Classificação não-supervisionada hierárquica de artigos jornalísticos

Cirillo Ribeiro Ferreira
Orientador: Prof. Dr. Alair Pereira do Lago

14 de novembro de 2014

Resumo

...

Sumário

1 Introdução

A tarefa de classificar e agrupar documentos textuais remonta desde a antiguidade, iniciando-se na criação da primeira biblioteca do mundo em Nínive, Assíria (atual Iraque), por volta do século VII a.C. Desde então, profissões como bibliotecário, documentalista e arquivista foram criadas para atuar na organização desses documentos que vem sendo produzidos pela humanidade.

Porém com a criação da internet e a popularização de seu uso como ferramenta de comunicação, houve uma explosão de informação que tornou praticamente impossível a classificação desses novos tipos de documentos de maneira manual.

A área de classificação de documentos é bem interessante e possui diversas aplicações práticas como classificação de spam, identificação do idioma utilizado e análise de sentimento, porém, em especial, artigos jornalísticos têm um enorme desafio devido à grande quantidade de novos documentos criados diariamente e a diversidade de temas abordados, especialmente em blogs, mas que carecem de melhor organização.

1.1 Motivação

A motivação desse trabalho vem da insatisfação com a falta de tempo para uma leitura mais crítica e ciente de um contexto maior devido a enorme quantidade de notícias que estamos sujeitos a receber todos os dias e da organização simplória feita pelos grandes portais de notícias.

O motivo de se utilizar algoritmos não-supervisionados vem da necessidade de encontrar relações implícitas entre os artigos e também da diversidade e quantidade de artigos disponíveis para classificação, uma vez que fica quase que impossível criar manualmente um conjunto de treinamento para subsídio de um algoritmo supervisionado.

Ao contrário da classificação supervisionada, a classificação não-supervisionada não necessita de um conjunto de treinamento como entrada, permitindo o seu uso em conjuntos de dados bem variados, algo que é bem comum em artigos jornalísticos.

A ideia não é separar os artigos jornalísticos em grupos muito generalistas comumente usados em jornais ou portais de notícias, como por exemplo as seções: Brasil, Mundo, Economia e Esportes. Mas sim, encontrar grupos mais intrínsecos que representem com mais acurácia a informação passada por esses artigos .....

[Talvez explicar a diferença entre classificação e agrupamento: Livro sistemas inteligentes, pag 318]

Segundo Martin Ester *et al.* [7], existem alguns desafios da área de agrupamento de documentos como: alta dimensionalidade da coleção, onde cada palavra distinta na coleção constitui uma dimensão, e quanto maior a dimensão, maior o desafio para a construção de um algoritmo escalável e eficiente; grande volume de dados; alta precisão e consistência, dependendo do tipo de documentos, alguns algoritmos apresentam resultados muito abaixo do esperado; e descrição mais significativa dos grupos, pois facilita a utilização pelos usuários.

[Agrupamento é uma das mais antigas atividades mentais humanas, usada para tratar uma quantidade enorme de informações que recebemos todos os dias [1].]

1.2 Objetivos

O trabalho de classificação não-supervisionada dos artigos jornalísticos foi dividido em duas partes:

- Criação de uma biblioteca para agrupamento de artigos jornalísticos.
- Proposta e implementação do sistema hVINA (*Hierarchical Viewer of News Articles*) para análise de agrupamento de artigos jornalísticos.

A representação de um agrupamento de documentos é geralmente feita através de um dendrograma, como na figura 1.

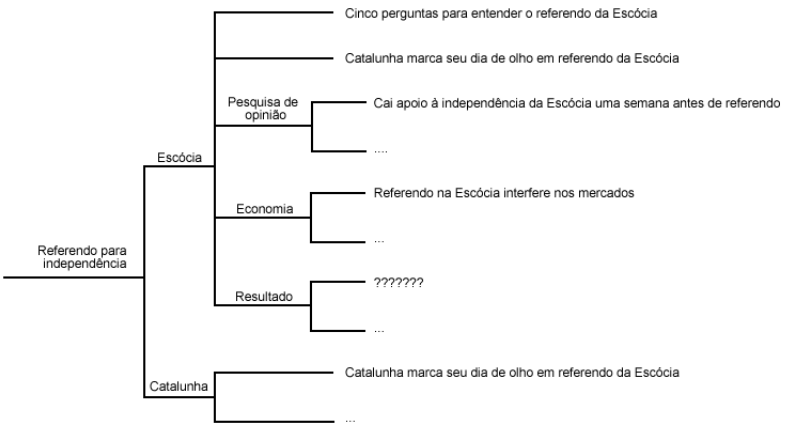

Figura 1: Esquema do resultado esperado

1.3 Organização do trabalho

No capítulo 2 é apresentado os fundamentos da área de reconhecimento de padrão e mais especificamente de análise de agrupamento. No capítulo 3 é discutido em detalhe o algoritmo FIHC, escolhido para ser implementado na biblioteca e sistema propostos no 4o capítulo para o agrupamento de artigos jornalísticos. No capítulo 5 é feito a avaliação do sistema na coleção de artigos do jornalista José de Paiva Netto e também da coleção da Reuters. E por fim, no capítulo 6 são apresentadas as conclusões desse trabalho.

2 Fundamentos

Antes de apresentar e discutir sobre a biblioteca desenvolvida e o sistema proposto, será apresentado alguns conceitos básicos necessários para a compreensão do conteúdo e dos resultados obtidos.

2.1 Reconhecimento de padrão

Segundo Theodoridis e Koutroumbas [1], reconhecimento de padrão é uma área da ciência cujo objetivo é a classificação de objetos dentro de um número de categorias ou classes. Esses objetos de estudo variam de acordo com cada aplicação, podem ser imagens, sinais em forma de ondas (como voz, luz, rádio) ou qualquer tipo de medida que necessite ser classificada. As implicações práticas desses estudos permeiam todas as áreas de atuação humana, principalmente numa sociedade pós-industrial, onde a necessidade de automação das diversas atividades é essencial. Sua aplicação vai desde sistemas que utilizam visão computacional, passando por reconhecimento de caracteres até sistemas para auxílio à domada de decisão.

2.1.1 Formas de processo de aprendizagem

Na área de reconhecimento de padrão os algoritmos utilizam técnicas de aprendizagem computacional para a realização de alguma tarefa e eles são geralmente classificados de acordo a necessidade de intervenção humana durante alguma fase de aprendizagem do algoritmo. São elas: aprendizagem supervisionada, aprendizagem não-supervisionada e aprendizagem semi-supervisionada.

A classificação supervisionada consiste na utilização de um conjunto de dados previamente classificados (ou anotados), que servirá na construção de um modelo para a classificação de novos objetos. Ou seja, algoritmos que utilizam esse conjunto de dados fazem uso de um conhecimento a priori, originado de uma fonte externa. Esse conjunto de dados inicial é denominado de conjunto de treinamento (*training set*). Porém, em muitos casos a criação de um conjunto de treinamento é inviável devido à natureza do problema [pegar exemplos: não se conhece previamente todas as classes]. Para esses tipos, é então empregado a classificação não-supervisionada, que não faz uso de um conjunto de treinamento, pois o próprio algoritmo busca descobrir similaridades intrínsecas nos objetos e agrupa aqueles objetos mais similares. [1, 2, 3]

Por fim, a classificação semi-supervisiona emprega técnicas da classificação supervisionada e não-supervisionada, geralmente com um conjunto pe-

queno de treinamento.

No contexto de reconhecimento de padrão, é comum chamar a classificação supervisionada de apenas classificação e a classificação não-supervisionada de análise de agrupamento, ou simplesmente agrupamento (do inglês *clustering*). A figura ? mostra o esquema das formas de aprendizagem.

[Seria muito bom se colocasse um esquema dessas formas de processo de aprendizagem]

2.2 Análise de agrupamento

Análise de agrupamento é uma classificação de padrão que emprega o processo de aprendizagem não-supervisionada e tem como objetivo o particionamento de objetos em grupos cujo membros sejam similares entre si e diferentes dos membros de outros grupos [2]. Porém, também tem os objetivos secundários de evitar grupos muito pequenos ou muito grandes e encontrar grupos que façam sentido ao usuário. Segundo Theodoridis e Koutroumbas [1], a análise de agrupamento é largamente utilizada na resolução de diversos problemas, e pode ser encontrada em outras áreas por nomenclaturas diferentes, como taxonomia numérica em biologia, e tipologia na área das ciências sociais.

Um grande problema enfrentado na análise de agrupamento é a dimensionalidade ou variáveis dos dados envolvidos, sejam eles observações de estrelas no céu, observações de organismos vivos na natureza ou documentos textuais disponíveis na internet, e uma técnica bastante utiliza para tentar solucionar tal problema é a redução dimensional através da extração de características, que envolve em sintetizar os dados em características mais relevantes para o problema principal sem perda de informação.

2.3 Agrupamento de documentos textuais

Uma aplicação da análise de agrupamento é na organização de documentos. Geralmente ao fazer alguma consulta em sistemas de busca são retornadas centenas de documentos que possuem algum tipo de relevância à consulta, porém esse grande número de resultado pode inviabilizar a utilização desses sistemas, por isso a maioria deles utiliza algumas técnicas de análise de agrupamento para organizar automaticamente os resultados em categorias que fazem sentido ao usuário.

Em especial, quando se fala em documentos textuais, o processo de redução dimensional através de extração de características dos dados é de sensível importância devido à grande redundância dos dados, que neste caso, são sentenças ou palavras. Processar os documentos sem antes usar uma boa técnica

de redução dimensional acarreta quase sempre em baixa eficiência da solução.

2.4 Tipos de algoritmo de agrupamento

Um algoritmo de agrupamento objetiva encontrar as classes naturais das observações, baseados em alguma similaridade. Existem diversos algoritmos para agrupamento desenvolvidos [1, 2], porém neste capítulo serão citadas duas categorias de algoritmos mais empregados nesse processo. Uma explicação com aspectos mais detalhados desses métodos pode ser encontrada em [1, 2].

2.4.1 Agrupamento plano

Agrupamento plano é a categoria de algoritmos cujo os grupos resultantes são dados num único nível, onde nenhum grupo tem relação com outro. O principal algoritmo dessa categoria é o *k-means* [3].

Antes de falar do *k-means* é preciso relembrar um ponto essencial ao entendimento desse algoritmo, que é a definição de centroide. Na geometria, o centroide é o ponto central de uma figura. Dessa forma, dado um grupo, se ligarmos com um segmento de linha as observações mais distantes podemos visualizar uma figura geométrica, e o centroide desse grupo seria a observação mais próxima ao centro dessa figura.

[Inserir algoritmo k-means]

Em termos gerais, o objetivo do *k-means* é dividir n observações em k grupos, minimizando a média das distâncias euclidianas quadráticas entre as observações e os respectivos centroides dos grupos, ou seja, cada observação é associada ao grupo cujo o centroide está mais próximo.

O algoritmo inicia com uma escolha aleatória dos k centroides e a cada iteração é feito um refinamento na escolha desses centroides até que não haja mais reescolha dos centroides ou até que se atinja um número predefinido de iterações. A figura 2 exemplifica a sequência descrita acima.

Há duas propriedades importantes do *k-means*:

- Cada objeto deve pertencer ao menos um dos k grupos.
- Nenhum objeto pertence a mais que um grupo.

Um cuidado especial na utilização desse algoritmo é na escolha do número de grupos desejado. Uma escolha inadequada poderá resultar em um mau agrupamento, unindo duas classes naturais em um único grupo ou dividindo uma classe natural em dois grupos, [falar do fato de não ser deterministico] porém algumas técnicas tem sido desenvolvidas para resolver esse problema

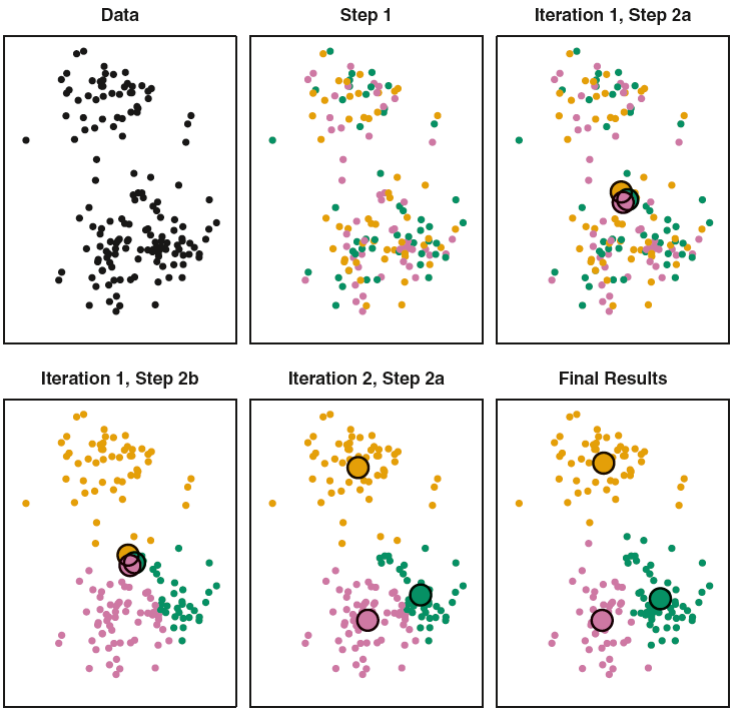

Figura 2: Sequência de passos do algoritmo k-means, retirado de [13]

[2, 3, ?]. (colocar mais referências a artigos para solução desse problema) Um exemplo de problema que pode ocorrer está ilustrado pela figura 3.

Conforme John Wang [9], o algoritmo *k-means* não é adequado para descobrir grupos de tamanhos que variam muito, um cenário comum em agrupamento de documentos. Além disso, é sensível a ruídos que pode ter uma grande influência sobre a escolha do centroide de um grupo, que por sua vez diminui a precisão do agrupamento. Todavia algoritmos como o *k-medoids* [9] foram propostos para resolver o problema do ruído.

2.4.2 Agrupamento hierárquico

Agrupamento plano geralmente é mais simples e fácil de implementar, porém existem certos tipos de problemas em que a visualização de um agrupamento de um único plano não é suficientemente útil, além disso, como visto no tópico anterior, esse tipo de agrupamento tem dois problemas que são a escolha do número de grupo na entrada e o fato de não serem determinísticos [3]. Dessa forma, algoritmos que criassem agrupamentos com vários níveis e que tivessem como requisito opcional a entrada do número de grupos foram

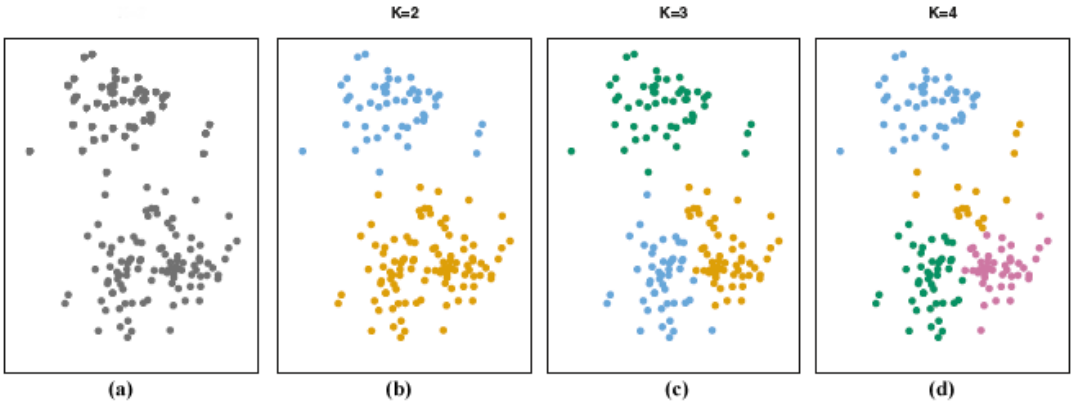

Figura 3: Em (a) é mostrado um conjunto de dados fictício com três classes naturais. Em (b) quanto é escolhido o número de grupos k = 2, as classes naturais 2 e 3 são unidas num único grupo. Em (c), com k = 3, os grupos refletem as classes naturais dos dados, e em (d) com k = 4 a classe natural 3 é dividida entre dois grupos.

desenvolvidos. Esses algoritmos criam uma árvore hierárquica de grupos, uma estrutura que gera mais informação, uma vez que as relações implícitas entre os grupos ficam mais evidentes.

Existem duas abordagens no funcionamento de um algoritmo hierárquico que são:

- Abordagem aglomerativa:. Gera a árvore de grupo ao continuamente calcular a similaridade de todos os pares de grupos e unificar o par mais similar (segundo um critério de similaridade), criando a árvore das folhas até a raiz, por isso é também conhecida como abordagem bottom-up. Um exemplo dessa abordagem é ilustrado na figura 4.

- Abordagem divisiva: Gera a árvore de grupo da raiz às folhas (top-down), utilizando em cada nível da árvore um algoritmo de agrupamento plano para a divisão dos grupos em nós filhos.

2.5 Critério de avaliação

Existem 2 tipos de erros que podem ocorrer no processo de agrupamento. O primeiro, chamado de falso positivo (FP) ou erro tipo I, ocorre quando o algoritmo associa dois documentos não similares a um mesmo grupo, e o segundo, chamado de falso negativo (FN) ou erro tipo II, ocorre quando o

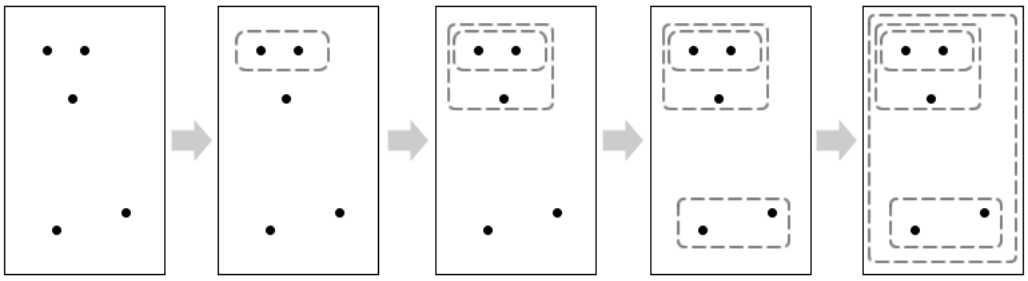

Figura 4: Exemplo de um algoritmo hierárquico aglomerativo, retirado de [8], onde o critério de similaridade utilizado é a distância euclidiana.

algoritmo associa dois documentos similares a grupos distintos. Um critério de avaliação justo tem de levar em conta as ocorrências desses dois erros. A tabela ? a seguir, mostra a matriz de contingência entre os documentos e grupos.

[Inserir tabela de contigencia]

Duas medidas importantes na área de reconhecimento de padrão e também na estatística, são a precisão e cobertura do resultado observado. O primeiro, no contexto de reconhecimento de padrão, mede a fração de documentos que estão classificados corretamente no grupo e é dado pela seguinte fórmula:

$$P = \frac{TP}{TP + FP} \tag{1}$$

Já a cobertura mede a fração de documentos que foram classificados corretamente dentre os documentos similares e é dado pela fórmula:

$$R = \frac{TP}{TP + FN} \tag{2}$$

Por exemplo, dado o conjunto de documentos {1, 2, 3, 4, 5, 6}, onde N_1 = {1, 3, 5} fazem parte de uma classe natural, ou seja, são similares; e N_2 = {2, 4, 6} fazem parte de outra classe natural. Dado um grupo C = {1, 3, 4, 5} a precisão de C em relação a N_1 é P = 3/4 e a cobertura é R = 3/3 = 1, ou seja, nem todos os documentos de C são similares entre si, apesar de que todos os documentos de N_1 foram agrupados corretamente em C.

Porém, a utilização de duas medidas para avaliação e comparação de algoritmos torna difícil determinar se um algoritmo é superior a outro ou não. Pois é melhor um algoritmo com alta precisão e baixa cobertura, ou vice-versa? A resposta para essa pergunta depende do contexto onde é aplicado o

agrupamento. Segundo Manning et al, na maioria das vezes é mais aceitável o erro de tipo I, pois separar documentos similares em grupos distintos é geralmente pior do que juntar documentos não similares num mesmo grupo [7].

Dado isto, um critério para avaliação bastante utilizado é a medida F (*F-measure*), pois avalia o agrupamento utilizando a média harmônica ponderada da precisão e da cobertura. A medida F é dada por:

$$F_\beta = (1 + \beta^2) \times \frac{P \times R}{\beta^2 P + R} \tag{3}$$

onde $\beta^2 \in [0, \infty]$.

Um valor para β bastante utilizado é $\beta = 1$, pois tanto a precisão quanto a cobertura terão o mesmo peso na medida. Neste caso, a medida seria simplificada da seguinte forma:

$$F_1 = 2 \times \frac{P \times R}{P + R} \tag{4}$$

Já definindo $\beta > 1$, a precisão será mais enfatizada em relação à cobertura. Note que para a medida F, o algoritmo não basta ter apenas boa precisão ou apenas boa cobertura, mas sim ambas características, o que a torna um dos critérios mais utilizados para a análise. Porém, ela não é o único critério de avaliação existente. Dentro da área de estudo sobre avaliação de agrupamentos existem critérios internos e externos e a medida F se enquadra na segunda categoria, pois avalia o agrupamento a partir de uma classificação de referência [7].

3 FIHC

Frequent Itemset-based Hierarchical Clustering (FIHC) é um algoritmo para agrupamento hierárquico de documentos proposto por Martin Ester *et al.* [7] e que utiliza o conceito de conjuntos de itens frequentes (do inglês *frequent itemset*).

Segundo Martin Ester *et al.*, a ideia utilizada para o critério de agrupamento dos documentos é que existe algum conjunto de itens frequentes para cada grupo (tópico) no conjunto de documentos, e diferentes grupos compartilham poucos conjuntos de itens frequentes.

Segundo os autores, o algoritmo tenta resolver alguns desafios da área como: alta dimensionalidade da coleção, onde cada palavra distinta na coleção constitui uma dimensão, e quanto maior a dimensão, maior o desafio para a construção de um algoritmo escalável e eficiente; grande volume de dados; alta precisão e consistência, dependendo do tipo de documentos, alguns algoritmos apresentam resultados muito abaixo do esperado; e descrição mais significativa dos grupos, pois facilita a utilização pelos usuários.

Um ponto de interesse é que, diferentemente dos algoritmos mais comuns para agrupamento hierárquico, no FIHC a parametrização do número de grupos desejado é opcional, assim o usuário não precisa ter nenhum conhecimento prévio da coleção de documentos.

O algoritmo utiliza a abordagem aglomerativa, descrita na seção 2.4.2, porém ao invés de construir a árvore baseando-se na similaridade entre as observações (documentos) ele a faz baseando-se nos grupos [7]. Possui dois passos principais, que é a construção dos grupos e posteriormente a criação da árvore hierárquica, entretanto antes de apresentá-los é necessário introduzir algumas definições.

3.1 O que é conjunto de itens frequentes

Por definição, conjunto de itens é o conjunto de palavras que ocorrem em conjunto na coleção de documentos. Logo conjunto de itens frequentes é o conjunto de palavras que ocorrem numa quantidade de documentos acima de um limiar de suporte definido.

Em outras palavras, assumindo que l é o limiar de suporte. Se F é um conjunto de palavras, o apoio (*support* em inglês) de F é a proporção de documentos no qual F é um subconjunto das palavras desses documentos. Assim, dizemos que F é um conjunto de itens frequentes se o seu valor de apoio é l ou superior. A tabela 1 ilustra um exemplo:

No exemplo acima, a palavra "Grécia" aparece nos documentos, (2), (3) e (4), assim seu suporte é 3; a palavra "crise" aparece em (3) e (5), logo

Documento	Palavras (itens)
doc.1	{Bolsa, europeia, opera, em, queda, após, anúncio, de, pacote, grego}
doc.2	{Japão, e, Grécia, ficam, no, 0, a, 0, no terceiro, dia, de, competição}
doc.3	{Descoberta, de, tumba, misteriosa, anima, Grécia, em, meio, à, crise, europeia}
doc.4	{Grécia, assume, presidência, da, União, Europeia, por, seis, meses}
doc.5	{Crise, está, prejudicando, saúde, mental, dos, gregos}

Tabela 1: Coleção de artigos

seu suporte é 2; a palavra "de" ocorre em todos os documentos, exceto em (4) e (5), assim seu suporte é 3; a palavra "europeia" ocorre em (1), (3) e (4), logo seu suporte é 3. Todas as outras palavras ocorrem apenas em um único documento. Logo se for definido o limiar de suporte $l = 2$, teremos como conjuntos de itens frequentes os conjuntos {Grécia}, {crise}, {de} e {europeia}. Além disso, as duplas {Grécia, europeia}, {Grécia, de} e {europeia, de} ocorrem em 2 documentos, logo também são conjuntos de itens frequentes.

No caso do algoritmo FIHC, conjuntos de itens frequentes são também denominados de conjuntos globais de itens frequentes, e suporte é denominado de *global support*.

3.1.1 Algoritmo Apriori

A extração dos conjuntos de itens frequentes é uma das técnicas mais utilizadas para caracterização de dados com objetivo de resumir os atributos gerais mais relevantes de um conjunto de dados. O exemplo mostrado na seção anterior é uma forma de encontrar os conjuntos de itens frequentes, porém esse método não é eficiente quando é utilizado em uma coleção grande de documentos. Um dos algoritmos mais populares para tal propósito, seja em um banco de dados ou em documentos textuais, é o algoritmo Apriori. (wikipedia?)

A algoritmo utiliza a propriedade de anti-monotonicidade [10], onde se um conjunto de itens não é frequente, então todos os seus superconjuntos também não serão.

Ele inicia a busca a partir de uma coleção de conjunto de itens frequentes com cardinalidade k = 1, chamado de F_1 . A partir daí todos os conjuntos de cardinalidades maiores são obtidos a partir de uma coleção de candidatos

C. O algoritmo para quando não encontrar mais nenhum conjunto de itens frequentes. A figura ? mostra o algoritmo:

Algorithm 1 Algoritmo Apriori para extração de conjuntos de itens frequentes em documentos textuais

$F_1 \leftarrow$ (Conjunto de itens frequentes de cardinalidade 1)
$k \leftarrow 1$
while $F_k \neq \emptyset$ **do**
    $C_{k+1} \leftarrow candidate\_gen(F_k)$
    for all documento d na coleção **do**
        $C'_d \leftarrow subset(C_{k+1}, d)$
        for all candidato c $\in C'_d$ **do**
            $c.count \leftarrow c.count + 1$
    $F_{k+1} \leftarrow \{c \in C_{k+1} | c.count \geq$ suporte mínimo$\}$
    $k \leftarrow k + 1$
    return $\cup_k F_k$

3.2 O que é *cluster frequent item* e *cluster support*

Dado o grupo C_i, um elemento de um conjunto global de itens frequentes é denominado de *cluster frequent item* de C_i se este elemento está contido em um número mínimo de documentos de C_i, definido como *minimum cluster support*.

Além disso, *cluster support* de um item em C_i é definido como a porcentagem de documentos de C_i que contém o item.

3.3 O que é tf-idf

Tf-idf é uma medida estatística usada para avaliar o quão importante uma palavra é para um documento em relação a uma coleção de documentos. (wikipedia) A ideia é que se uma palavra é tão comum em diferentes documentos da coleção, então provavelmente ela tem pouca relevância, por exemplo, a conjunção aditiva "e". A importância aumenta proporcionalmente ao número de vezes que uma palavra aparece no documento, mas é compensado pela frequência da palavra no corpus. O tf-idf é dado pelo produto de duas medidas: a frequência do termo e a frequência inversa do documento, dado por:

$$tf - idf(t, d) = tf(t, d) \times \log \frac{N}{df(t)} \tag{5}$$

onde $tf(t,d)$ é a razão entre o número de vezes que o termo t aparece no documento d e a quantidade de termos em d; N é a quantidade de documentos na coleção e $df(t)$ é a quantidade de documentos da coleção que possui o termo t. Existem diversas variantes dessa medida como visto em [3].

3.4 Construção dos grupos

Apresentado alguns conceitos relacionados ao FIHC, vamos ao algoritmo propriamente dito. Como já dito, o primeiro passo do algoritmo é construir os grupos, e isso é feito em duas etapas. Primeiramente, os grupos iniciais são construídos a partir dos conjuntos globais de itens frequentes extraídos da coleção e então é feito o processo de disjunção desses grupos para que no final um documento pertença somente a um único grupo.

Na criação dos grupos iniciais, para cada conjunto global de itens frequentes é criado um grupo que inicialmente incluirá todos os documentos que contêm o conjunto de itens em seu corpo. Logo cada grupo possuirá um conjunto de itens frequentes gerador.

Note que esses grupos podem não ser disjuntos, pois um documento pode conter vários conjuntos de itens frequentes.

[Inserir tabela 1]

[Inserir tabela 2]

3.4.1 Disjunção dos grupos sobrepostos

A disjunção é um processo importante, pois garante que ao final do algoritmo cada documento pertença somente ao grupo que melhor o descreva. Para cada documento D, é listado todos os grupos no qual ele pertence, e após isso é calculado para cada um desses grupo C uma medida que indicará a "relevância" de D em relação a C. Essa medida tem o nome de Score e é proposto em [7].

Após os cálculos da medida Score de D para cada grupo C, é mantido o documento somente no grupo que maximizou essa medida. Esse cálculo de relevância de um grupo inicial C_i para um documento D_j é definido por:

$$\begin{aligned} Score(C_i \leftarrow D_j) = &|\sum_{x} tf - idf(x, D_j) * cluster\_support(x)| \\ &-|\sum_{x'} tf - idf(x', D_j) * global\_support(x')| \end{aligned} \tag{6}$$

onde x representa um *global frequent item* que está contido em D_j e também é um *cluster frequent item* de C_i; x' representa um *global frequent item* que

está contido em D_j, mas que não é um *cluster frequent item* de C_i; $tf - idf(x, D_j)$ e $tf - idf(x', D_j)$ são as medidas tf-idf dos itens x e x' em D_j; $cluster\_support(x)$ é o *cluster support* de x e $global\_support(x')$ é o *global support* de x'.

[Inserir tabela 3]

O primeiro termo da função $Score(C_i \leftarrow D_j)$ recompensa C_i se o *global frequent item* x em D_j é também um *cluster frequent item* de C_i e o segundo termo penaliza C_i se o *global frequent item* x' em D_j não é um *cluster frequent item* de C_i. De acordo com Martin Ester *et al.* [7], intuitivamente, um grupo C_i é "bom" para um documento D_j se há muitos *global frequent items* em D_j que aparecem em "muitos" documentos em C_i. Se há mais de um grupo que maximiza a medida $Score(C_i \leftarrow D_j)$, então é escolhido o grupo com maior número de itens em seu conjunto de itens frequentes gerador.

Seguindo o exemplo da tabela ?, note que o documento artigo.3 pertence aos grupos C(mercado), C(crise), C(Grécia) e C(mercado, crise). Para se decidir em qual desses grupos o artigo deve permanecer, são calculados os seguintes *Score*:

$Score(C(mercado) \leftarrow artigo.3) = 1 * 1 + 4 * 0,5 + 3 * 0,75 = 5,25$

$Score(C(crise) \leftarrow artigo.3) = 1 * 0,5 + 4 * 1 - 3 * 0,3 = 3,6$

$Score(C(Grcia) \leftarrow artigo.3) = 1 * 1 + 3 * 1 - 4 * 0,4 = 2,4$

$Score(C(mercado, Grcia) \leftarrow artigo.3) = 1 * 1 + 3 * 1 - 4 * 0,4 = 2,4$

Logo, o documento artigo.3 seria mantido apenas no grupo C(mercado).

[Inserir tabela 4]

3.5 Criação da árvore de grupos

Uma vez computado os grupos iniciais, podemos visualizar cada um deles como um tópico ou subtópico na coleção de documentos. Agora é necessário criar a árvore, colocando os tópicos mais generalistas no topo e os mais específicos abaixo. Essa árvore tem como objetivos cria uma estrutura para a navegação entre os documentos e também facilitar a poda dos grupos muito específicos.

Os grupos são inicialmente aninhados na árvore de acordo com a quantidade de itens em seu conjunto global gerador. Dessa forma, aqueles que possuem somente um item, estão no primeiro nível da árvore, e assim por diante. A raiz (nível 0) da árvore é por definição um grupo vazio e sem nenhum item em seu conjunto global de itens frequentes. Como dito no início desse capítulo, o FIHC utiliza a abordagem aglomerativa para a criação da hierarquia, ou seja, para cada grupo C no nível k é escolhido um pai dentre os potenciais grupos do nível *k-1*, cujo o conjunto global seja subconjunto dos conjuntos globais de itens frequentes de C. Se houver mais de um potencial

pai no nível *k-1*, então é escolhido aquele grupo que maximiza uma medida de ???, similar ao processo descrito na seção 3.4.1. Isso é melhor detalhado em [7].

Uma vez escolhido um grupo pai para cada grupo da coleção, é necessário verificar se é possível unificar os grupos similares ou com tópicos muito específicos, garantindo uma hierarquia mais natural para a navegação e melhorando a acurácia do algoritmo. Isso é feito de duas formas: 1) através do processo de poda dos grupos filhos e 2) unificação dos grupos similares do nível 1.

A medida de similaridade utilizada no algoritmo FIHC utiliza a ideia de tratar um grupo como um documento conceitual, definindo a partir da combinação de todos os documentos do grupo, e a partir daí é tirado a medida através de um cálculo bem similar à medida *Score* apresentada na seção anterior. Dado um grupo C_a e C_b, a medida de similaridade entre os grupos é calculado a partir da média geométrica entre a similaridade de C_a para C_b e da similaridade de C_b para C_a. Da seguinte forma:

$$Inter\_Sim(C_a \leftrightarrow C_b) = [Sim(C_a \leftarrow C_b) * Sim(C_b \leftarrow C_a)]^{\frac{1}{2}} \tag{7}$$

onde a similaridade Sim de C_a para C_b é dado como:

$$Sim(C_a \leftarrow C_b) = \frac{Score(C_a \leftarrow doc(C_b))}{\sum_x tf - idf(x, doc(C_b)) + \sum_{x'} tf - idf(x', doc(C_b))} + 1 \tag{8}$$

Em $Sim(C_i \leftarrow C_j)$, a medida *Score* é a mesma discutida na seção anterior, porém a única diferença é que o documento $doc(C_j)$ utilizado no cálculo é um documento conceitual, construído a partir da combinação de todos os documentos da subárvore de C_j, e $tf - idf(x, doc(C_b))$ e $tfidf(x', doc(C_b))$ são, respectivamente, as medidas tf-idf dos itens x e x' em $doc(C_j)$.

3.6 Rotulagem do grupo

Uma etapa importante no processo de agrupamento é a de rotulagem dos grupos, pois para um usuário entenda o resultado do algoritmo e o utilize sem dificuldades é necessário que a rotulagem seja legível e faça sentido a ele.

No caso do FIHC, a própria abordagem de criação dos grupos a partir dos conjuntos globais de itens frequentes da coleção e posterior escolha dos *cluster frequent items* torna natural pensar que esses conjuntos de palavras também são bons candidatos para dar nome aos próprios grupos.

[Exemplo]

Referências

[1] Y. Ma, H. Zhang. *Contrast-based image attention analysis by using fuzzy growing*. Proceedings of the Eleventh ACM international Conference on Multimedia Berkeley, CA, USA. 2003.

[2] L. Itti, C. Koch. *A saliency-based search mechanism for overt and covert shifts of visual attention* Vision Research, Volume 40, Issues 10-12, Pages 1489-1506, ISSN 0042-6989, DOI: 10.1016/S0042-6989(99)00163-7. 2000.

[3] H. Liu, S. Jiang, Q. Huang, C. Xu, W. Gao *Region-based visual attention analysis with its application in image browsing on small displays*. In Proceedings of the 15th international Conference on Multimedia (Augsburg, Germany, September 25 - 29, 2007). 2007.

[4] S. Frintrop, E. Rome, H. I. Christensen. *Computational visual attention systems and their cognitive foundations: A survey*. ACM Trans. Appl. Percept. 7, 1 (Jan. 2010), 1-39. 2010.

[5] T. M. Breuel. *High performance document layout analysis*. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.13.1303 2003

[6] E. Gamma, R. Helm, R. Johnson, J. Vlissides. *Design Patterns: Elements of Reusable Objected-Oriented Software*. Addison-Weskey Professional; 1 edition, November 10th, 1994

[7] E.R. Dougherty, R.A. Lotufo. *Hands-on Morphological Image Processing*. SPIE press, 2003

[8] *Tesseract OCR* http://code.google.com/p/tesseract-ocr/

[9] R. O. Duda, P. E. Hart, D. Stork. *Pattern Classification*. Wiley-Interscience; 2nd edition, October 2000.