UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Campus de Presidente Prudente

FCT ¿ Faculdade de Ciências e Tecnologia
DMEC ¿ Departamento de Matemática, Estatística e Computação
Bacharelado em Ciência da Computação

Revisão Bibliográfica

Orientador: Prof. Dr. Danilo Medeiros Eler

2013

Sumário

Sumário 2

Lista de ilustrações 3

1 INTRODUÇÃO 5

1.1 Objetivos do Trabalho 6

1.2 Organização deste Trabalho 6

2 RECUPERAÇÃO DE INFORMAÇÃO 7

2.1 Introdução 7

2.2 Modelo Vetorial 8

2.2.1 Métricas de Ocorrência nos Documentos 10

2.2.2 Lei de Zipf - Corte de Luhn 12

2.3 Medidas de Dissimilaridade entre Documentos 12

2.3.1 Produto Interno 13

2.3.2 Coeficiente de Dice 13

2.3.3 Euclidiana 13

2.3.4 Lei do Cosseno 14

2.3.5 City Block 14

2.3.6 Overlap co-efficient 15

2.3.7 Jaccard Distance 15

2.3.8 Extended Jaccard 15

2.3.9 The Pearson r correlation 16

2.4 Considerações Finais 16

3 VISUALIZAÇÃO DE INFORMAÇÃO 17

3.1 Introdução 17

3.2 Técnicas de Projeções Multidimensionais 18

3.2.1 FastMap 18

3.2.2 NNP - Nearest Neighbor Projection 19

3.2.3 Least Squares Projection - LSP 20

3.2.4 Neighboor Joining - NJ 20

3.2.5 Interactive Document Map - IDMap 22

3.3 PEX 22

4 CONCLUSÃO 25

REFERÊNCIAS 27

Lista de ilustrações

Figura 1 – Procedimento para recuperação de informação em coleções de documentos 8
Figura 2 – Exemplo da Lei de Zipf(esquerda) e do Corte de Luhn(direita) 12
Figura 3 – Exemplos de Posicionamento com o Algoritmo NNP - Extraído de [TEJADA E.; MINGHIM 2003] 19
Figura 4 – Projeções pelo método LSP realizada na ferramenta PEx utilizando o conjunto de dados CBR-ILP-IR utilizando a métrica de dissimilaridade Lei do Cosseno no pré-processamento 21
Figura 5 – Projeções pelo método IDMAP realizada na ferramenta PEx utilizando o conjunto de dados CBR-ILP-IR utilizando a métrica de dissimilaridade Lei do Cosseno no pré-processamento e a técnica FastMap como primeira fase da projeção 22
Figura 6 – Exemplo - Interface da ferramenta PEX 23

Capítulo 1

Introdução

Esta revisão bibliográfica apresenta uma fundamentação teórica sobre a recuperação de informação em documentos textuais, destacando as técnicas que são utilizadas para o cálculo de dissimilaridade entre os documentos, demonstrando o procedimento que é usado em cada técnica. Destaca-se ainda a visualização de informação que dará o suporte à esses resultados obtidos, onde cada resultado obtido será representado através de um plano do espaço através do posicionamento de pontos, que auxiliará no entendimento dos resultados.

A Recuperação de Informação é um ramo da ciência que tem por objetivo pesquisar dados em documentos e recuperar informações relacionadas a eles. A partir de uma coleção de documentos, é possível realizar uma seleção de elementos (documentos) e procurar por documentos relacionados na mesma coleção ou em coleções distintas.

Para que técnicas de Recuperação de Informação sejam aplicadas em coleções de documentos, é necessário realizar algumas etapas de pré-processamento. Por exemplo, pode-se construir uma matriz de documentos e seus respectivos termos, armazenando também a frequências destes termos em cada documento. Neste processo, há uma etapa de eliminação de termos que não são interessantes para discriminar os documentos (e.g., artigos, pronomes, preposições). Para isso, uma lista deve ser construída com esses possíveis termos que não são relevantes. Essa lista é conhecida como lista de stop words. Adicionalmente, com os termos relevantes, outro tratamento pode ser efetuado para que os termos sejam reduzidos aos seus radicais. Esse processo é conhecido como stemming.

Com esta matriz de documentos criada, a análise da similaridade entre os documentos pode ser realizada, a partir de algumas métricas que calculam a distância entre os documentos a partir da ocorrência das mesmas palavras em ambos documentos. Nesse cálculo, o resultado é dado numa escala de 0 à 1, e quanto mais próximo de 1, mais similar são os documentos.

Neste projeto propomos utilizar os conceitos e técnicas acima para recuperação de documentos de coleções distintas, buscando similaridade entre os conteúdos dos documentos. Esse processo de recuperação será apoiado por técnicas de visualização de informação para

construção de mapas de similaridade por meio de posicionamento de pontos no plano, os quais representam os documentos. Assim, o usuário pode aproveitar tirar proveito tanto das técnicas de Recuperação de Informação quanto das técnicas de visualização, ou seja, a visualização exibirá os resultados em formas gráficas, mostrando os elementos da coleção de acordo com a recuperação de informação.

Convém realçar que o desenvolvimento do projeto me ajudou a ter um contato com ferramentas que colaboraram para a edição do presente documento, entre elas o LaTeX(processador de textos científico).

1.1 Objetivos do Trabalho

Esse projeto tem como objetivo principal demonstrar os resultados obtidos pelas medidas de dissimilaridade entre documentos, através da Visualização de Informação. Estes dados serão exibidos em conjuntos de pontos no espaço - que será determinado, de forma que quanto maior a proximidade entre os pontos - que representarão cada documento, maior será a semelhança quanto ao conteúdo entre os documentos. A visualização além de posicionar os pontos no espaço para demonstrar a proximidade entre os conteúdos, destacará na região do documento que foi selecionado, os documentos (pontos, no caso) que são mais similares.

1.2 Organização deste Trabalho

Neste trabalho apresentaremos além deste capítulo de introdução, mais 3 capítulos. O primeiro, destacará o processo da recuperação de informação, e o segundo, a representação de dados apoiada à visualização de dados.

No terceiro será feito um comentário sobre o que foi escrito nos capítulos anteriores e também será feita uma analise de como será feita as próximas etapas do projeto.

Capítulo 2

Recuperação de Informação

2.1 Introdução

Com o crescimento dos meios de informações, da tecnologia e do interesse e necessidade humana de buscar informações nos mais diversos meios de informação, uma tarefa que está em crescimento e passando pelas mais diversas pesquisas, é a Recuperação de Informação. Informações estão armazenadas atualmente em um grande número de elementos, donde podemos citar documentos, artigos, a Internet, dispositivos móveis, imagens, áudio e, a partir desses meios que armazenam informações, técnicas são pesquisadas para recuperar informações a fim de classificá-las de acordo com a necessidade.

A Recuperação de Informação busca encontrar dados em um conjunto de elementos, a fim de classificá-los e retirar informações particulares destes elementos. Essa característica é o que mais evidência a diferença entre a recuperação de informação e os sistemas de bancos de dados.

Sistemas Gerenciadores de Banco de Dados, ou SGBDs, são ferramentas utilizadas para retornar após uma consulta, todos os elementos - com todos os seus itens, presentes no banco de dados que satisfaçam a operação. Já a recuperação de informação extrai dados particulares dos elementos pertencentes a coleção/seleção que satisfaçam ao usuário do sistema de recuperação. [FERNEDA 2003]

Na recuperação de informações através de uma coleção de documentos, alguns procedimentos e técnicas devem ser aplicadas para a extração da informação. Neste tipo de recuperação, a busca pela informação se baseia pelo contexto das palavras-chaves passadas pelo usuário, ou até documentos que são analisados realizando uma comparação entre seus conteúdos.

Como neste trabalho visamos a recuperação de informação em coleções de documentos a partir de documentos, esta revisão visará destacar os procedimentos e as características deste tipo de recuperação de dados.

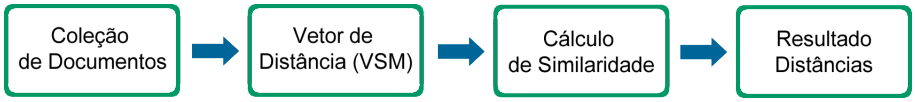

Figura 1 – Procedimento para recuperação de informação em coleções de documentos

O processo de recuperação de informação se inicia - como visto na Figura 1, com a identificação do documento ou da coleção de documentos que serão analisados na recuperação de informação. Nesta etapa são identificados os documentos que serão utilizados como 'modelos', e os documentos que serão comparados à estes modelos definidos pelo usuário do sistema de recuperação de informação. Após esta etapa de identificação dos elementos pertencentes a recuperação, se inicia o processo que formará o vetor de ocorrência e por sequência, de similaridade entre os documentos.

Na Seção 2.2 teremos uma descrição detalhada do processo que cria o modelo vetorial, ou Modelo do Espaço Vetorial (VSM - Vector Space Model). Na Subseção 2.2.1 destacaremos o processo que define a ocorrência dos termos analisados no modelo vetorial.

O processo que define o resultado final, a similaridade entre os documentos, será destacado na Seção 2.3.

2.2 Modelo Vetorial

Com a identificação dos documentos que farão parte dos objetos de análise, se inicia o processo de reconhecimento e armazenamento do conteúdo de cada um destes documentos. Como a recuperação se baseia em similaridade por conteúdo, este procedimento de armazenar os termos relevantes de cada documento tem uma importância muito significativa. Caso um termo seja descartado incorretamente, o usuário do sistema de recuperação poderá receber respostas que não satisfaçam a busca.

No Modelo Vetorial, ou modelo de espaço vetorial (VSM -Vector Space Model), os documentos são representados em uma matriz com n-documentos, onde cada linha desta matriz é um vetor representando cada documento da coleção. A matriz representa nas colunas os termos relevantes de todos os documentos da coleção. [RAGHAVAN]

Neste processo destaca-se que os termos são adicionados a matriz conforme a sua descoberta nos documentos, fazendo com que todos os vetores tenham o mesmo número de atributos para se manterem em igualdade para a comparação de similaridade a ser realizada posteriormente. Se após o processamento do documento X, for descoberto um termo relevante *'a1'* no documento Y, então o campo *'a1'* é adicionado na matriz, fazendo com que todos os documentos que já foram processados também possuam este campo. Neste caso, a ocorrência deste termo neste documento será nula ou vazia.

Conforme os termos de cada documento são identificados e adicionados ao vetor correspondente, a matriz do modelo vetorial fica composta pelas colunas - que representam

os termos, as linhas - que representam os documentos, e o valor daquela posição no vetor, ou na matriz. Este valor, é o número de ocorrência deste termo no documento analisado.

Abaixo temos um exemplo de uma matriz formada por um conjunto X de documentos D, sendo D = $\{d_1,d_2,...,d_X\}$, uma quantidade N de termos T na matriz, donde T = $\{t_1,t_2,..t_N\}$, e a composição de cada elemento de D, onde cada posição $\alpha_{i,j}$ definem a frequência de cada termo no documento d_x analisado.

	t_1	t_2	t_3	..	t_N
d_1	$\alpha_{1,1}$	$\alpha_{1,2}$	$\alpha_{1,3}$	..	$\alpha_{1,N}$
d_2	$\alpha_{2,1}$	$\alpha_{2,2}$	$\alpha_{2,3}$	..	$\alpha_{2,N}$
d_3	$\alpha_{3,1}$	$\alpha_{3,2}$	$\alpha_{3,3}$	..	$\alpha_{3,N}$
..	..	..	..	..	..
d_X	$\alpha_{X,1}$	$\alpha_{X,2}$	$\alpha_{X,3}$	..	$\alpha_{X,N}$

Tabela 1 – Exemplo da Matriz VSM - Modelo Vetorial

Para começar este procedimento é necessário separar todos os termos de cada documento, para que os mesmos passem pelas análises. Este processo chamado de **tokenização** separa cada termo do documento para que sejam tratados separadamente. Esta característica impossibilita que expressões sejam recuperadas, pois cada palavra que for encontrada, será tratada separadamente. Como este projeto visa recuperação a partir de documentos, busca com expressões não irão ocorrer, e não trarão nenhum problema ao resultado final. [MANNING C.D.; RAGHAVAN 2008]

Para que apenas os termos com relevância sejam adicionados ao modelo vetorial, alguns procedimentos são adotados para que alguns termos sejam eliminados. Esses termos podem atrapalhar o processamento da recuperação por não terem importância no contexto do documento, terem alta ocorrência - o que pode indicar que este termo tem importância apenas a este documento, ou pelo próprio contexto do termo (preposições por exemplo) que não é relevante a recuperação.

Como foi dito, termos que são classificados como preposições, artigos, pronomes, advérbios não são adicionados ao VSM para que seja reduzida a dimensionalidade da matriz do modelo vetorial, para que estes termos não alterem o resultado final da recuperação e também para que os mesmos não atrapalhem no processamento, já que os cálculos deixarão de ser aplicados à estes termos. Este procedimento é chamado de '**remoção de stop words**'.

Para exemplificar o método, temos abaixo um exemplo:

Entrada: 'O rato roeu a roupa do Rei de Roma'.

Termos Relevantes: 'rato roeu roupa Rei Roma'.

Após eliminar as palavras que apenas aumentarão a dimensão do modelo vetorial, outro procedimento pode ser aplicado para reduzir o número de palavras com contextos

semelhantes. O '**stemming**' é um algoritmo que é executado para reduzir duas palavras com o mesmo radical, para um termo só, que no caso, é o próprio radical. Este processo como o de '*stop words*' reduz a dimensão do modelo vetorial, pois elimina ocorrências repetidas do mesmo radical. [BASSIL 2012]

Para exemplificar o método, temos abaixo um exemplo:

Entradas: 'pesca', 'pescaria', 'pescador', 'pescado'.

Termos Relevantes: 'pesc'.

Os processos de remoção das *stopwords* e do *stemming* ocorrem baseados em dicionários particulares de cada linguagem. Isto significa que este processo ocorre apenas se a ferramenta de recuperação de informação dar suporte à língua em que os elementos da coleção estão escritos. Existem dicionários para ambas operações. Os dicionários para *stopwords* são mais fáceis de ser encontrados ou criados já que além de possuírem um número menor de termos, eles não envolvem nenhuma lógica no seu contexto.

Os dicionários para *stemming* são mais difíceis de serem encontrados para algumas línguas já que é um dicionário mais complicado de ser criado. Os melhores criados, segundo estudos, foram para a língua inglesa. Este dicionário é mais difícil de ser encontrado já que um bom dicionário tem que saber diferenciar, para algumas linguagens como o português, mudanças nos radicais de alguns termos com o mesmo contexto semântico.

Veja o exemplo abaixo para a língua portuguesa:

Entrada: 'Viajar' 'Viagem'.

Neste caso surge a dúvida entre o radical ser 'Viaj' ou 'Viag', e isto o dicionário tem que analisar e identificar para um melhor agrupamento dos radicais.

Com a dimensionalidade dos termos reduzida, e a identificação de todos os termos de cada documento, teremos a matriz de ocorrência dos documentos da coleção montada e preparada para o cálculo de similaridade. Até o momento foi destacado o processo de identificação dos termos que irão indexar, ou compor a matriz do modelo vetorial (VSM).

No mesmo momento que é realizada a análise dos termos, a frequência de cada termo é calculada e por isso, as análises acabam juntas. No processo de identificação dos termos relevantes, caso um termo já está indexado, ou presente, na matriz, a frequência deste termo é atualizada de acordo com a métrica utilizada.

Neste momento, a ferramenta de recuperação de informação aplicará algoritmos que serão destacados nas próximas sub-seções; a métrica de ocorrência, e também os Algoritmos da Lei de Zipf e o Corte de Luhn.

2.2.1 Métricas de Ocorrência nos Documentos

Para o cálculo de ocorrência, ou frequência, de um termo no documento, algumas métricas foram desenvolvidas.

- Frequência do Termo

  Com uma lógica muito simples, este método se baseia em contabilizar o número de vezes que o termo apareceu no documento. Quanto maior o número, maior foi a ocorrência deste termo no documento. O valor ser maior que 0 (zero) indica também que o termo existe no documento.

- Booleana

  A métrica booleana é uma das métricas mais simples que existem, pois ela descreve com valores booleanos (1 e 0), a existência do termo no documento. Se o termo existir no documento, o campo frequência é preenchido com 1. Caso não exista, o campo da frequência é preenchido com 0.

$$\alpha_{i,j} = 1 \text{ ou } \alpha_{i,j} = 0$$

- Term Frequency Inverse Document Frequency (tf-idf)

  Alguns termos podem ser frequentes em muitos documentos, ou ao contrário, podem ter um índice de ocorrência muito baixo. Com essa característica, esses termos não são relevantes à recuperação de informação, já que eles descrevem por este ponto de vista, particularidades de alguns documentos. Um termo que está presente em muitos documentos, será incapaz de diferenciá-los, assim como um termo que aparece poucas vezes na coleção, será de pouca relevância na diferenciação dos documentos. [ALENCAR 2013]

  Para isso temos o fator *idf* que estima a frequência favorecendo termos que não apareceram tanto na coleção de documentos. Um termo que aparece muito no documento, tem um idf baixo, já um termo que quase não é encontrado no documento possui um idf alto.

$$idf = \log \frac{N}{d(t_j)}$$

  N é o número de documentos da coleção,

  $d(t_j)$ é o número de documentos que contêm o termo.

  Para obtermos a frequência do termo pelo inverso do documento(tf-idf), precisamos combinar o fator *idf* com a frequência do termo ($\alpha_{i,j}$).

$$tf - idf(t_j, d_i) = tf(t_j, d_i) \times idf(t_j) = \alpha_{i,j} \times \log \frac{N}{d(t_j)}$$

2.2.2 Lei de Zipf - Corte de Luhn

Afim de que a dimensionalidade da matriz de termos seja reduzida, alguns algoritmos ainda são executados para eliminarem termos que não ajudarão na comparação entre os documentos por terem uma frequência muito alta ou muito baixa na coleção de documentos. [ALENCAR 2013]

Lei de Zipf

A lei de Zipf, ou Curva de Zipf (George K. Zipf - 1940) é uma lei matemática que calcula dimensões, frequências de elementos. O resultado é armazenado ordenadamente em uma lista onde os termos são ordenados pelos que possuem maior frequência até os que são raros ou pouco frequentes. Na recuperação de informação essa lei pode ser aplicada buscando identificar e ordenar os termos que mais se destacaram na recuperação.

Corte de Luhn

O Corte de Luhn é um algoritmo utilizado na recuperação de informação com o auxílio da Lei de Zipf. O Corte de Luhn estabelece limites inferiores e superiores sobre o resultado de Lei de Zipf. Estes cortes eliminam os termos que tiveram uma frequência alta no número de ocorrências - significando que não há como diferenciar os documentos já que o termo é comum à todos, e também aqueles que não apareceram tanto nos documentos, que com isso, não irão diferenciar os documentos já que são particulares a poucos documentos na coleção.

Figura 2 – Exemplo da Lei de Zipf(esquerda) e do Corte de Luhn(direita)

Com este corte podemos notar que na Curva de Zipf, ou Lei de Zipf, os termos mais importantes e relevantes se encontram normalmente no meio do conjunto.

2.3 Medidas de Dissimilaridade entre Documentos

Medidas de similaridade são métricas usadas para demonstrar quanto um documento é parecido ao outro através de valores numéricos. Esses valores são sempre iguais ou maiores

que 0(zero), onde quanto maior for o valor, maior será a similaridade entre os objetos.

Para cálculos envolvendo documentos no espaço vetorial, temos que assumir uma regra importante para o cálculo. Dados dois elementos x e y, e n sua dimensionalidade, os objetos x e y serão representados no espaço vetorial da forma abaixo: (Zhang, 2008).

$$\mathrm{x} = (x_1, x_2, x_3, .., x_n)$$
$$\mathrm{y} = (y_1, y_2, y_3, .., y_n)$$

2.3.1 Produto Interno

Na medida do produto interno, o valor de similaridade é obtido através do produto simples entre os objetos, sendo que só serão calculadas as dimensões que ambos objetos possuírem. Isto quer dizer que se um objeto possuir uma ocorrência e o outro a ser comparado, não possuir, esta posição do objeto será ignorada e o cálculo irá para a próxima dimensão.

Esta medida é um tanto quanto tendenciosa já que dado um objeto que possuir 100 dimensões, e outro com 30 dimensões, levando em considerações que todas as dimensões do segundo objeto estão também presentes no primeiro, este resultado segundo esta medida, será de 100% compatível, sendo que o correto seria 30%, já que as outras 70 dimensões do objetos não existem e não foram analisadas.

A medida de similaridade deste método é dada pela fórmula abaixo:

$$\mathrm{D(a,b)} = \sum_{i=1}^{N} a_i \times b_i$$

2.3.2 Coeficiente de Dice

O coeficiente de Dice foi desenvolvido a partir da medida do Produto Interno, para tentar normalizar os resultados e fazer com que o 'erro' apresentado na medida do Produto Interno seja corrigido.

Para isto é adicionado um denominador na medida do Produto Interno, para que todas as dimensões sejam levadas em consideração;

O algoritmo é descrito abaixo:

$$\mathrm{D(a,b)} = \frac{2\sum_{i=1}^{N} a_i \times b_i}{\sum_{i=1}^{N} a_i + \sum_{i=1}^{N} b_i}$$

2.3.3 Euclidiana

A distância Euclidiana se baseia na equação matemática que extrai um valor a partir da raiz quadrada da soma dos quadrados das diferenças entre os pontos. Isto para a recuperação de documentos implicaria em comparações entre cada dimensão dos objetos analisados.

Sendo x = $(x_1, x_2, x_3, .., x_n)$ e y = $(y_1, y_2, y_3, .., y_n)$, a comparação seria a raiz da somatória $((x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 + ... + (x_n - y_n)^2)$.

O algoritmo é descrito abaixo:

$$\mathrm{D(a,b)} = \sqrt[2]{\sum_{i=1}^{N}(a_i - b_i)^2}$$

da mesma forma,

$$\mathrm{D(a,b)} = \sqrt[2]{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 + ... + (x_n - y_n)^2}$$

2.3.4 Lei do Cosseno

A lei do Cosseno é uma medida de similaridade que encontra a distância a partir do ângulo que os dois objetos formam no plano. Dado dois objetos, o cálculo de sua similaridade é o cosseno do ângulo que existe entre os objetos no espaço vetorial. Com isso, os valores obtidos serão sempre entre os valores 0 e 1 que são o intervalo do cosseno.

Este método reconhece melhor a similaridade em objetos porque acaba analisando melhor a distribuição de pesos nos objetos. A medida é dada pela divisão do produto interno pela raiz da soma dos quadrados de cada termo dos objetos.

O algoritmo é descrito abaixo:

$$\mathrm{D(a,b)} = \frac{\sum_{i=1}^{N} a_i \times b_i}{(\sum_{i=1}^{N} a_i^2 \times \sum_{i=1}^{N} b_i^2)^{\frac{1}{2}}}$$

2.3.5 City Block

A distância City Block, ou Manhattan, é um algoritmo que calcula a distância entre os pontos de uma maneira bem simples. O calculo é realizado apenas com a somatória das diferenças entre os pontos. Lembrando que as diferenças são sempre positivos (módulo).

Prestando mais atenção no algoritmo podemos notar que é bem semelhante ao algoritmo Euclidiano, onde para o City Block o valor de p na equação abaixo é 1, e no caso do Euclidiano, p receberia o valor 2.

$$(\sum_{i=1}^{N} |a_i - b_i|^p)^{\frac{1}{p}}$$

A partir destas analises temos então o algoritmo City Block descrito abaixo:

$$\mathrm{D(a,b)} = \sum_{i=1}^{N} |a_i - b_i|$$

2.3.6 Overlap co-efficient

Este algoritmo é baseado também no algoritmo do Produto Interno, e muito similar ao Coeficiente de Dice. A mudança vem no denominador da equação que nos levará a medida de similaridade.

Neste algoritmo, o denominador é o mínimo entre as somas de pesos entre cada objeto analisado. O algoritmo está descrito abaixo:

$$\mathrm{D(a,b)} = \frac{\sum_{i=1}^{N} a_i \times b_i}{MIN(\sum_{i=1}^{N} a_i, \sum_{i=1}^{N} b_i)}$$

2.3.7 Jaccard Distance

A distância de Jaccard é calculada levando em consideração dimensões que são presentes em ambos objetos. O cálculo tem base no conceito de conjuntos matemáticos. O coeficiente de similaridade é dado pela divisão dos atributos pertencentes em ambos objetos pela união de todas os atributos (dimensões) existentes.

$$\mathrm{J(a,b)} = \frac{|a \cap b|}{|a \cup b|}$$

A partir deste coeficiente, a distância de Jaccard é dada subtraindo o coeficiente de Jaccard de 1. A medida de similaridade é dada abaixo:

$$D_j(a,b) = 1 - \frac{|a \cap b|}{|a \cup b|}$$

2.3.8 Extended Jaccard

A medida Extended Jaccard é derivada da Jaccard Distance diferindo apenas em seu denominador.

Na Jaccard Distance o denominador leva em consideração os elementos presentes em ambos objetos. No algoritmo Extended Jaccard, o denominador é definido pela soma do quadrado dos atributos(dimensões) pertencentes a cada objeto subtraindo os elementos que são pertencente em ambos.

Este algoritmo leva em consideração objetos com valores binários (e.g: se uma dimensão está presente ao objeto, será identificado por 1, senão 0). Como o vetor é identificado como um vetor binário, a soma dos quadrados abaixo acaba ficando igual a soma simples já que o quadrado de 1 permanece 1. Então temos:

$$\|a\|^2 = \sum_{i=1}^{N} a_i^2 = \sum_{i=1}^{N} a_i$$

O algoritmo é dado por:

$$D(a,b) = \frac{a \cdot b}{\|a\|^2 + \|b\|^2 - a \cdot b}$$

2.3.9 The Pearson r correlation

Esta medida avalia os objetos baseados na força da relação entre duas varáveis que foram assumidas [ZHANG]. Esta força é encontrada através de uma equação que leva em consideração a média de valores de cada variável assumida.

Na equação, o numerador é composto por uma somatória do produto da diferença entre o valor da posição atual do objeto e a média de ocorrência do objeto. No denominador temos o produto das raízes de cada objeto, sendo o produto formado pela diferença entre o valor do objeto e a média de ocorrência dos valores no mesmo objeto.

Os valores da equação são dados entre -1 e 1, e por isso, o resultado da medida é obtido pelo valor absoluto.

Abaixo temos a equação:

$$D(a,b) = \frac{\sum_{i=1}^{N}(a_i - \frac{\sum_{i=1}^{N} a_i}{n}) \times (b_i - \frac{\sum_{i=1}^{N} b_i}{n})}{\sqrt[2]{\sum_{i=1}^{N}(a_i - \frac{\sum_{i=1}^{N} a_i}{n})} \times \sqrt[2]{\sum_{i=1}^{N}(b_i - \frac{\sum_{i=1}^{N} b_i}{n})}}$$

2.4 Considerações Finais

Após a construção do modelo vetorial e da aplicação de uma medida de similaridade na coleção, teremos a matriz de distâncias dos objetos. Lembrando que esta matriz de distâncias é totalmente diferente da matriz que foi criada no modelo vetorial. Naquela etapa a matriz guardava os termos de cada documento, já na matriz de distâncias guardaremos as distâncias provenientes das medidas de similaridades entre cada objeto.

Essa matriz possui uma dimensão n x n, onde n é o numero de documentos que foram selecionados para a comparação, incluindo os documentos que são os modelos para a comparação. Isto implica que o cálculo de similaridade será aplicado $n \cdot n$ vezes, o que traz uma complexidade $\sigma(n^2)$ ao processo.

Este processo é realizado desta forma para que o usuário possa saber a distância de cada objeto em relação aos outros. Caso o usuário só queira saber a distância entre um objeto e os outros elementos da coleção, a complexidade é $\sigma(n)$, mas caso o usuário queira utilizar ferramentas de visualização de informação por exemplo, será necessário ter uma matriz $n \cdot n$ para que os elementos possam ser representados em relação ao conjunto todo.

No próximo capítulo será descrito técnicas de projeções multidimensionais que utilizaram a matriz de distâncias para executarem as projeções.

Capítulo 3

Visualização de Informação

3.1 Introdução

A Visualização de Informação é um ramo da computação que busca a representação de dados visualmente, de forma que fique mais compreensível a visualização e o entendimento dos dados pelos usuários.

A compreensão dos objetos num ambiente de visualização de informação, faz com que os usuários analisem com maior rapidez, intuitividade e precisão o conjunto analisado, reduzindo o esforço que seria necessário em uma análise tradicional dos dados. O reconhecimento de padrões também acontece com a visualização, o que melhora no geral o entendimento da coleção.

Atualmente alguns objetos podem carregar consigo um número grande de atributos, e devido a grande dimensão destes objetos e o volume destes dados, a representação destes conjuntos se torna uma tarefa complicada. O fluxo e a dimensão dos dados se alteram constantemente, e com isso a representação se torna um desafio já que a capacidade humana de analisar e identificar todas as informações disponíveis permanecem a mesma. [SALAZAR 2012]

Por estas razões, a visualização e a recuperação de informação - que tratamos no capítulo 2, possuem uma ligação muito grande. Técnicas de visualização de informação são combinadas à recuperação de informação para que os dados sejam expostos de forma que suas características, assim como alguns padrões sejam identificados de um jeito mais claro pelos usuários.

A criação do espaço de visualização pode utilizar itens como pontos, linhas, direções para representar no plano o posicionamento e/ou as distâncias entre os objetos. Isto pode facilitar também a descrição do conteúdo, relação e contexto dos objetos.

A utilização de cores nos itens pode também ajudar na visualização. Mudanças de cores e da composição delas (saturação, brilho) podem ajudar a diferenciação dos itens no conjunto representado.

Estas técnicas podem ser aplicadas por exemplo em coleções de documentos. A Visualização neste caso trabalharia com técnicas de posicionamento de pontos no espaço, onde os pontos representariam os documentos, e a distância entre cada ponto seria retirado dos cálculos de dissimilaridade entre os documentos, como visto no no capítulo 2 na subseção 2.3.

Sendo assim, pontos que estejam próximos representam documentos que são similares quanto ao conteúdo segundo a recuperação de informação. Pontos que estejam mais distantes, logo, representam documentos que não possuem conteúdos semelhantes.

A distância entre os pontos dependem também das técnicas de recuperação que foram utilizadas, onde cada uma pode alterar bastante a visualização.

3.2 Técnicas de Projeções Multidimensionais

Para que os documentos que foram analisados na recuperação de informação sejam representados através da visualização de informação, os dados provenientes da recuperação são utilizados por técnicas de Projeções Multidimensionais.

Essas técnicas buscam receber - das técnicas de Recuperação de Informação, os dados com n-dimensionalidade e transformá-los - para que possam ser representados em um plano t-dimensional, onde t = {1,2,3}, para que a representação possa ser mais clara, intuitiva e que represente da melhor forma o conjunto de dados no plano.

Com o conjuntos de dados provenientes da recuperação - VSM, teremos o conjunto de documentos com todas as ocorrências dos termos nos documentos. Juntando estes dados com a matriz de distância da coleção, como visto no capítulo anterior, teremos a distância de cada documento da coleção para todos os outros.

A técnica então busca posicionar os itens no plano bi ou tridimensional de forma que fiquem posicionados de acordo com as distâncias calculadas, sendo que o posicionamento respeite também o conteúdo da vizinhança, a partir do VSM. A visualização deve respeitar a distância posicionando os documentos de forma que estejam respeitando a distância em relação a todos os documentos da coleção.

Para isso são realizados alguns processamentos para que um VSM com uma dimensão alta, seja reduzido até que chegue a dimensão do plano, que geralmente é 2 ou 3. A seguir estaremos destacando algumas técnicas de projeção multidimensaional.

3.2.1 FastMap

O algoritmo de projeção FastMap busca realizar a projeção de um conjunto de objetos com dimensão D em um espaço com dimensão D_1, sendo $D_1 \leq$ D. [FALOUTSOS C.; LIN] O processo começa com a seleção de 2 objetos, chamados pivôs, sendo que para o segundo vale a regra de que o mesmo deve ser o mais distante possível em relação ao primeiro.

Após a seleção destes dois pivôs, o algoritmo traça uma reta entre eles, o que acaba formando um hiperplano perpendicular a reta. Os outros objetos do conjunto são posicionados neste hiperplano que tem a dimensão D - 1. O processo é repetido até que a dimensão da representação seja igual à D_1. [VALDIVIA 2007]

Para calcular a posição dos itens entre os dois pivôs, a Lei dos Cossenos pode ser utilizada. Dado os pivôs A_1 e A_2, o ponto A_i será dado a partir da equação:

$$\delta^2(A_2, A_i) = \delta(A_1, A_i)^2 + \delta(A_1, A_2^2 - 2x_1\delta(A_1, A_2)$$

onde a posição do objeto é dada por:

$$x_1 = \frac{\delta(A_1,A_i)^2+\delta(A_1,A_2)^2-\delta(A_2,A_i)^2}{2\delta(A_1,A_2)}$$

Este algoritmo não é recomendado para coleções de dados com alta dimensionalidade, porque sua representação sofre com a perda de informação e com isso, o relacionamento entre os objetos pode ser afetado. Segundo Platt [PLATT], esta perca ocorre porque o método realiza (D - D_1) iterações

3.2.2 NNP - Nearest Neighbor Projection

O método NNP criada por pesquisadores do ICMC, projeta dados n-dimensionais no plano bidimensional sem precisar de nenhuma outra váriavel exceto a matriz de distância dos objetos. O algoritmo posiciona os pontos no plano a partir de 2 pontos que já foram posicionados anteriormente.

O algoritmo posiciona o ponto p do conjunto a partir de 2 pontos que estejam mais próximos de p e que já foram projetados no espaço. Após identificar os pontos que já foram projetados (x e y), o ponto p é posicionado na intersecção de dois círculos que são 'projetados' a partir dos pontos x e y, onde o raio dos círculos x e y é retirado do vetor de distância (dissimilaridade $\delta(x, p)$ e $\delta(y, p)$) entre os pontos(x e y) e o ponto a ser projetado. [ELER D.M.; MINGHIM 2007]

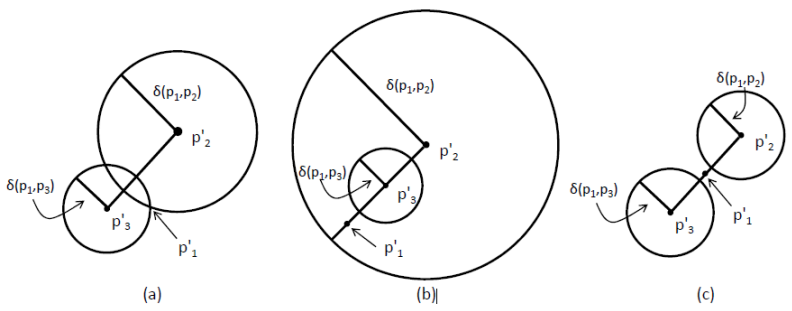

Figura 3 – Exemplos de Posicionamento com o Algoritmo NNP - Extraído de [TEJADA E.; MINGHIM 2003]

Caso os círculos tenham o ponto de intersecção, a posição do ponto p é definida na intersecção. Se os círculos não possuírem intersecção, o ponto p é posicionado num ponto médio entre os círculos. Se os círculos possuírem interessecção mas não forem tangentes, existem duas possibilidades sendo a solução escolhida ao acaso.

3.2.3 Least Squares Projection - LSP

Dado um conjunto de pontos com m-dimensionalidade, o algoritmo LSP busca representar estes pontos em um plano com uma dimensão x, com x $\leq$ m, sendo que as distâncias que representam a vizinhança entre os pontos sejam preservadas, oque representa a similaridade entre eles. [PAULOVICH F 2009]

A técnica LSP tem como objetivo encontrar posições no espaço R^d, onde é d foi definido previamente, para posicionar os itens da coleção. O processo possui duas fases de execução, onde a primeira é a busca e identificação dos chamados ***pontos de controle***, e na segunda fase o restante dos pontos da coleção são posicionados de acordo com a proximidade em relação aos pontos de controle.

No ínicio do processo, pontos de controle são definidos e projetados por alguma técnica de redução de dimensionalidade (MDS -Multidimensional Scaling), como a FastMap ou ForceScheme. Os pontos são definidos a partir de técnicas de clustering e mineração de dados, onde pequenos clusters são formados com os pontos, e cada cluster proporcionará um ponto de controle, sendo este ponto o centro geométrico deste agrupamento. [PAULOVICH 2008]

Após esta fase, o restante dos itens de dados são divididos em pequenos conjuntos, onde cada conjunto são pontos que estão próximos a cada ponto de controle. Com esses conjuntos de pontos, o algoritmo procura projetar os pontos de acordo com as coordenadas do ponto de controle, visando manter a distância(representando a similaridade) entre os itens e levando em consideração também o restante dos pontos que não estão neste conjunto local.

3.2.4 Neighboor Joining - NJ

A técnica Neighboor Joining (NJ) utiliza o conceito de árvores filogenéticas para a representação de pontos no espaço. Árvores Filogenéticas são usadas para representação porque podem mostrar com mais facilidade a evolução (ancestralidade) dos objetos em questão. As distâncias são representadas pelo tamanho das arestas que ligam cada nó da árvore, e cada folha (ramificação) da árvore representa uma nova espécie dos dados. [VALDIVIA 2007]

O algoritmo NJ utiliza a representação por árvores sem raiz, que representa apenas as distancias entre os nós mas não a ancestralidade. O processo consiste em encontrar dois nós que possuam o menor valor na soma das distância entre os ramos.

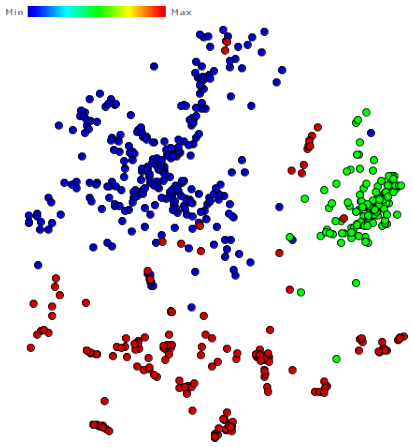

Figura 4 – Projeções pelo método LSP realizada na ferramenta PEx utilizando o conjunto de dados CBR-ILP-IR utilizando a métrica de dissimilaridade Lei do Cosseno no pré-processamento

Para encontrar os nós que possuam a menor distância entre os ramos, uma matriz de distância entre os ramos é criada onde será armazenado as distâncias entre cada nó da árvore. Após a matriz ser criada são encontrados os nós que possuam a menor distância entre os ramos e a partir deles, é definido um novo objeto X que ligará os objetos escolhidos.

Sendo os 2 nós com a menor distância a e b respectivamente, nesta etapa é calculada a distância dos ramos que ligarão a e b ao novo objeto X.

Subsequente a tudo isso, também será calculado a distância que ligará X aos outros objetos da árvore. Este processo é repetido até que a matriz possua apenas 2 elementos.

Abaixo temos o algoritmo completo.

1. Para cada objeto calcular a divergência da árvore,

$$\mathrm{r(i)} = D_{i1} + D_{i2} + D_{i3} + .. + D_{ij},$$

onde i $\neq$ j e j = 1,2,..,n

2. Calcular a nova matriz de distâncias com a equação:

$$M_{ij} = D_{ij} - \frac{[r_i + r_j]}{n-2}$$

3. Escolher os objetos i e j para s quais M_{ij} é mínimo e criar um novo objeto U (nó internet da árvore) que una os objetos i e j.

4. Calcular o tamanho do ramo que una o objeto U aos objetos i e j.

$$S_{iU} = \frac{D_{ij}}{2} + \frac{[r_i - r_j]}{2[n-2]}$$

$$S_{jU} = D_{ij} - S_{iU}$$

5. Calcular as distâncias entre os novo objeto U com os objetos restantes.

$$D_{kU} = \frac{D_{ik}+D_{jk}-D_{ij}}{2},$$

onde k $\neq$ i, k $\neq$ j e j = 1,2,..,n

6. n = n -1
7. Volte ao passo um da iteração enquanto que n > 2.

3.2.5 Interactive Document Map - IDMap

A técnica de projeção multidimensional IDMap foi desenvolvida para análise e projeção de documentos. A técnica busca projetar os documentos no plano como pontos e realiza a combinação de duas técnicas para tentar diminuir o erro de projeção dos dados. O erro de projeção é a distância que não foi possível ser preservada na representação. [MINGHIM R.; PAULOVICH 2006]

Para a representação por esta técnica são utilizados 2 métodos de posicionamento de pontos no plano. Inicialmente o método FastMap ou o NNP (Nearest Neighbor Projection) é aplicado ao conjunto criando assim uma projeção inicial. Como esta projeção inicial, a técnica IDMAP aplica a ForceScheme sobre estes pontos para que os pontos possam ser melhores posicionados na tentativa de reduzir erros do tipo pontos que ficaram muito próximos numa representação.

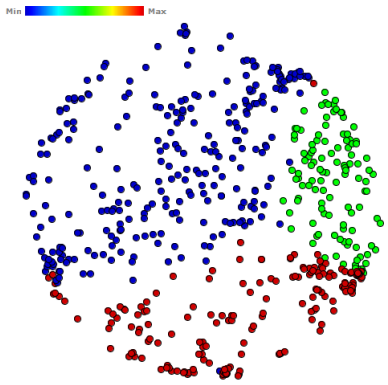

Figura 5 – Projeções pelo método IDMAP realizada na ferramenta PEx utilizando o conjunto de dados CBR-ILP-IR utilizando a métrica de dissimilaridade Lei do Cosseno no pré-processamento e a técnica FastMap como primeira fase da projeção

3.3 PEX

A ferramenta de visualização de informação PEX (Projection EXplorer) foi desenvolvida no ICMC - USP como parte do doutorado de Fernando Vieira Paulovich, e alguns outros pesquisadores contribuíram para o desenvolvimento da aplicação.

Ela foi desenvolvida na linguagem Java e é utilizada para criar visualizações de informação como descritos neste capítulo. A ferramenta é utilizada para representações a partir de coleções de documentos e cada objeto da coleção é representado na visualização por pequenos círculos. Utilizando conceitos citados acima, se um ponto está próximo ao outro, isto significa que os documentos são semelhantes quanto ao conteúdo.

Para que a ferramenta seja utilizada para dados que não sejam coleções de documentos, é necessário que métricas para cálculo de distância sejam passadas à ferramenta. Para qualquer tipo de visualização de informação gerada pela ferramenta, técnicas de recuperação de informação são aplicadas para o cálculo de distância entre os elementos.

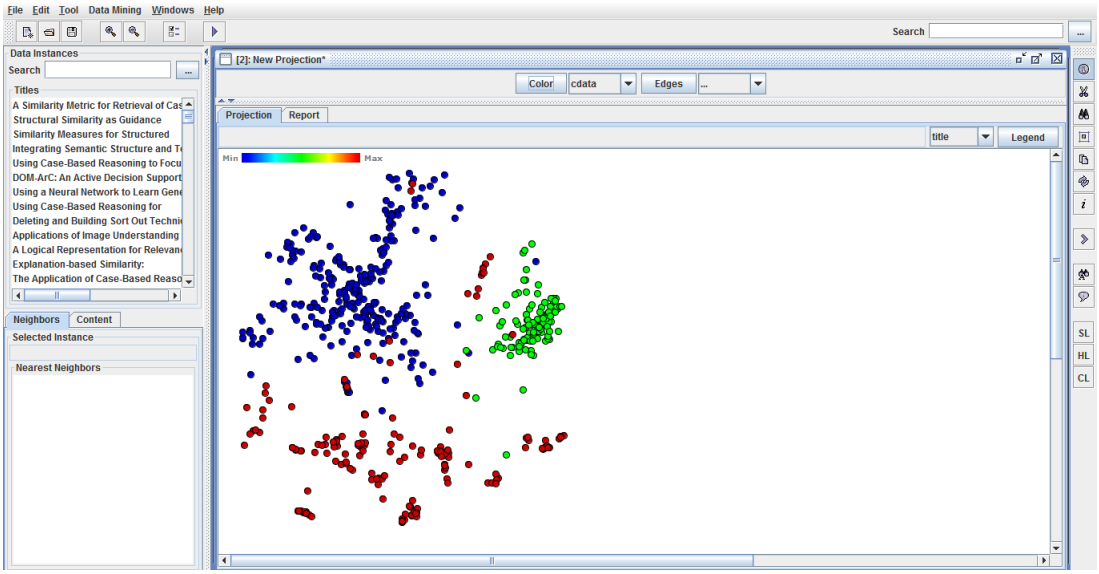

Figura 6 – Exemplo - Interface da ferramenta PEX

Para obter a matriz de distâncias que é o resultado final desta parte do processo, a ferramenta aplica os procedimentos aplicados no capítulo anterior, como a remoção de stopwords, aplicação de stemming e cortes de frequência. Os resultados obtidos são armazenados na matriz de distância.

Para realizar os cálculos de similaridade entre os documentos e criar a matriz de distâncias, a ferramenta possui algumas métricas como a Euclidiana, a Lei do Cosseno e City Block. A ferramenta também dispõe de algumas técnicas de projeções multidimensionais como as citadas neste documento LSP, IDMAP e NJ, como também várias outras que não foram citadas como a ISOMAP, ProjClus, Local Linear Embedding (LLE) dentre outras.

As ferramenta PEX disponibiliza ao usuário em suas projeções, algumas pequenas funcionalidades que podem ajudar ao usuário a identificar e compreender melhor as coleções. As labels são um exemplo destas pequenas funcionalidades, e com elas, o usuário pode visualizar para cada ponto(documento) na projeção informações que podem ser o nome do documento ou palavras-chaves do mesmo.

A representação da vizinhança entre os pontos pode ser visualizada com as ligações de arestas entre eles, ou com o destaque dos vizinhos, após o usuário selecionar algum ponto na projeção.

Existe a possibilidade também de vizinhanças serem destacadas e/ou selecionadas na projeção. Normalmente, as cores que são usadas para caracterizarem as distâncias entre os objetos, são também utilizadas pelas técnicas para separarem na projeção grupos de itens que formam pequenas vizinhanças, ou pequenos clusters. [PAULOVICH 2008]

Capítulo 4

Conclusão

Após a produção desta revisão bibliográfica que trouxe o conhecimento teórico tanto da área de recuperação de informação em coleções de documento quanto da área de visualização de informação para estes tipos de dados, este capítulo é reservado para uma descrição das atividades que serão desenvolvidas nas próximas etapas deste trabalho de graduação.

Na próxima fase, o foco do trabalho será o desenvolvimento e implementação de técnicas de similaridade. Essas técnicas serão adicionadas as medidas de similaridade já existentes na ferramenta PEX, que foi descrita no capitulo anterior.

Outra funcionalidade que será implementada, é uma maneira de destacar na visualização os pontos mais próximos a um documento selecionado na projeção.

A visualização implementada na ferramenta PEX já expõe através da posição dos pontos na projeção, as distâncias entre os pontos o que possibilita ao usuário entender a similaridade entre eles. Nesta proposta ainda temos a implementação de uma forma de analisar mais de um conjunto de documentos, em visualizações distintas (e.g: dado uma projeção de um conjunto A, selecionando um item desta projeção a ferramenta destacará os elementos mais próximas nesta coleção A e também será possível destacar itens numa coleção B em relação ao mesmo documento da coleção A).

Terminada a implementação, testes serão realizados para que as técnicas sejam comparadas e os resultados analisados em relação as medidas de similaridade desenvolvidas e também, sobre os itens implementados na visualização.

Finalizando este processo, um artigo científico será produzido com o conteúdo teórico dos temas que foram tratados neste documento, aliado a uma descrição da implementação das medidas e os resultados obtidos.

Referências

[ALENCAR 2013]ALENCAR, A. *Visualização da evolução temporal das coleções de artigos científicos*. [S.l.], 2013.

[BASSIL 2012]BASSIL, Y. A survey on information retrivial, text categorization, and web crawling. *Journal of Computer Science and Research (JCSCR)*, v. 1, n. 6, p. 1–11, 2012.

[ELER D.M.; MINGHIM 2007]ELER D.M.; MINGHIM, R. *Projeçao e Arranjo*. [S.l.], 2007.

[FALOUTSOS C.; LIN]FALOUTSOS C.; LIN, K. *FastMap: A Fast Algorithm for Indexing, Data-Mining and Visualization of Tradicional and Mutimedia Datasets*. [S.l.].

[FERNEDA 2003]FERNEDA, E. *Recuperação de Informaçao: Análise sobre a contribuiçao da Ciência da Computação para a Ciência da Informaçao*. [S.l.], 2003.

[MANNING C.D.; RAGHAVAN 2008]MANNING C.D.; RAGHAVAN, P. S. H. *Introduction to Information Retrivial*. [S.l.: s.n.], 2008.

[MINGHIM R.; PAULOVICH 2006]MINGHIM R.; PAULOVICH, F. P. L. A. Content-based text mapping using multi-dimensional projections for exploration of document collections. *Visualization and Data Analysis 2006*, 2006.

[PAULOVICH F 2009]PAULOVICH F, P. N. L. M. R. L. H. Least square projection: a fast high precision multidimensional projection technique and its application to document mapping. *Visualization and Computer Graphics, IEEE Transactions*, v. 14, p. 564–575, 2009.

[PAULOVICH 2008]PAULOVICH, F. V. *Mapeamento de dados multi-dimensionais - integrando mineração e visualização*. [S.l.], 2008.

[PLATT]PLATT, J. *FastMap, MetricMap, and LandMark MDS are all Nyström Algorithms*. [S.l.].

[RAGHAVAN]RAGHAVAN, P. *Information Retrivial Algorithms: A Survey*. [S.l.].

[SALAZAR 2012]SALAZAR, F. *Um estudo sobre o papel de medidas de similaridade em visualização de coleções de documentos*. [S.l.], 2012.

[TEJADA E.; MINGHIM 2003]TEJADA E.; MINGHIM, R. N. L. On improved projection techniques to support visual exploration of multidimensional data sets. *Information Visualization*, Palgrave Macmillan, v. 2, n. 4, p. 218–231, dez. 2003. ISSN 1473-8716. Disponível em: <http://dx.doi.org/10.1057/palgrave.ivs.9500054>.

[VALDIVIA 2007]VALDIVIA, A. *Mapeamento de dados multidimensionais usando árvores filogenéticas: foco em mapeamento de textos*. [S.l.], 2007.

[ZHANG]ZHANG, J. *Visualization for Information Retrieval (The Information Retrieval Series)*. 1. ed. [S.l.: s.n.]. ISBN 3540751475, 9783540751472.