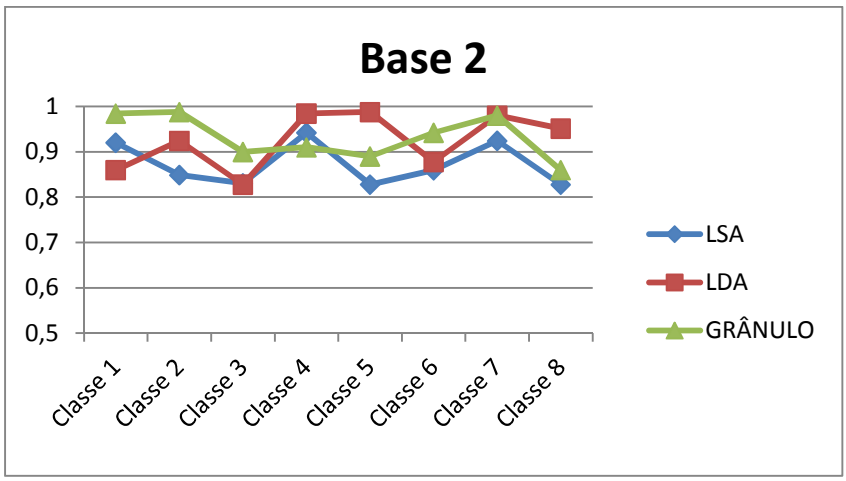


Figura 51. Resultados Macro-F1 para Algoritmo Boosting sobre Base 2.

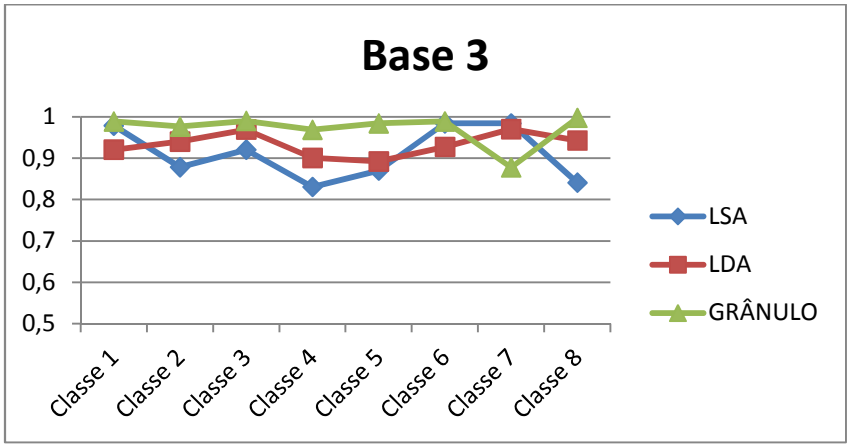


Figura 52. Resultados Macro-F1 para Algoritmo Boosting sobre Base 3.

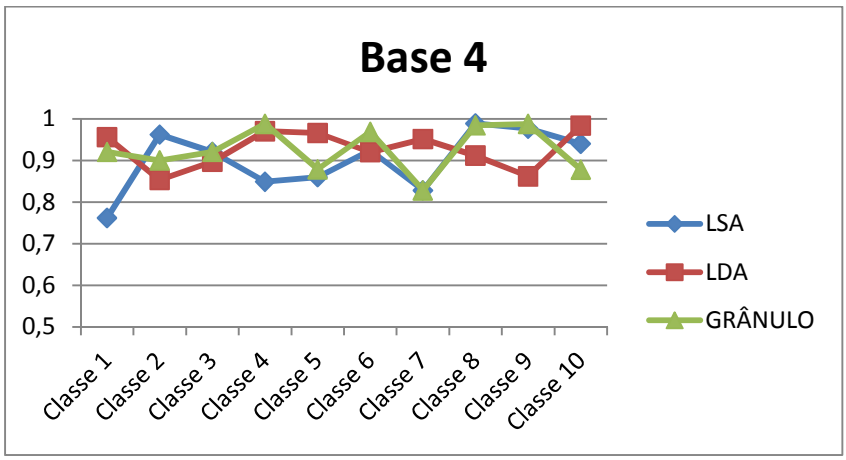


Figura 53. Resultados Macro-F1 para Algoritmo Boosting sobre Base 4.

5.6.9 Avaliação dos Resultados

Avaliar os resultados alcançados em tarefas de classificação de documentos é uma tarefa bastante difícil. Estes resultados estão sempre vinculados a fatores que transformam estes resultados a cada vez que são alterados. Alguns exemplos destes fatores são a quantidade de grânulos escolhida, a quantidade de documentos em cada coleção, o percentual separado para treinamento em teste, bem como os classificadores escolhidos para a avaliação.

Neste trabalho foram adotadas técnicas, tanto para o experimento realizado quanto para os critérios de avaliação, que vêm sendo utilizadas na maioria dos trabalhos

publicados nesta área. Pode-se afirmar, apenas com base nos critérios de avaliação adotados, que os resultados alcançados são bastante coerentes com os resultados apresentados nestes trabalhos, apesar da existência de diferença entre os fatores anteriormente relacionados. Como consequência, cabe a afirmativa de que o modelo de documentos baseado em grânulos, proposto neste trabalho, pode representar uma alternativa aos modelos que buscam identificar as relações entre as palavras contidas nos documentos com o objetivo de enriquecer a sua representação.

Analisando os gráficos apresentados na seção 5.7 podemos observar que existe coerência no comportamento dos valores alcançados pelos critérios de avaliação para todas as coleções e para todos os classificadores adotados. Um destaque especial pode ser dado ao desempenho da técnica de classificação baseada em comitê, mesmo considerando que este melhor desempenho esteja vinculado ao já conhecido bom desempenho da técnica de maneira geral.

5.8 Considerações Finais

Este capítulo apresentou a aplicação dos grânulos de palavras na representação de documentos. Mostrou a sua influência sobre os resultados na classificação destes documentos e comparou estes resultados com a técnicas Análise de Semântica Latente (LSA) e Latent Dirichlet Allocation (LDA), comumente usadas com o objetivo criar modelos capazes de representar as relações entre as palavras contidas nestes documentos. O capítulo seguinte apresenta um estudo semelhante a este voltado para a tarefa de agrupamento de documentos.

6 AGRUPAMENTO BASEADO EM GRÂNULOS

Segundo (AGGARWAL, 2012), agrupamento é a tarefa de encontrar objetos similares em um conjunto de dados. A similaridade entre os objetos é medida através de uma função de similaridade. O problema de agrupamento é muito útil no domínio de dados do tipo texto, especialmente para a organização de documentos com o objetivo de improvisar a recuperação e como suporte ao browsing. Assim como na tarefa de classificação de textos, a representação dos documentos, e o esquema de redução de características são determinantes na eficiência dos algoritmos de agrupamento. Este capítulo apresenta a aplicação dos grânulos de palavras na representação destes documentos, a sua influência sobre os resultados do agrupamento destes documentos e a comparação destes resultados com a técnicas Análise de Semântica Latente (LSA) e Latent Dirichlet Allocation (LDA).

6.1 Introdução e Trabalhos Relacionados

Análise de grupos (clusters) é um método de aprendizado não supervisionado que tem por objetivo agrupar objetos em diferentes subconjuntos, ou clusters, de modo que cada subconjunto contenha objetos similares de acordo com algum critério pré-definido. A tarefa de agrupamento tem sido assunto de intensivo estudo em diferentes campos incluindo estatística, aprendizado de máquina, mineração de dados, processamento de imagens e recuperação de informação.

Da mesma forma como ocorre em tarefas de classificação, como apresentado no Capítulo 5, aplicações que envolvam agrupamento são extremamente afetadas pela representação dos documentos da coleção. Assim, como nos trabalhos que tratam de classificação, para lidar com a incapacidade do modelo Bag of Words em representar as relações entre as palavras, muitos trabalhos têm sugerido o emprego de LSA e LDA como soluções para a melhoria dos resultados na tarefa agrupamento. (SONG, 2007), (KIM, 2010) e (WANG, 2011) empregam LSA na representação dos documentos e os submetem a variados algoritmos de agrupamento. (SONG, 2007) utiliza o algoritmo K-means e algoritmos genéticos, (KIM, 2010) emprega agrupamento hierárquico e (WANG, 2011) avalia o resultado da representação reduzida com o uso de algoritmos representativos das variadas abordagens de agrupamento de dados. (WEI, 2006),

(BANERJEE, 2007) e (MILLAR, 2009) empregam LDA na representação dos documentos e, da mesma forma que os anteriores, os submetem a variados algoritmos de agrupamento. (WEI, 2006) e (BANERJEE, 2007) utilizam K-means em seus experimentos, (MILLAR, 2009) emprega mapas auto-organizáveis de Kohonem para construção dos grupos. Todos os trabalhos mostram ganho no desempenho dos algoritmos, tanto em qualidade dos grupos gerados, quanto na velocidade de computação destes grupos.

6.2 Agrupamento

Existem basicamente três abordagens para o problema do agrupamento: a abordagem baseada em similaridade, a abordagem baseada em projeção e a abordagem baseada em densidade (GHOSH, 2003) (ZHONG, 2005). As técnicas de agrupamento baseadas em similaridade constroem k partições dos dados, onde cada partição otimiza um critério de agrupamento baseado em uma medida de similaridade. As abordagens baseadas em projeção, em geral, fazem uso do conjunto de autovalores de uma matriz de similaridade entre os dados para efetuar uma redução de dimensionalidade para que o agrupamento ocorra em uma quantidade menor de dimensões. As técnicas de agrupamento baseadas em densidade estimam a distribuição de probabilidade dos grupos e associam cada instância ao grupo mais provável. Neste trabalho, estaremos interessados no emprego de algoritmos que se baseiam na primeira abordagem, dada a sua simplicidade e facilidade de análise nos resultados.

6.2.1 Agrupamento Baseado em Similaridade

Algoritmos de agrupamento baseados em similaridade particionam os dados com base no quanto similares as instâncias são. As mais similares pertencem ao mesmo grupo. O processo de agrupamento opera sobre a similaridade entre as instâncias de acordo com uma dada medida de distância. Assim, técnicas de agrupamento baseadas em similaridade buscam identificar os grupos que maximizam a distância entre os grupos e minimizam a distância entre os objetos do grupo. O objetivo é obter grupos distintos de entidades similares.

A distância euclidiana é uma das medidas mais comuns de similaridade. Dois pontos são vistos como vetores em um espaço vetorial e a distância entre eles é representada pelo tamanho da reta que os une.

$$d_2(x_i, x_j) = \sqrt{\sum_{k=1}^{M} (x_{k,i} , x_{k,j})2} = \| x_i - x_j \|^2 \qquad (6.1)$$

Esta métrica é um caso especial da distância de Minkowski $d_p(x_i,x_j) = \sum_{k=1}^{M}(x_{k,i} , x_{k,j})2)^{1/p}$ para p =2. A distância euclidiana apresenta bons resultados quando os dados são compactos ou estão em grupos isolados . A desvantagem do uso direto da distância de Minkowski é a tendência de características com maiores escalas dominarem as demais como é, em geral, o caso do problema de agrupamento de documentos. Soluções para este problema incluem a normalização de valores contínuos ou outro esquema de pesagem semelhante a medida do cosseno que expressa a similaridade entre objetos de acordo com o ângulo formado entre os vetores que os representam.

$$\cos(x_i,x_j) = \frac{x_i . x_j}{||x_i||||x_j||} = \frac{\sum_{k=1}^{M} x_{k,i} \times x_{k,j}}{\sqrt{\sum_{k=1}^{M} x_{k,i}^2 \sum_{k=1}^{M} x_{k,j}^2}} \qquad (6.2)$$

A correlação linear entre as características pode distorcer as medidas de distância. Esta distorção pode ser diminuída pela aplicação de uma transformação nos dados ou pelo uso da distância de Mahalanobis.

$$D_M(x_i,x_j) = (x_i - x_j)^T \sum_{=1} (x_i - x_j) \qquad (6.3)$$

onde, $\sum$ é a matriz de covariância de exemplos. A distância de Mahalanobis associa diferentes pesos a diferentes características com base suas variâncias e na correlação linear entre os pares.

Em muitas abordagens de agrupamento de documentos, as instâncias são representadas através do modelo espaço vetorial (SALTON, 1975), onde o peso de cada termo específico no vetor é produto de parâmetros locais e gerais. Neste caso, o vetor de pesos D_i para um documento d_i é $D_i = < x_{1,i}, ..., x_{M,i} >$ onde

$$X_{i,j} - tf_{tj,di} \times idf_{tj} \quad (6.4)$$

onde $tf_{tj,di}$ representa a frequência do termo t_j no documento d_i e idf_i representa a frequência inversa do termo t_j.

6.2.1.1 Agrupamento Aglomerativo

Agrupamento aglomerativo é um tipo de agrupamento hierárquico que inicia com a associação dos documentos a diferentes grupos. A partir desta associação inicial, um par de grupos ou clusters cuja similaridade seja máxima é associado a cada iteração. Este processo é repetido até que um dendograma é obtido. Um exemplo simples é mostrado na Figura 46. Existem vários critérios para se definir um valor representativo da similaridade ou dissimilaridade entre as partições P_1 e P_2 . Como medida de dissimilaridade, a distância máxima ou completa ligação de dois grupos é decidida por um valor máximo de distâncias de pares de distâncias entre os documentos, dis (a,b), a$\in$ P_1 e b $\in$ P_2. A distância mínima ou a ligação simples do agrupamento é, em contra partida, selecionada pelo valor mínimo destas distâncias. A distância média, também conhecida como ligação média é dada pelo valor médio de todos os pares de distâncias. Um destes critérios é escolhido para avaliar todas as dissimilaridades de todos os pares de grupos. Finalmente, dois grupos com mínima dissimilaridade ou máxima similaridade são fundidos em um único grupo. O procedimento aglomerativo continua até que uma determinada quantidade de grupos seja alcançada. Uma medidade de similaridade entre documentos pode ser igualmente aplicada a estes critérios ao invés de uma medidade de distância. Podem existir muitas variações do agrupamento aglomerativo pois estas capacidades fornecem uma estrutura para as alaterações nas habilidades do agrupamento. (TISHBY, 1999) apresenta o método de gargalo de informação (bottleneck method) para agrupamento de documentos que utiliza informação mútua como medida de compartilhamento de informação entre dois grupos.

Mesmo se a medida e o processo são diferentes dos listados acima, o objetivo é sempre iniciar com a associação dos documentos aos grupos e aglomerá-los durante as iterações.

O principal problema do agrupamento aglomerativo é a complexidade do algoritmos que é pelo menos $O(N^2)$, onde N é o número de documentos. Este problema pode ser crítico para as atuais aplicações de agrupamento em larga escala.

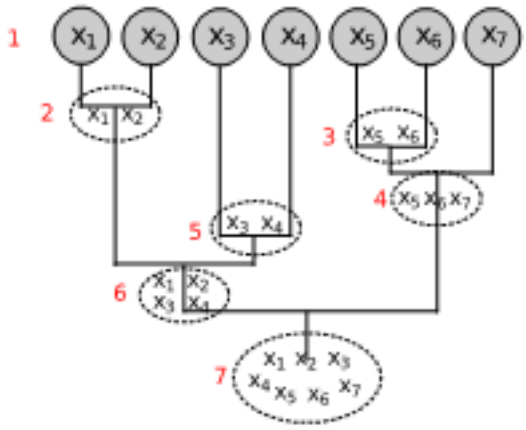


Figura 54. Agrupamento Aglomerativo

6.2.1.2 Agrupamento por Particionamento

Os algoritmos de agrupamento por particionamento obtêm um particionamento simples dos dados ao invés de uma estrutura de grupos. São vantajosos em aplicações com grande quantidade de dados onde a construção de um dendograma é computacionalmente proibitiva. A desvantagem é a necessidade da escolha de uma quantidade de grupos. A técnica por particionamento, em geral, produz os grupos através da otimização de uma função definida localmente (sobre um subconjunto de padrões) ou globalmente (definida sobre todos os padrões). A busca combinatória para o valor ótimo da critério de avaliação é proibitiva em termos computacionais. O que ocorre, na prática, é a execução do algoritmo por diversas iterações com diferentes estados iniciais. A melhor configuração, obtida a partir destas iterações, é usada como o

agrupamento de saída. O critério mais intuitivo e empregado com maior frequência nesta técnica é o critério erro quadrático, que tende a trabalhar bem com grupos compactos e isolados. O erro quadrático para o agrupamento de uma coleção contendo k grupos ou clusters é

$$\sum_{x_i \in P_c} ||x_i - \mu_c||^2 \quad (6.5)$$

onde x_i é o vetor que representa a instância x_i e μ_c é o centroide do c$^{(th)}$ grupo, P_c.

O algoritmo mais simples e mais comum que emprega o critério do erro quadrático é o algoritmo *k-means* (MACQUEEN, 1967). Ele inicia com uma partição inicial randômica e permanece redistribuindo os padrões dos grupos com base na similaridade entre os padrões e o centro do grupo até que o critério de convergência seja alcançado. A Figura 55 ilustra os resultados do agrupamento encontrados pelo algoritmo K-Means para um mesmo problema e diferentes valores de K.

O algoritmo K-Means é popular porque é fácil de implementar e possui complexidade O(N). O maior problema é ser sensível a seleção do particionamento inicial podendo convergir para um mínimo local da função de avaliação no caso deste particionamento inicial não ter sido apropriadamente construído.

Devido a sua simplicidade e bom desempenho em diversas áreas de aplicação, o algoritmo K-Means é amplamente utilizado para a tarefa de agrupamento. Recentemente, algumas variações deste algoritmo foram apresentadas em alguns trabalhos, tais como, K-Means Bi-Seção (STEINBACH, 2000), K-Means Esférico (DHILLON, 2001) e K-MeansEspectral (ZHA, 2001).

A alta dimensionalidade tende a ser um desafio especial aos algoritmos baseados em similaridade devido a dispersão dos dados. Em um espaço de altas dimensões todos os pares de pontos tendem a ser equidistantes uns dos outros o que torna irreal a similaridade ou dissimilaridade nos grupos.

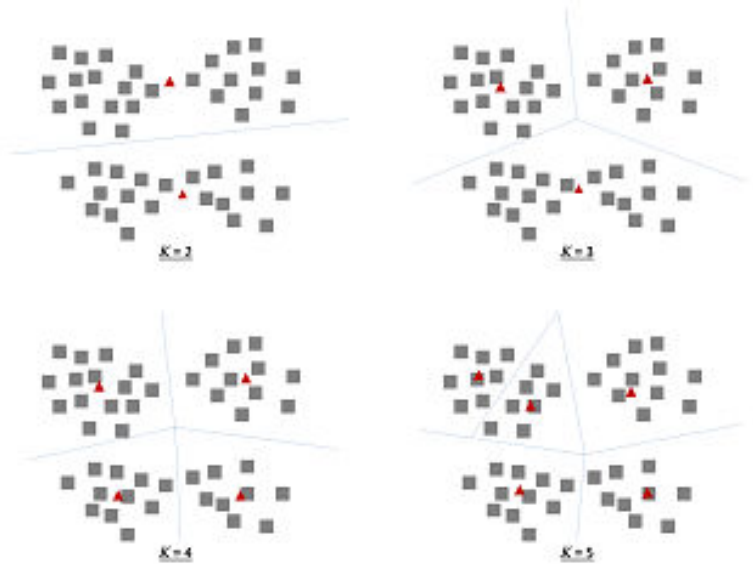


Figura 55. Agrupamento por Particionamento

6.3 Avaliação do Agrupamento

A escolha de uma métrica para a avaliação da qualidade do agrupamento depende da aplicação (GRABMEIER, 2002). No contexto do agrupamento de documentos, várias opções estão disponíveis. Uma delas é comparar o resultado do agrupamento com uma solução já existente, preparada manualmente por especialistas. Esta abordagem é conhecida como validação externa do grupo (JAIN, 1999). Podemos medir a qualidade da separação dos grupos e o quanto compactos eles são. Esta abordagem é a validação ou validade interna do grupo (JAIN, 1999). Também podemos avaliar o agrupamento em conjunto com alguns especialistas, fornecendo a mesma tarefa a ser exercutada e comparando os resultados. Como diferentes organizações dos documentos são possíveis, especialistas diferentes podem escolher caminhos distintos conferindo subjetividade à análise do resultado. Por esta dificuldade, esta abordagem é considerada impraticável por muitos pesquisadores.

As medidas de qualidade interna dos grupos consideram aspectos estruturais destes grupos tais como o grau de separação e de compactação. Estas medidas fornecem uma ideia de qualidade que pode não corresponder a realidade e a percepção dos especialistas. Como estamos considerando uma aplicação de agrupamento no contexto

da organização dos dados com o objetivo de facilitar a busca da informação, esta desconexão entre a qualidade e a utilidade real é inapropriada. Se uma aplicação requer uma avaliação em um contexto exploratório ou de descoberta de conhecimento onde uma solução conhecida não existe, a validade interna não é aplicável.

Neste trabalho podemos contar com as coleções já classificadas utilizadas para a avaliação do desempenho do modelo proposto na tarefa de classificação. Tais coleções podem ser usadas para avaliar a qualidade do agrupamento sob a ótica de validação externa. Assim, podemos avaliar como cada documento é recuperado para um agrupamento específico. Esta abordagem parece ser a mais razoável já que as soluções de classificação apresentadas pelos especialistas de cada uma das coleções utilizadas fornecem uma organização útil para a maioria dos usuários em potencial. Sendo assim, cumpre o objetivo imediato de avaliação da qualidade.

A medida adotada, especificamente, foi a medida F_1 para avaliar a qualidade do agrupamento. Esta medida é amplamente utilizada para avaliação da qualidade da classificação supervisionada de documentos, mas também é empregada para na avaliação do agrupamento . A medida fornece um bom equilíbrio entre abrangência e precisão, o que é vantajoso no âmbito da recuperação de informação.

A qualidade F_1 é computada como F_1 = 2*precisão*abrangência/(precisão + abrangência) onde valor 1 indica a qualidade máxima e o valor 0 indica a pior qualidade. Precisão e abrangência são definidas como p = a/(a+b) e r = a/(a+c) onde a é a quantidade de exemplos positivos verdadeiros, isto é, a quantidade de documentos colocados juntos em uma dada solução e que realmente foram agrupadas pelo algoritmo de agrupamento; b é a quantidade de exemplos falso positivos, isto é, a quantidade de documentos que não deveríam estar agrupados porém estão; e c é a quantidade de exemplos falso negativos, isto é, a quantidade de documentos que deveriam estar agrupados mas não foram agrupados pelo algoritmo.

Para computar a, b e c é necessário que tenhamos uma solução $S = \{S_j \mid j = 1, 2, ..., ..., M^S\}$, onde $M^S = |S|$ é a quantidade de tópicos que o especialista encontrou na coleção de documentos. Cada tópico S_j , por sua vez, é um conjunto de documentos que "pertence" àquele tópico. A saída de um sistema de classificação de textos é um conjunto de classes $C = \{C_i \mid i = 1, 2, ..., M^S\}$. A classe C_i corresponde ao tópico S_i. De forma similar, a saída de um sistema de agrupamento de textos é um conjunto de grupos (clusters) $C = \{C_i \mid i = 1, 2, ..., M\}$ onde M é a quantidade de grupos encontrada. M^S é, geralmente, desconhecida no contexto do agrupamento e portanto M pode não ser igual

a M^S. E ainda, com o agrupamento não existe garantia de que a classe C_i corresponde ao tópico S_i. Por outro lado, a saída de um algoritmo de agrupamento é indistinguível da saída de um algoritmo de classificação supervisionado, que é um conjunto de conjuntos de documento, sendo, portanto denotados por C. No caso da classificação de textos supervisionada, estabelecer os valores de a, b e c é simples desde que sabemos que para cada tópico S_j a classe correspondente será C_j. Portanto, os únicos pares de tópicos-classes (S_j, C_i) que necessitam consideração são aqueles onde i=j. Temos, então, para cada tópicos j:

$a_j = | C_j \cap S_j|$ (quantidade de documentos tanto em C_j quanto em S_i)
$b_j = |C_j| - a_j$
$c_j = |S_j| - a_j$

Estes valores são então reunidos em um valor global de F_1 ou macro-média ou micro-média.

A forma como a, b e c são computados em classficação de textos não pode ser aplicada ao agrupamento de textos já que nós não sabemos a princípio a correspondência existente entre os grupos a os tópicos da solução manual. Em outras palavras, não podemos considerar os pares tópicos-grupos (S_j, C_i) para cada i=j. A abordagem usada em geral para computar a, b e c e, consequentemente F_1 em agrupamento de textos é considerar o melhor grupo i^* (aquele com o maior valor $F_1{}^{i}*$) para cada tópico j como o grupo correspondente ao tópico e calcular a média destes melhores valores F_1:

$$F_1 = \frac{\sum_{j=1}^{M^S} |S_j| F_1^{i^*}}{\sum_{j=1}^{M^S} |S_j|} \qquad (6.6)$$

Este modo de computar F_1 é similar a forma de computação da macro-média F_1 para a classificação de textos, mas devido a ponderação de cada tópico de tamanho $|S_j|$ deve se comportar como a micro-média. Por considerar apenas os melhores casamentos tópico-grupo, muitos falso negativos e falso positivos não são incluídos no cálculo.

6.4 Representação dos Documentos

A representação dos documentos para o experimento com agrupamento não difere da representação adotada para o experimento de classificação apresentado na seção 5.5, exceto pelo fato de não exigir a representação da classe no vetor do documento. Desta forma, para facilitar a leitura, o conteúdo desta seção 5.5 encontra-se transcrito a seguir.

Nas abordagens convencionais, documentos são representados como vetores de palavras, denominados "*bag-of-words*", como representado na Figura 56.

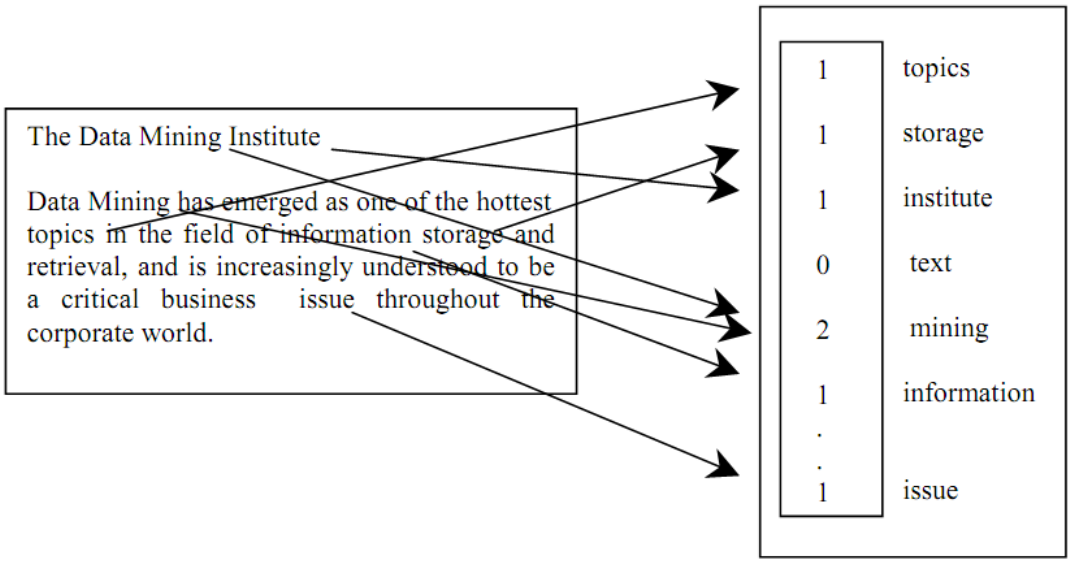


Figura 56. Reprodução do Vetor de Palavras

A partir de um dicionário previamente confeccionado, os documentos são transformados e representados como vetores, contendo valores binários indicando a presença ou ausência de cada palavra ou números, onde cada número corresponde a frequência, normalizada ou não, da ocorrência de cada palavra no documento.

Como já mencionado, tal representação apresenta dois problemas graves: a alta dimensionalidade de características, já que o tamanho do vetor é proporcional ao tamanho do dicionário, e a perda da informação contida nos textos originais, pois as relações mantidas entre as palavras não são capturadas.

A representação utilizada neste experimento é uma adaptação desta abordagem. O dicionário é composto pelos grânulos da coleção. Os vetores são construídos com base na proporção de palavras de cada grânulo em cada um dos documentos. Tal proporção é definida pela divisão da quantidade de palavras no documento e no grânulo pela quantidade de palavras no grânulo. A figura 57 apresenta a ideia.

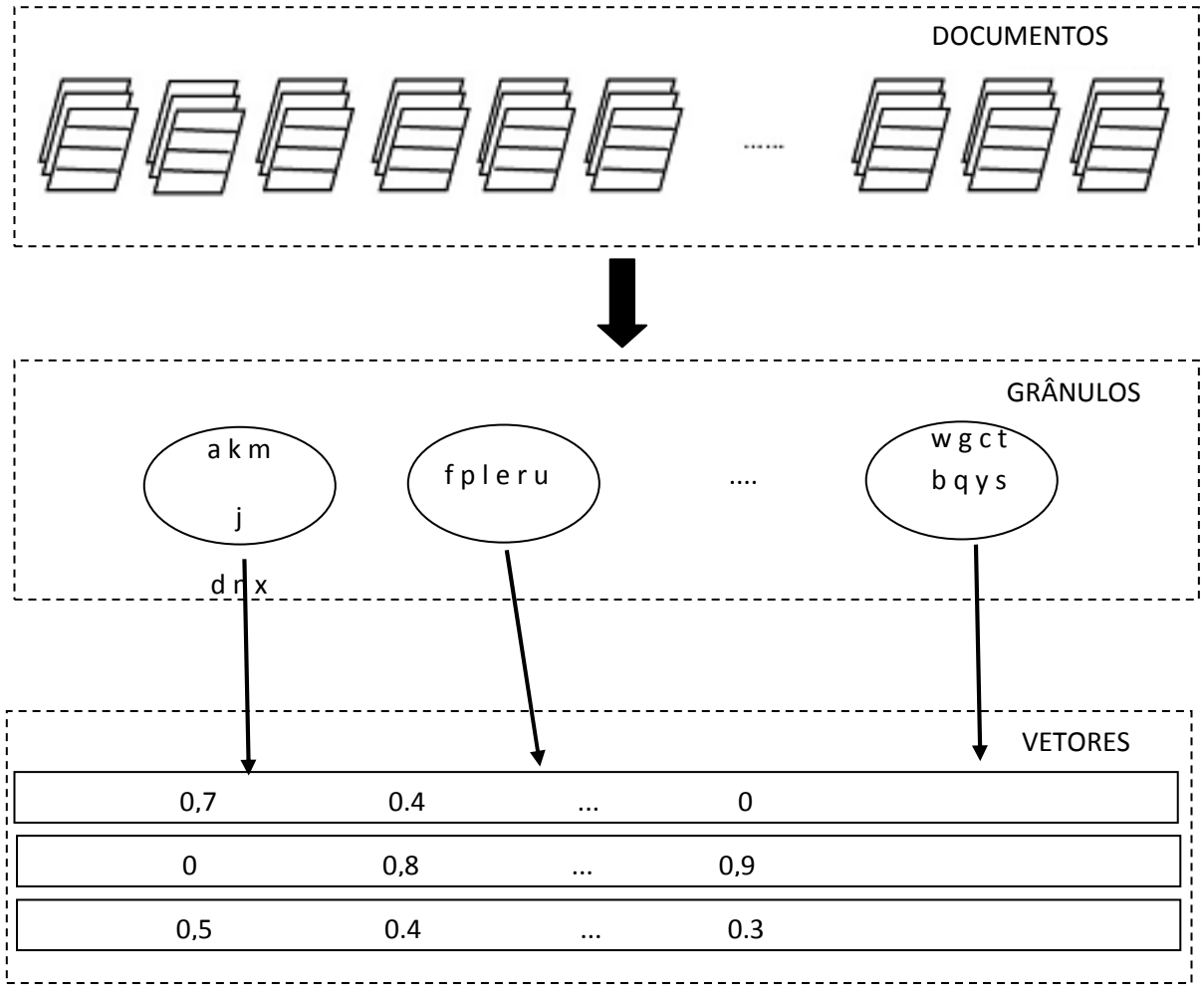


Figura 57. Reprodução Vetor de Grânulos.

6.5 Experimento

Para o experimento foram utilizadas as mesmas coleções de documentos descritas no Capítulo 4. Cada uma destas bases foi submetida aos procedimentos de

pré-processamento, descritos na seção 4.3.2 e ao algoritmo de agrupamento espectral apresentado na seção 4.3.3. Assim como no experimento com classificação, a quantidade de grânulos gerada foi definida de forma proporcional a quantidade de documentos em cada coleção. Pela mesma razão apresentada para o experimento com a classificação, foi escolhido o percentual de 10% para a proporção de grânulos em relação quantidade de documentos na coleção. A Tabela 17 ação apresenta estes números.

Tabela 17. Quantidade de Grânulos por Coleção (reprodução)

Coleção	Quantidade de Documentos	Quantidade de Grânulos
1	200	20
2	160	16
3	5.485	548
4	2.500	250

Ao final da construção dos dicionários de grânulos, os vetores representantes de cada coleção foram construídos conforme definido na seção 6.5. Cada um dos conjuntos de vetores foi submetido aos algoritmos de agrupamento apresentados na seção 6.4 e os resultados avaliados conforme os critérios de avaliação apresentados na seção 6.5. Procedimentos semelhantes foram adotados para criação dos vetores construídos a partir das técnicas LSA e LDA, também submetidos aos mesmos algoritmos. Os resultados alcançados estão organizados em tabelas na seção 6.7.

6.6 Resultados

Para cada uma das coleções, representadas a partir de cada um dos modelos de documentos e submetidas a cada um dos algoritmos de agrupamento usados no experimento, foram calculados os valores média como definido na seção 6.3.

6.6.1 Média F1 – Algoritmo K-Means

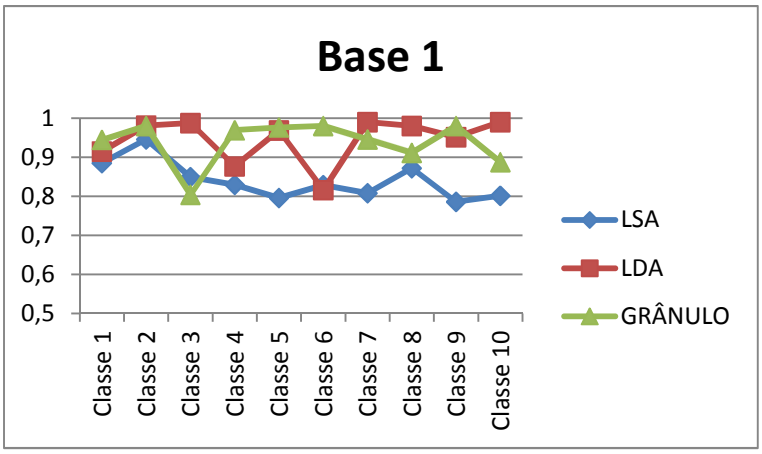


Figura 58. Resultados Média F1 para Algoritmo K-Means sobre Base 1.

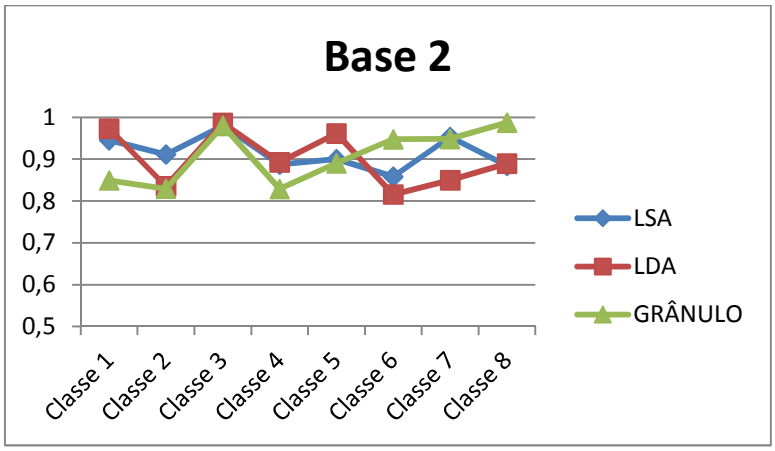


Figura 59. Resultados Média F1 para Algoritmo K-Means sobre Base 2.

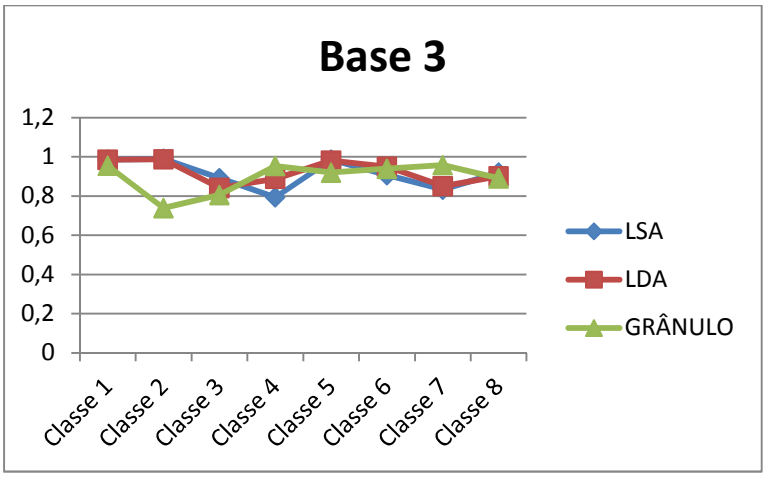


Figura 60. Resultados Média F1 para Algoritmo K-Means sobre Base 3.

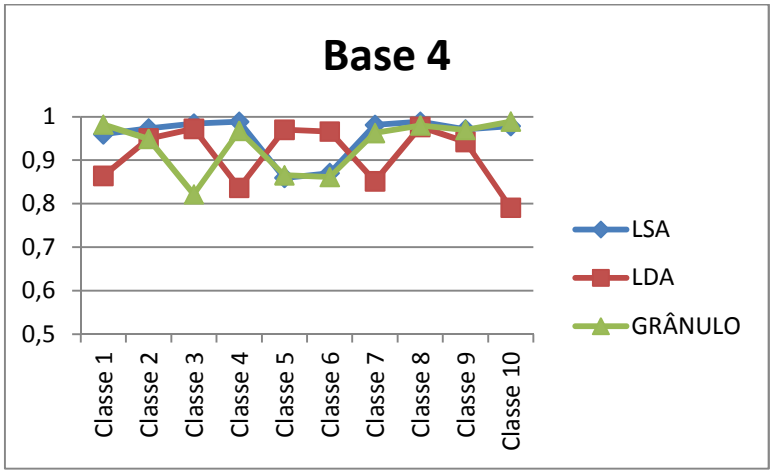


Figura 61. Resultados Média F1 para Algoritmo K-Means sobre Base 4.

6.6.2 Média F1 – Algoritmo Aglomerativo

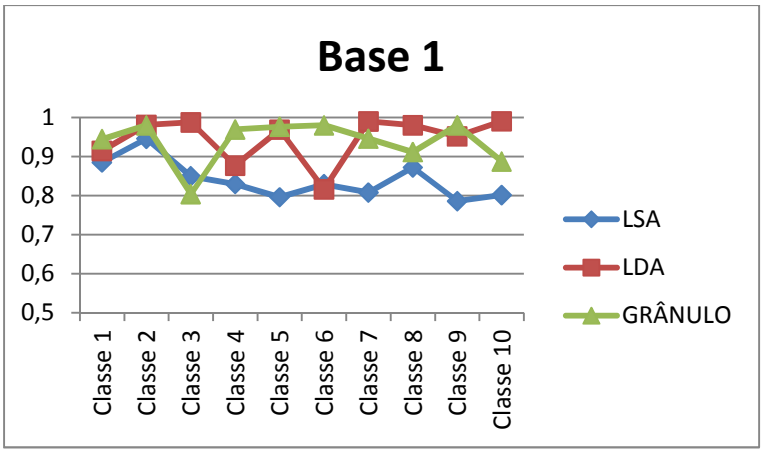


Figura 62. Resultados Média F1 para Algoritmo Aglomerativo sobre Base 1.

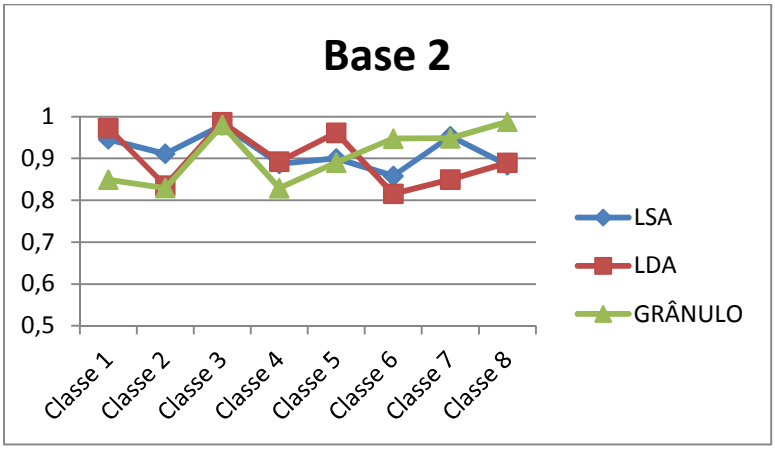


Figura 63. Resultados Média F1 para Algoritmo Aglomerativo sobre Base 2.

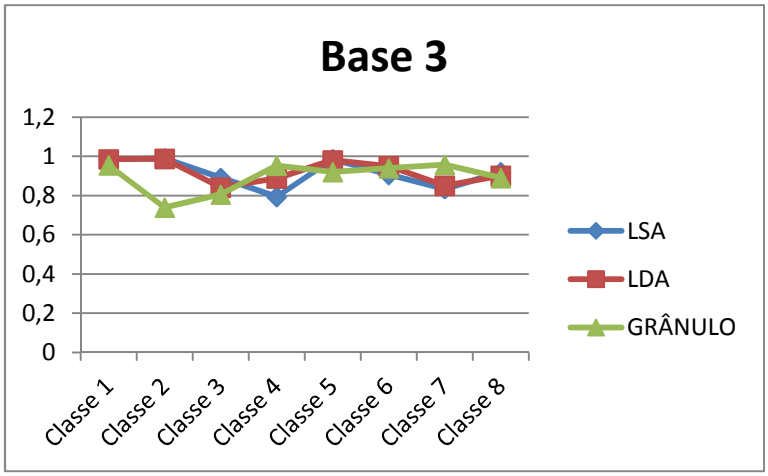


Figura 64. Resultados Média F1 para Algoritmo Aglomerativo sobre Base 3.

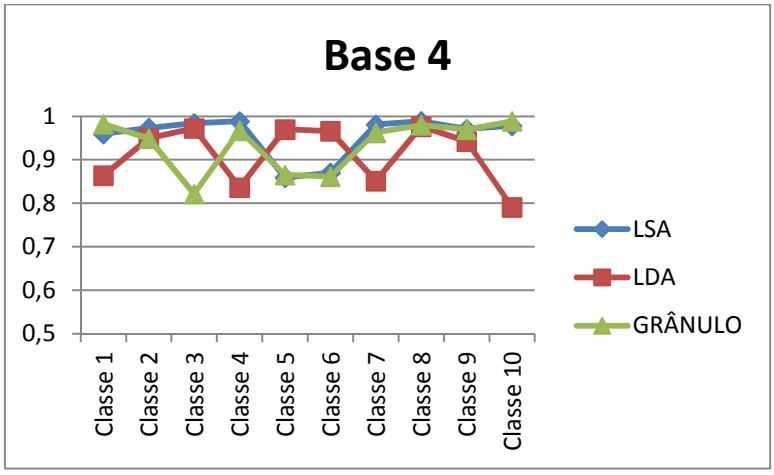


Figura 65. Resultados Média F1 para Algoritmo Aglomerativo sobre Base 4.

6.6.3 Avaliação dos Resultados

Valem aqui as mesmas considerações feitas a análise dos resultados alcançados pelo modelo quando aplicado a tarefa de classificação de documentos apresentada na seção 5.7.9. Os resultados do agrupamento também estão sempre vinculados a fatores que os modificam a cada vez que são alterados. Alguns exemplos destes fatores são a quantidade de grânulos escolhida, a quantidade de documentos em cada coleção, os algoritmos escolhidos para o agrupamento, bem como os parâmetros adotados nestes algoritmos.

Neste trabalho foram adotadas técnicas, tanto para o experimento realizado quanto para os critérios de avaliação, que vêm sendo utilizaas na maioria dos trabalhos publicados nesta área. Pode-se afirmar, apenas com base nos critérios de avaliação adotados, que os resultados alcançados são bastante coerentes com os resultados apresentados nestes trabalhos, apesar da existência de diferença entre os fatores anteriormente relacionados. Sendo assim, da mesma forma como afirmado no capítulo 5, que tratou da avaliação do modelo em tarefas de lcassdificação, cabe a afirmativa de que o modelo de documentos baseado em grânulos, proposto neste trabalho, pode também representar uma alternativa aos modelos que buscam identificar as relações entre as palavras contidas nos documentos com o objetivo de enriquecer a sua representação para aprimorar o desempenho dos algoritimos de agrupamento.

6.7 Considerações Finais

Este capítulo apresentou a aplicação dos grânulos de palavras na representação de documentos. Mostrou a sua influência sobre os resultados no agrupamento destes documentos e comparou estes resultados com a técnicas Análise de Semântica Latente (LSA) e Latent Dirichlet Allocation (LDA), comumente usadas com o objetivo criar modelos capazes de representar as relações entre as palavras contidas nestes documentos. O capítulo seguinte apresenta as conclusões deste trabalho e algumas sugestões para trabalhos futuros.

7. CONCLUSÃO

Este trabalho apresentou uma proposta de modelo de documentos baseado nos princípios da computação granular. Para esta apresentação todos os aspectos relacionados a esta abordagem de solução de problemas foram levantados e analisados em seus detalhes. A computação granular é um processo duplo de solução de problemas: após um processo de granulação, um processo de integração destes grânulos resulta na solução do problema. Estes grânulos formam um espaço, denominado espaço de aproximação, que é utilizado para aproximar conceitos e padrões. Existem vários espaços de aproximação, mas aquele definido como sistema de vizinhança mostrou-se mais adequado para tratar o problema em questão. No contexto de tratamento de textos, a vizinhança pode ser vista como uma relação de proximidade ou similaridade entre as palavras contidas nos documentos. Para a avaliação desta similaridade foi proposta uma relação fuzzy, dado que este aspecto é inerente ao processo de granulação sob quaisquer domínios.

A abordagem por agrupamento espectral foi, então, adotada como complementar a criação do espaço de aproximação, já que é capaz de efetuar o agrupamento de dados com base na relação entre os dados. Foi feito um estudo amplo sobre esta abordagem de agrupamento e, devido a semelhança de resultados apresentada pelos três principais algoritmos com esta finalidade, foi o escolhido a abordagem de agrupamento espectral normalizado proposta em (SHI & MALIK, 2000). Este agrupamento finaliza a construção dos grânulos de palavras que compõem o espaço de aproximação da coleção de documentos. A partir deste espaço o modelo de documento foi construído através de uma adaptação do modelo vetorial.

Para avaliação do modelo proposto, quatro coleções de documentos, com características distintas, foram selecionadas e tiveram seus documentos representados com base no modelo. A título de comparação as mesmas coleções foram submetidas a duas abordagens de representação de documentos com objetivos semelhantes a abordagem proposta neste trabalho. Os resultados das três representações foram comparados e o modelo proposto apresentou um desempenho bastante satisfatório. Finalmente, e com base nos resultados positivos alcançados, o modelo foi analisado para as tarefas de classificação e agrupamento de documentos.

O modelo proposto é uma alternativa ao tradicional e permite o desenvolvimento de tipos diferentes de ferramentas, capazes de explorar as relações mantidas entre as palavras contidas nos documentos. O modelo também irá permitir que todos os aspectos relacionados a computação com grânulos, mencionados no Capítulo 2, sejam explorados de forma efetiva. Os resultados alcançados demonstram que se trata de uma abordagem promissora para o desenvolvimento de novas ferramentas que auxiliem mais eficientemente a recuperação da informação ou qualquer outra tarefa que envolva texto.

7.1 Contribuições

Ao final deste trabalho, pode-se listar as seguintes contribuições:

- Foi proposto um novo modelo de documentos, alternativo ao tradicional baseado em *bag of words* , que possui como característica principal a possibilidade de captura das relações mantidas entre as palavras contidas no documentos.
- O modelo proposto apresenta vantagens sobre os modelos já existentes com abordagem semelhante:
    - Os modelos baseados em análise de semântica latente (LSA) e modelo de tópicos (LDA) criam conceitos ou tópicos (grânulos) em um único nível de detalhe. Nestas abordagens não existe um mecanismo que possibilite a variação da granularidade dos conceitos abstraídos. O modelo proposto permite que sejam explorados vários níveis de granularidade, bastando para isso variar a quantidade de grânulos gerados. Quanto maior a quantidade de grânulos, mais específicos serão os conceitos e vice-versa. Esta característica permite que vários níveis de conhecimento sejam construídos a parte de um mesma coleção de documentos.
    - O modelo baseado em LDA é baseado em análise estatística do conteúdo dos documentos. Não há, na realidade uma

análise das relações entre as palavras. O modelo proposto analise e utiliza estas relações na construção dos grânulos.

- Foi proposta uma análise de similaridade fuzzy para as relações entre as palavras que restringe o domínio destes valores ao intervalo [0,1]. Os algoritmos de agrupamento, em geral, apresentam melhor desempenho em domínios discretos como o que o trabalho propõe.
- O trabalho apresenta, de forma detalhada, um estudo comparativo de desempenho do modelo proposto com os dois principais modelos de captura de conceitos existentes atualmente (LSA e LDA). Além disso, faz um estudo detalhado da aplicação do modelo às tarefas de classificação e agrupamento de documento. Este estudo também envolve a comparação de resultados com estes modelos mais conhecidos.
- (YAO, 2007) (HOEBER, 2008) (HOEBER, 2012) propõem que a evolução dos sistemas de recuperação de informação é os sistemas de suporte a recuperação de informação. Neste contexto, os sistemas, deixarão de ter como foco a busca e a recuperação e serão desenvolvidos para suportar as várias tarefas que os usuários podem executar ao interagir com uma coleção de documentos. Tais tarefas envolvem processos como recuperação, filtragem, extração e coleta de informação, bem como, classificação, agrupamento e sumarização de documentos com o objetivo de auxiliar estas pessoas na localização mais eficiente de documentos que satisfaçam suas necessidades de informação. Tais necessidades podem ser definidas como descobrir ou derivar novas informações, encontrar padrões entre estas informações ou separar a informação útil da não útil. Para viabilizar sistemas deste tipo, algumas das características dos sistemas atuais devem ser repensadas, dentre elas o modelo de representação dos documentos. O modelo proposto neste trabalho representa um avanço neste sentido, já que é capaz de capturar características nos documentos que são fundamentais para o bom desempenho destas tarefas. Os resultados obtidos na análise com as tarefas de classificação e agrupamento comprovam esta afirmação.

- O modelo proposto pode ser usado para incorporar todas as características da computação granular, já reconhecida como uma boa técnica para tratamento de problemas complexos, aos sistemas que lidam com dados do tipo texto. A partir do novo modelo de documentos, novos modelos de recuperação de informação e novos modelos de apresentação da informação podem ser propostos e implementados.
- Teste do modelo proposto em classificação e agrupamento, considerando os diversos níveis de granularidade.

7.2 Trabalhos Futuros

Este trabalho (embrionário) pode evoluir basicamente em duas direções. A primeira com o objetivo de aperfeiçoar o modelo e outra com o objetivo de desenvolver ferramentas com funcionalidades capazes de usufruir da sua capacidade de captura da semântica presente nos documentos e da capacidade de construção do conhecimento a partir de mais de um nível de granularidade.

Em se tratando da evolução do modelo propriamente dito, o aperfeiçoamento está vinculado à tarefa de construção dos grânulos, mas especificamente ao algoritmo de agrupamento. O algoritmo adotado não é capaz de lidar com a possibilidade de uma palavra estar associada a mais de um grânulo simultaneamente, o que é uma característica muito comum do universo textual (polissemia). O estudo da aplicação de uma abordagem fuzzy, como a proposta em (COMINETTI, 2010), para a etapa de agrupamento das palavras é a sugestão mais imediata. Além disso, todos os algoritmos de agrupamento espectral estudados têm como característica a necessidade da informação do valor de k, a quantidade de grânulos. Também é uma evolução importante a possibilidade de adoção de um algoritmo onde esta informação não fosse obrigatória.

Em um ambiente de recuperação de informação ou qualquer outro cujo foco seja o conteúdo textual, os documentos servem como dados brutos. Os modelos de documento tratam apenas da sua representação e interpretação. Os modelos de recuperação tratam das funcionalidades da busca, fornecendo linguagens e ferramentas para auxiliar o

usuário a executar tarefas tais como pesquisa e navegação. Os modelos de apresentação tratam da representação e interpretação dos resultados da busca. Eles permitem que o usuário veja e organize os resultados da busca. O modelo proposto viabiliza múltiplas representações e interpretações dos documentos, já que permite a variação da granularidade dos dados e do nível do conhecimento representado. O desenvolvimento de novos modelos, tanto de recuperação quanto de apresentação, são o próximo passo na evolução desta proposta no sentido de aperfeiçoar a sua aplicação.

Referências

AGGARWAL, 2012. Aggarwal C.C., Zhai C.X. (eds.). Mining Text Data. DOI 10.1007/978-1-4614-3223-4_4, Springer Science+Business Media, LLC 2012.

ANAYA, 2011. Anaya L. Comparing Latent Dirichlet Allocation and Latent Semantic Analysis as Classifiers. Tese de Doutorado. University of North Texas.

BANERJEE, 2002. S. Banerjee S, T. Pedersen.An adapted Lesk algorithm for word sense disambiguation using WordNet, Proceedings of the third international conference on intelligent text processing and computational linguistics. Mexico City, Mexico, 136–45.

BANERJEE, 2007. Banerjee A., Basu S. . Topic models over text streams: A study of batch and online unsupervised learning. In SDM. SIAM, 2007.

BEEFERMAN, 2000. D. Beeferman and A. Berger, "Agglomerative clustering of a search engine query log", in Proceedings of KDD'00, pp. 407-415, 2000.

BILLHART, 2002. H.Bilhardt, D. Borrajo, V.Maojo. A context vector model for information retrieval. Journal of the American Society for Information Science and Technology. Volume 53 , Issue 3 (February 2002). Pages: 236 - 249 . ISSN:1532-2882.

BIRÓ, 2009. Bıro, I. , Szabo, J. Latent Dirichlet Allocation for Automatic Document Categorization. In Proceedings of the 19th European Conference on Machine Learning and 12th Principles of Knowledge Discovery in Database.

BLEI, 2003. Blei D, Ng A, Jordan M : Latent Dirichlet Allocation. Journal of Machine Learning Research 3: 993-1022

BOLLEGARA, 2007. D. Bollegara, Y. Matsuo, M Isizuka, Measuring Semantic Similarity between Words Using Web Search Engines, Proceedings of the 16thInternational World Wide Web Conference (WWW2007), 757-766, Banff, Alberta, Canada.

CARPINETO, 2004. Cláudio Carpineto, Giovanni Romano. Concept Data Analysis: Theory and Applications. Hardcover. ISBN: 978-0-470-85055-8. Willey. 2004.

CHUNG, 1997. Chung F.R.K. Spectral Graph Theory. American Mathematical Society (CBMS series 92).

CILIBRASI 2007. R Cilibrasi, P Vitanyi, The Google Similarity Distance, IEEE Transactions on Knowledge and Data Engineering, 19, 370-383.

COMINETTI, 2010. Cominetti O., Matzavinos A., Samarasinghe S., Kulasiri D. Liu S., Maini P., Erban R. DifFUZZY: a Fuzzy Clustering Algorithm for Complex DataSets.

International Journal of Computational Intelligence in Bioinformatics and Systems Biology, v.1 n.4 402.

CRISTIANI, 2000. Cristiani N., Taylor J. An Introduction to Support Vector and other Kernel-based Learning Methods. Cambridge, Cambridge Univerty.

CROFT, 1980. Croft, W. B., A Model of Cluster Searching Based on Classification. Information Systems, Vol. 5, 189-195.

CROFT, 1992. Croft W. Bruce, Turtle Howard R.: A Comparison of Text Retrieval Models. Comput. J. 35(3): 279-290.

CROUCH, 1989. C Crouch, An Approach to The Automatic Construction of Global Thesauri. Information Processing & Management, 26(5), 629-640.

DASH & LIU, 1997. Dash M., Liu H. Feature selection for classification. Intelligent Data Analysis, 1(3), 131–156.

DEERWESTER, 1990. Deerwester S , Dumais S, Landauer T, Furnas G, Harshman R : Indexing by Latent Semantic Analysis. JASIS 41(6): 391-407.

DHILLON, 2001. Dhillon I.S., Fan J., Guan Y. Efficient Clustering of Very Large Document Collections. Data Mining for Scientific and Engineering Applications, p 357-381. Kluwer Academic Publishers.

DING, 2001. Ding C.H.C., He X., et al. A Min-Max Cut Algortithm for Graph Partitioning and Data Clustering. Proceedings of the 2001 IEEE International Conference on Data Mining. IEEE Computer Society.

DING, 2004. Ding C.H.Q. Spectral Clustering. Tutorial presented at ICML 2004. Banff, Alberta, Canada.

DOAN, 2005. S. Doan, Q. Ha, S. Horiguchi. A Fuzzy-Based Approach for Text Representation in Text Categorization. 14th IEEE International Conference on Fuzzy Systems, 2005. FUZZ '05. Publication Date: 25-25 May 2005. On page(s): 1008-1013. ISBN: 0-7803-9159-4.

FELLBAUM, 1998. C Fellbaum. WordNet: An Electornic Lexical Database. Bradford Books.

FILIPPONE, 2008. Filippone I., Camastra F., et al. A Survey of Kernel and Spectral Methods for Clustering. Pattern Recognition, v. 41, n. 1, p. 176-190. ,

FISHBEIN, 2008. J FishBein, Integrating Structure and Meaning Using Holographic Reduced Representation to Improve Automatic Text Classification. Master Thesis. University of Waterloo.

GABRILOVICH, 2007. E Gabrilovich, S Markovitch, Computing Semantic Relatedness using Wikipedia-based Explicit Semantic Analysis, Proceedings of the 20th

International Joint Conference on Artificial Intelligence (IJCAI), 1606-1611, Hyderabad, India.

GOLUB & VAN LOAN, 1996. Golub G.H., Van Loan C.F. Matrix Computations. John Hopkins University Press. 694 p. John Hopkins Studies in the Mathematical Sciences.

HAGEN & KAHNG, 1992. Hagen L., Kahng A. New Spectral Methods for Ratio Cut Partitioning and Clustering. IEEE Trans on Computed Aided Design, v.11, p. 1074-1085.

GOSH, 2003.Ghosh J.,Zhong. A Unified Framework for Model-based Clustering. Journal of Machine Research, 4:1001–1037.

GRABMEIER, 2002. Grabmeier J., Rudolph A. Techniques of Cluster Algorithms in Data Mining. Data Mining Knowledge Discovery 6(4), p. 303-360.

HAMAD, 2008. Hamad D., Biela P. Introduction to Spectral Clustering. Informationand Communication Technologies. From Theory to Applications. ICTTA 2008. p. 1-6.

HOBBS J., 1985. Granularity. Proceedings of the Ninth International Joint Conference on Artificial Intelligence, 432-435.

HOGBEN, 2009. HogBen L. Spectral Graph Theory and the Inverse Eigenvalue Problem of a Graph. Chamchuri Journal of Mathematics, p.51-72.

HOEBER, 2008.Hoeber O. , Web Information Retrieval Support Systems: The future of Web Search. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence - Workshops (International Workshop on Web Information Retrieval Support Systems), pp. 29-32.

HOEBER, 2012. Hoeber O. , Human-Centred Web Search, In Next Generation Search Engines: Advanced Models for Information Retrieval (C. Jouis, I. Biskri, J-G Ganascia, and M. Roux, editors), IGI Global, pp. 217-238.

HOFMANN, 1999. Probabilistic Maps: Navigating Through Large Text Collections. Advances in Intelligent Data Analysis. Lecture Notes in Computer Science. Vol 1642, p. 161-172.

INDERJIT, 2004. Inderjit S.D. Yuqiang, et al. Kernel K-means: Spectral Clustering and Normalized Cuts. Proceedings of the ACM SIGDKK. Seatlle, WA, USA. ACM Press 2004.

INGERSEN, 2008 M. Skov, B. Larsen and P. Ingwersen. Inter and intra-document contexts applied in polyrepresentation for best match IR Information Processing and Management: an International Journal. Volume 44 , Issue 5. Pages 1673-1683.

JAIN, 1999. Jain, A.K., Hong, L., Kulkarni Y. A multimodal biometric system using Fingerprint, face and speech. Proceedings of the Second International Conference on Audio- and Video-based Biometric Person Authentication,Washington DC, p. 182-187.

JIANG, 1997. J Jiang , D Conrath . Semantic Similarity Based on Corpus Statistics and Lexical Taxonomy. In: Proceedings of the 10th International Conference on Research in Computational Linguistics, 19–33,Taipei, Taiwan.

JOACHIMS, 1997. Joachims T, Text Categorization with Support Vector Machines: Learning with Many Relevant Features. LS8-Report 23, Universität Dortmund, LS VIII-Report, 1997.

JORDAN & WEISS, 2001. Ng A., Jordan M., et al. On Spectral Clustering: Analysis and an Algorithm.

KHALLED, 2006. S Khalled. A Semantic Graph Model for Text Representatiion and Matching in Document Mining. PhD Thesis. University of Waterloo. Canadá.

KIM, 2010. Kim Y., Pessiot J., Amini M., Gallinari P.: Improving document clustering in a learned concept space. Inf. Process. Manage. 46(2): 180-192.

KOZIMA, 1993. Kozima, T. Furugori. . Similarity between words computed by spreading activation on an English dictionary. In Proceedings of the 6 th Conference of the European Chapter of the ACL, 232- 239.

LEACOCK, 1998. C. Leacock ,M. Chodorow, Combining Local Context and WordNet Similarity for Word Sense Identification in WordNet, An Electronic Lexical Database, 265—283, MIT Press.

LI, 2003. Y Li, Z Bandar, D McLean, An Approach for Measuring Semantic Similarity between Words Using Multiple Information Sources, IEEE Transactions on Knowlege and Data Engineering, 15, 871-882.

LIN, 1989. Lin, T. Y. (1989) Neighborhood Systems and Approximation in Database and Knowledge Base Systems, Proceedings of the Fourth International Symposium on Methodologies of Intelligent Systems, 75-86.

LIN, 1997. Lin, T. Y. Neighborhood Systems –A Qualitative Theory for Fuzzy and Rough Sets, Advances in Machine Intelligence and Soft Computing, Volume IV. Ed. Paul Wang, Duke University, North Carolina, 132-155.

LIN, 1998. Lin, T. Y. Granular Computing on Binary Relations I: Data Mining and Neighborhood Systems, In: Rough Sets In Knowledge Discovery, A. Skowron and LPolkowski (eds), Physica-Verlag, 107-121.

LIN, 1998A. D.Lin. An Information Theoretic Definition of Similarity, Proceedings of the 15th International Conference on Machine Learning, , 296–304, Madison , Wisconsin USA.

LIN, 2004 Lin, T. Y., Cercone, N., Hu, X., and Han, J. Intelligent Query Answering Based on Neighborhood Systems and Data Mining Techniques, Proc. of 8th International Database Engineering and Applications Symposium, 91-96.

LIN, 2007 T. Young Lin. Granular Computing and Modeling the Human Thoughts in Web Documents. Lecture Notes In Artificial Intelligence; Vol. 4529. Proceedings of the 12th international Fuzzy Systems Association world congress on Foundations of Fuzzy Logic and Soft Computing. Cancun, México. Pages: 263 - 270 . ISBN:978-3-540-72917-4.

LIU, 1994 G. Liu. The Semantic Vector Space Model (SVSM) A Text Representation and Searching Technique System Sciences, 1994. Vol.IV: Information Systems: Collaboration Technology Organizational Systems and Technology, Proceedings of the Twenty-Seventh Hawaii International Conference. Pag 928-937.

LUCARELLA, 1993. Lucarella D., Parisotto S., Ozi A.: MORE: Multimedia Object Retrieval Environment. Hypertext 1993: 39-50.

MANNING, 2008. Manning C., Hall D., Jurafsky D. Studying the History of Ideas Using Topic Models. Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing, pp. 363-371

MACQUEEN, 1967. MacQueen J.B. Some Methods for Classifications and Analysis of Multivariate Observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. Berkeley. University of California Press., v. 1. P. 281-297.

MOHAR, 1992. Laplace Eingenvalues of Graphs – a Survey. Discrete Math. V.109. n.1-3, p. 171-183.

MOHAR, 1997. Mohar B. The Laplacian Spectrum of Graphs. In: Alavi Y., Schwenk A.J. (Ed.). Graph Theory, Combinatorics and Applications. New York. John Willey, v2, p.871-898.

MILLAR, 2009. Millar J., Peterson G., Mendenhall M. Document Clustering and Visualization with Latent Dirichlet Allocation and Self-Organizing Maps. Artificial Intelligence Research Society Conference, May 19-21, 2009, Sanibel Island, Florida, USA.

MITCHELL T, 1997. Machine Learning. McGraw Hill.

NGUYEN, 2008. H Nguyen, T Ho. Rough Document Clustering and the Internet. In Handbook of Granular Computing , pp 987- 1003. John Wiley & Sons.

NIELSEN, 1990. Nielsen, J. Hypertext and Hypermedia. Academic Press, Boston, ISBN 0-12-518410-7 (hardcover), 0-12-518411-5 (paperback).

PALMER, 1994. Z Wu, M. Palmer. Verb semantics and lexical selection, Proceedings of the Annual Meeting of the Association for Computational Linguistics, 133-138, Las Cruces, New Mexico.

PAWLAK, 1982. Pawlak, Z. Rough Sets, International Journal of Computer and Information Sciences 11: 341-356.

PAWLAK, 1998. Granularity of knowledge, indiscernibility and rough sets, Proceedings of IEEE International Conference on Fuzzy Systems, 106-110.

PETERS, 2006. Peters, J. F., Skowron, A., Stepaniuk, J. (2006) Nearness in Approximation Spaces, Proc. Of Conference on Concurrency, Specification & Programming, 434-445.

PETERS, 2007. Peters, J. F. (2007) Near Sets, Special Theory about Nearness of Objects, Fundamenta Informatocae 75 (1-4), 407-433.

PETERS, 2007A. Peters, J. F. (2007) Near Sets, Toward Approximation Space-Based Object Recognition, Proc. of the 2nd International Conference on Rough Sets and Knowledge Technology (LNAI 4481), 22-33, Springer.

POLKOWSKI, 2002. Polkowski, L Rough Sets: Mathematical Foundations, Springer, Physica-Verlag, Heidelberg.

PEDRYCZ, 2005 Pedrycz W. W. Knowledge-Based Clustering. From Data to Information Granules. John Wiley & Sons, Inc., Hoboken, New Jersey.

MANNING, RAGHAVAN & SCHUTZ, 2008. Manning C., Raghavan P., Schutz H. Introduction to Information Retrieval. Cambridge University Press 2008, isbn 978-0-521-86571-5, pp. I-XXI, 482p.

RADA, 1989. R.Rada, H.Mili, E.Bichnell, M.Blettner, Development and Applicationof a Metric on Semantic Nets, IEEE Trans. Systems, Man, and Cybernetics, 9, 1-30.

RANZATO, 2008 M Ranzato. Semi-supervised Learning of Compact Document Representations with Deep Networks. Proceedings of the 25 th International Conference on Machine Learning, Helsinki, Finland.

RAPP, 2002. R Rapp. The computation of word associations: Comparing syntagmatic and paradigmatic approaches. In Proceedings of COLING-02, Taipei, Taiwan.

RESNIK, 1999. P Resnik. Semantic Similarity in Taxonomy: An Information Based Measure and its Application to Problems of Ambiguity in Natural Language. Journal of Artificial Intelligence Research, 11, 95–130.

RIJSBERGEN, 1975. RijBergen C.J. Van. Information Retrieval. Butterworth, (Publishers) Limited. 152p. ISBN 0408707178.

ROTMAN, 2009. Rotman J. An Introduction to homological algebra. Springer. 2009.

SALAHLI, 2009. Salahli M. An Approach for Measuring Semantic Relatedness Between Words Via Related Terms. Mathematical and Computational Applications, Vol. 14, No. 1, pp. 55-63.

SALTON, 1973. Salton G. Recent Studies in Automatic Text Analysis and Document Retrieval *J. ACM*, Vol. 20, No. 2. (April 1973), pp. 258-278, doi:10.1145/321752.321757.

SALTON, 1983 G. Salton and M. McGill, Introduction to Modern Information Retrieval, New York, McGraw Hill, 1983.

SCHAPIRE, 2012. Schapire R. and Freund Y. Boosting: Foundations and Algorithms. MIT Press, 2012.

SCHAEFER, 2007. Schaefer S.E. Graph Clustering. Computer Science Rewiew. V.I, p. 27-64.

SEBASTIANI, 2005. Sebastiani F. Text Categorization. In Alessandro Zanasi (ed.), Text Mining and its Applications, WIT Press, Southampton, UK, 2005, pp. 109-129.

SERRANO, 2006. J. I. Serrano, M. D Castillo. Text Representation by a Computational Model of Reading. I. King et al. ICONIP 2006, Part I, LNCS 4232, pp. 237 – 246, Springer-Verlag Berlin Heidelberg.

SHI & MALIK, 2000. Shi J. e Malik J. Normalized Cuts and Image Segmentation. Transactions on Pattern Analysis and Machine Intelligence, v 22, n 8, p 888-905.

SKOWRON, 2004. Skowron, A., Swiniarski, R., and Synak, P. Approximation spaces and information granulation, Lecture Notes in Computer Science 3066, 116-126, Springer Berlin/Heidelberg.

SMOLA, 1999. Smola A.J., Scholkopf B., Burges C.J.C. (Eds.) Advances in KernelMethods—Support Vector Learning. MIT Press, Cambridge, MA.

SONG, 2007. Song W., Park S. A novel document clustering model based on latent semantic analysis. In Proceedings of the Third International Conference on Semantics, Knowledge and Grid, pages 539–542.

SPIELMAN, 2007. Spelman D.A., Teng S.H. Spectral Partitioning Works. Plana Graphs and Finite Elements Meshes. Linear Algebra ans its Applications, v 421, n 2-3, p, 284- 305.

SRIURAI, 2011. Sriurai W. Improving Text Categorization by Using a Topic Model. Advanced Computing: An International Journal (ACIJ), vol 2, n. 6.

STEINBACH, 2000. Steinbach M., Karypis G., Kumar V. A Comparison of Document Clustering Techniques. KDD Worshop on Text Mining.

STOER & WAGNER, 1997. Stoer M., Wagner F. A Simple Min-Cut Algorithm. J. ACM 44(4), p. 585-591.

STEYVERS & GRIFFITHS, 2006. Steyvers M., Griffiths, T.L.. Probabilistic topic models. In T. Landauer, D McNamara, S. Dennis, and W. Kintsch (Eds.), Latent Semantic Analysis: A Road to Meaning. Mahwah, NJ: Erlbaum.

STRUBE, 2006. M Strube and S Ponzetto, WikiRelate! Computing Semantic Relatedness Using Wikipedia, Proceedings of the 21st National Conference on Artificial Intelligence, 1419-1424, Boston, Mass.

SUNG & MUKKAMALA, 2003. Sung A., Mukkamala S: Identifying Important Features for Intrusion Detection Using Support Vector Machines and Neural Networks. SAINT 2003: 209-217

TISHBY, 1999. Tishby N, Pereira F.C., Bialek W.: The Information Bottleneck method. The 37th annual Allerton Conference on Communication, Control and Computing, Sep 1999: pp. 368–377

VAPNIK, 1995. Vapnik V, Cortes C. Support vector networks. Machine Learning, 20:273–297, 1995.

VON LUXBURG, 2007. Von Luxburg U. A Tutorial on Spectral Clustering. Statistic and Computing. v.17, n.4, p. 395-416.

WANG, 2003. Wang M, Nie J. A Latent Semantic Structure Model for Text Classification, Workshop on Mathematical/Formal methods in information retrieval, 26th ACM-SIGIR, Toronto, Aug. 2003.

WANG, 2008. J. Wang, C. Wang, J. Liu. A Text Network Representation Model. Fifth International Conference on Fuzzy Systems and Knowledge Discovery. DOI 10.1109/FSKD.2008.215.

WANG, 2011. Wang C. Research ont the Text Clustering Algorithm based on Latent Semantic Analysis and Optimization. Proceedings of Computer Science and Automation Engineering (CSAE) 2011 IEEE International Conference, v.4, p.470-473.

WASHTELL, 2009. J Washtell, K Market. A Comparison of Windowless and Window-Based Computational Association Measures as Predictors of Syntagmatic Human Associations. Proceedings of the Conference on Empirical Methods in Natural Language Processing, pages 628–637, Singapore, 6-7.

WAZLAWICK, 2008. Wazlawick R.S. Metodologia de pesquisa para ciência da computação. – Rio de Janeiro: Elsevier, 2008. ISBN 978-85-352-3522-7.

WEI, 2006. Wei X., Croft W.: LDA-based document models for ad-hoc retrieval. SIGIR 2006: 178-185

WHITE & SMYTH, 2005. White S., Smyth P. A Spectral Clustering Approach to Finding Communities in Graphs. SIAM International Conference on Data Mining. P 76-84.

XING, 2007. Xing W., Croft W.B.: Investigating Retrieval Performance with Manually-Built Topic Models. RIAO 2007.

YANG & PEDERSEN, 1997. Yang Y., Pedersen O. A comparative study on feature selection in text categorization. In Douglas H.Fisher, editor, Proceedings of ICML-97, 14th International Conference on Machine Learning, pages 412–420, Nashville,1997. Morgan Kaufmann Publishers.

YAO, 2006. Yao, Y. Neighborhood Systems and Approximation Retrieval, Information Science 178(23), 331-345.

YAO, 2007. Yao Y. A Ten-year Review of Granular Computing. Proceedings of IEEE International Conference on Granular Computing, Silicon Valey, USA, Nov 2-5, pp 734-739.

YAO, 2007a. The Art of Granular Computing, LNAI 4585, Springer, 101-112.

YAO & ZHONG, 2007. Supporting Literature Exploration with Granular Knowledge Structures, Y Yao, Y Zeng, and N Zhong. In: Proceedings of the 11th International Conference on Rough Sets, Fuzzy Sets, Data Mining, and Granular Computing, Lecture Notes in Artificial Intelligence 4482, Springer, Toronto, Canada, May 14-16, 2007, 182-189.

YAO & ZHONG, 2002. Granular Computing using Information Tables, in: Lin, T.Y., Yao, Y.Y., and Zadeh, L.A. (eds.) Data Mining, Rough Sets and Granular Computing, Physica, Heidelberg, 102-124.

ZADROZNY, 2009. Zadrożny S., Tré G. Guest Editorial: The Application of Fuzzy Logic and Soft Computing in Information Management, Fuzzy Sets and Systems, v.160 n.15, p.2117-2119.

ZADEH, 1965. Fuzzy Sets. *Information and Control*. 1965; 8: 338–353.

ZADEH, 1979. Zadeh, L. A. (1979) Fuzzy sets and information granularity, Advances in Fuzzy Set Theory and Applications, M Gupta, R. Ragade and R. Yager (eds.), North-Holland Publishing Co., 3-18.

ZADEH, 1996 Zadeh, L. A. (1996) Fuzzy Logic = Computing with Words, IEEE Transactions on Fuzzy Systems, 2, 103-111.

ZADEH, 1997. Zadeh, L.A. Towards a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic, Fuzzy Sets and Systems 19: 111-127.

ZHA, 2001. Zha H. Gu M., Ding C., He X., Simon H. Spectral Relaxation Models and Structure Analysis for K-Way Graph Clustering and Bi-Clustering. Technical Report, Penn State Univ. Computer Science and Engineering.

ZHANG, 2003. Zhang, L. and Zhang, B. (2003) The quotient space theory of problem solving, Lecture Notes in Computer Science 2639: 11-15, Springer, Berlin.

ZHONG, 2005. Zhong S. Efficient online spherical k-means clustering. Proc.IEEE Int. Joint Conf. Neural Networks. July 31 - August 4, 2005.Montreal, Canada.

ZHONG & YAO, 2008. Zhong N, Yao Y, Qin Y, Lu S, Hu J, Zhou H. Towards Granular Reasoning on the Web. In Proceedings of the 2008 Workshop on New forms of Reasoning for the Semantic Web: scalable, tolerant and dynamic (NEFORS2008), the 3rd Asian Semantic Web Conference (ASWC2008).