Contextual Document Clustering

Vladimir Dobrynin

St.Petersburg State University and SOPHIA Search Ltd.

25 December 2009

Outline

1. Contextual Document Clustering
2. Discussion
3. US patents abstracts. IPC G06: computing; calculating; counting
4. References

Context

Compare contexts of these words and phrases:

- **computer**
  It is hard to predict what it is about – text about computing or about health, art, sport, etc.
- **motherboard**
  With high probability this text should be about computer hardware.

Simple Ideas

- All authors (and texts by these authors) can be grouped into **communities** in such a way that members of one community share the same background in terms of education, profession, experience, interests an so on. They use the same language to express the same ideas.
- Members of any specific community usually use few but **very specific terms** that aren't used in any of other communities. These terms can be considered as **markers** that distinguish texts written by members of this community from texts written by members of other communities.

Notations

- $\mathcal{X}$ – set of all documents in the document corpus
- $\mathcal{Y}$ – set of all words occurring in documents from $\mathcal{X}$
- $tf(x, y)$ – term frequency – the number of occurrence of the word $y \in \mathcal{Y}$ in the document $x \in \mathcal{X}$
- $p(Y|z)$ – context of word z
  A probability distribution of a set of words which co-occur with a given word z in a document from $\mathcal{X}$.

Context of word z:

$$p(y|z) = \frac{\sum_{x \in D_z} tf(x, y)}{\sum_{x \in D_z, y'} tf(x, y')},$$

where D_z is the set of all documents that contain the word z.

Narrow context

Selection of words with narrow contexts is based on consideration of

1. the entropy

$$H(Y|z) = -\sum_{y} p(y|z) \log p(y|z)$$

2. and document frequency

$$df(z) = |\{x \ : \ tf(x,z) > 0\}|.$$

Main idea – given collection $\mathcal{X}$, extract set $\mathcal{Z}$ of words with narrow contexts and use contexts of these words $p(Y|z), \ \ z \in \mathcal{Z}$ as cluster attractors.

Entropy and Context Narrowness

Let

$$p(y|z) = \frac{1}{|T(D_z)|}, \quad y \in T(D_z)$$

where $T(D_z)$ is the set of words belonging to documents where z occurs. Then

$$H(p(Y|z)) = \log|T(D_z)|.$$

For any non-uniform distribution

$$H(p(Y|z)) < \log|T(D_z)|.$$

Small $H(Y|z)$ means that context of the word z can be described by a relatively small set of words (concepts)

Entropy and Document Frequency

Heaps law:

$$|\mathcal{Y}| = O(n^{\beta})$$

where constant $\beta < 1$ and n is size of collection in words. Let $n(z)$ – size of D_z in words and l_{avg} – average document size in $\mathcal{X}$. Then

$$n(z) \approx l_{avg} \cdot df(z)$$

and

$$|T(D_z)| = O(n(z)^{\beta}) = O((l_{avg} \cdot df(z))^{\beta}) = \\ = O(df(z)^{\beta}).$$

Hence

$$H(Y|z) = O(\log|T(D_z)|) = c + O(\log df(z)^{\beta}) = \\ c + O(\beta \log df(z)) = c + O(\log df(z)).$$

Contextual Document Clustering -1-

Input

- Term frequencies $tf(x, y), \;\; x \in \mathcal{X}, \; y \in \mathcal{Y}$
- N – number of clusters
- Parameters $df_{min}, r, \alpha > 1$

Output

- "Hard" clustering of all documents into N clusters

Contextual Document Clustering -2-

Context calculation

$$p(y|z) = \frac{\sum_{x \in D_z} tf(x,y)}{\sum_{x \in D_z, y'} tf(x,y')}, \quad z \in \mathcal{Y}$$

Set $\mathcal{Z}$ of words with narrow contexts selection

For every $i = 1, \ldots, r$

$$\mathcal{Y}_i = \{z : df_i \leq df(z) < df_{i+1}\},$$

$$df_1 = df_{min}, \; df_{i+1} = \alpha \cdot df_i,$$

$$\mathcal{Z}_i \subseteq \mathcal{Y}_i, \; |\mathcal{Z}_i| = \frac{N \cdot |\mathcal{Y}_i|}{\sum_{j=\overline{1,r}} |\mathcal{Y}_j|},$$

$$z_1 \in \mathcal{Z}_i, \; z_2 \in \mathcal{Y}_i - \mathcal{Z}_i \;\rightarrow\; H(Y|z_1) \leq H(Y|z_2).$$

$$\mathcal{Z} = |\cup_i \mathcal{Z}_i|.$$

Contextual Document Clustering -3-

Document clustering

For every document $x \in \mathcal{X}$ calculate word probability distribution

$$p(y|x) = \frac{tf(x,y)}{\sum_{y'} tf(x,y')}, \; y \in \mathcal{Y}$$

and document x will be assigned to the cluster with centroid $p(Y|z)$ if

$$z = \text{argmin}_{z'} JS_{0.5,0.5}[p(Y|z'), p(Y|x)]$$

where $JS_{0.5,0.5}[p,q] = H(\frac{p+q}{2}) - 0.5H(p) - 0.5H(q)$ is Jensen-Shannon divergence of probability distributions p and q.

Why we use *r* and α parameters?

- If $z \in \mathcal{Z}_i, \quad i = 1, \ldots, r$ then

$$H(Y|z) = O(\log |T(D_z)|) = c + O(\log df(z))$$

$$= c + O(\log(df_{min}\alpha^i)) = c + O(i)$$

- Given a theme (Math) we would like to disclose all *narrow* topics of the theme (equations, algebra,...) presented by significant number of documents in $\mathcal{X}$ and accumulate all other Math documents into a more *broad* topic.

Complexity

CDC complexity is

$$O(K \cdot |S|),$$

where K is the number of clusters and S is the set of non-zero elements in document-term matrix. By the way, complexity of K-means algorithm is $O(t \cdot K \cdot |S|)$ where t is the number of iterations.

Clustering principle of CDC

Split document corpus into relatively large groups of documents that are covered by relatively small number of concepts.

Comparing with K-means. Notations

- Document collection $D, |D| = N$
- Every document is represented by
  - Index $n = 1, \dots, N$
  - Feature vector $\mathbf{x}^{(n)} \in \Re^M$ where M is a number of features
- Every cluster is represented by
  - Index $k = 1, \dots, K$
  - centroid $\mathbf{m}^{(k)} \in \Re^M$ – average of feature vectors representing all documents from the cluster
- $d(\mathbf{x}, \mathbf{y})$ is a distance measure between two vectors $\mathbf{x}, \mathbf{y} \in \Re^M$

Comparing with K-means. "Soft" K-means

1. Initialization

- Set centroids $\{\mathbf{m}^{(k)}\}_{k=1,\dots,K}$ to random values.
- Set a value for parameter β.

2. Assignment step

Responsibility $r_k^{(n)}$ determines degree to which $\mathbf{x}^{(n)}$ is assigned to cluster k:

$$r_k^{(n)} = \frac{\exp(-\beta d(\mathbf{m}^{(k)}, \mathbf{x}^{(n)}))}{\sum_{k'} \exp(-\beta d(\mathbf{m}^{(k')}, \mathbf{x}^{(n)}))}$$

3. Update step

$$\mathbf{m}^{(k)} = \frac{\sum_n r_k^{(n)} \mathbf{x}^{(n)}}{R^{(k)}},$$

where

$$R^{(k)} = \sum_n r_k^{(n)}.$$

Comparing with K-means. Mixture of two Gaussians -1-

Points from $\{x_n\}_{n=1}^{N}, x_n \in \Re$ are distributed according to mixture of two Gaussians:

$$P(x|\mu_1, \mu_2, \sigma) = \sum_{k=1}^{2} p_k \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu_k)^2}{2\sigma^2}\right),$$

where $p_k = 0.5$.

Comparing with K-means. Mixture of two Gaussians -2-

Given standard deviation σ we can find means μ_1 and μ_2 using iterations:

$$\mu_k = \frac{\sum_{n=1}^{N} \rho_k^{(n)} x_n}{\sum_{n=1}^{N} \rho_k^{(n)}}, \quad k = 1, 2,$$

$$\rho_1^{(n)} = \frac{1}{1 + \exp[-(\omega_1 x_n + \omega_2)]},$$

$$\rho_2^{(n)} = \frac{1}{1 + \exp[+(\omega_1 x_n + \omega_2)]},$$

$$\omega_1 = \frac{\mu_1 - \mu_2}{\sigma^2}, \; \omega_2 = \frac{\mu_2^2 - \mu_1^2}{2\sigma^2}.$$

If in "soft" K-means we have $d(x, y) = \frac{1}{2}(x - y)^2$ and $\beta = \frac{1}{\sigma^2}$ then responsibility $r_k^{(n)} = \rho_k^{(n)}$ and hence in this case "soft"

Problems with K-means

This analysis shows that using k-means clustering we implicitly suppose (at least in case of special type of distance function) that:

1. points in a cluster are generated by a Gaussian generator;
2. size (number of points) of every cluster is the same ($p_k = \frac{1}{K}$, where K is the number of clusters);
3. every cluster is spherical in shape and has the same diameter (same σ for all generators).
4. initialization? number of iterations (complexity)?

Clustering principle of K-means

Depends of choice of distance measure $d(x, y)$.

Comparing with Information Bottleneck. Notations

- X – discrete random variable with values from $\mathcal{X}$ – documents
- $p(x)$ – probability distribution of X –is proportional to size of document x
- T – discrete random variable with values from $\mathcal{T}$ – clusters
- $p(t|x)$ – "soft" clustering documents from $\mathcal{X}$ into clusters $\mathcal{T}$
- $\mathcal{Y}$ – set of documents features – words
- Y – random variable with values in set $\mathcal{Y}$
- $p(x, y),\ x \in \mathcal{X}, y \in \mathcal{Y}$ – known joint probability distribution

$$p(x, y) = \frac{tf(x, y)}{\sum_{x', y'} tf(x', y')},$$

where $tf(x, y)$ – a number of times term t occurs in document x.

Comparing with Information Bottleneck. Optimization Criterion

Minimize

$$\mathcal{L}[p(t|x)] \equiv I(X;T) - \beta I(T;Y)$$

where Lagrange multiplier $\beta > 0$.

- Minimization of mutual information $I(X;T)$ – maximization of **degree of compression** of information presented in document corpus.
- Maximization of mutual information $I(T;Y)$ – all documents from the same cluster have the same features – in other words they are **similar to each other**.
- If $\beta = 0$ then all documents are assigned to one cluster.
- If $\beta \to \infty$ then we have one cluster per document.

Comparing with Information Bottleneck. Optimal Clustering

Theorem

Probability distribution $p(t|x)$ that minimizes functional $\mathcal{L}[p(t|x)]$ can be presented in the following form

$$p(t|x) = \frac{p(t)}{Z(x,\beta)} e^{-\beta D_{KL}[p(y|x)||p(y|t)]}, \quad t \in \mathcal{T}, y \in \mathcal{Y}.$$

Here $Z(x,\beta)$ is a normalizer.

Problems with Information Bottleneck

- How to select reasonable value for β? This parameter should reflect user's idea about "good" clustering
- Number of clusters ($|\mathcal{T}|$) should be less than 200 (Ron Bekkerman. Private communication)

Clustering principle of Information Bottleneck

Depends on choice of parameter β value.

Database statistics

Number of documents	553,792
Number of words	57,096
Number of clusters	1,074
Max cluster size	4,170

Top 10 clusters

Context word	Cluster size	Stem	Stem document frequency	Stem rank
sdram	4170	sdram	368	7
gradation	3827	gradat	584	7
pll	3520	pll	298	7
epg	3246	epg	409	7
reproducing	2953	reproduc	4272	6
inode	2922	inod	120	7
uninstall	2910	uninstal	119	7
spooler	2696	spooler	144	7
mining	2694	mine	1434	6
metrology	2653	metrolog	319	7

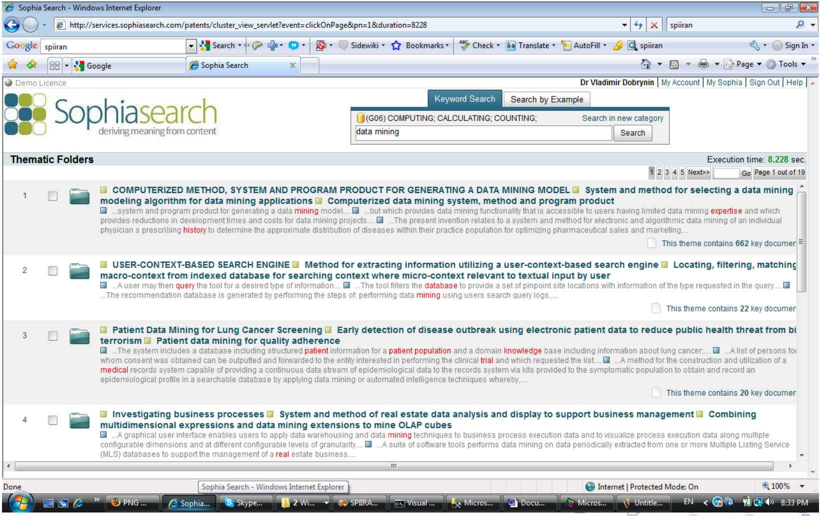
Top 10 words from context "mining"

Word	Stem	Probability
data	data	0.0616228538861579
mining	mine	0.0319750184037424
system	system	0.02626392154069
method	method	0.0210990002612143
information	inform	0.0132506945928617
model	model	0.0128944931253117
database	databas	0.0122652038659733
user	user	0.0108997649070314
includes	includ	0.0100805015316663
processing	process	0.00867944242596944

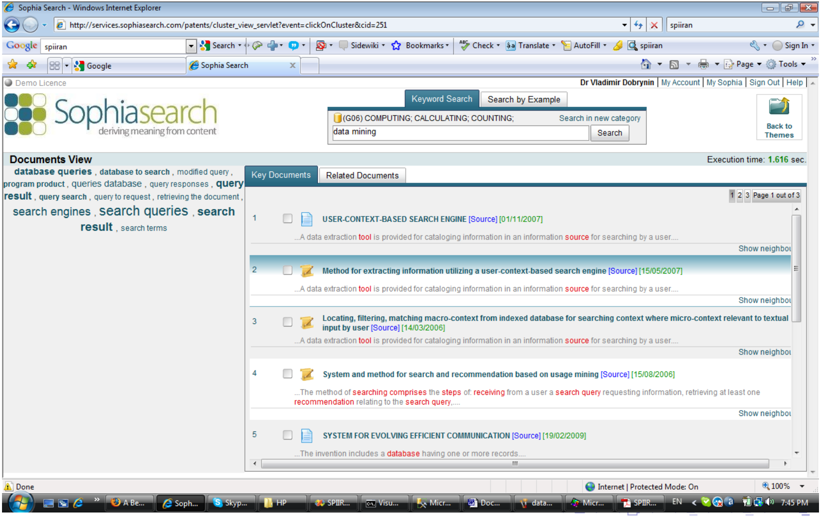
Top 5 patents nearest to the context "mining"

Company	Date	Title
Oracle	2005-03-08	Data mining application programming interface
Oracle	2007-02-06	In-database clustering
Lucent Technologies	2002-05-07	System and method for analyzing and displaying telecommunications switch report output
IBM	2009-04-21	Computerized data mining system, method and program product
Oracle	2006-10-03	Enterprise web mining system and method

Clusters for query "data mining"



Patents from cluster 2



-1-

- Baeza-Yates and Ribeiro-Neto. *Morden Information Retrieval*. ACM Press, 1999
- Daniel Chandler. *Semiotics for Beginners.* http://www.aber.ac.uk/media/Document/S4B
- Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory. Second Edition.* John Wiley & Sons, Inc., Hoboken, New Jersey, 2006
- Vladimir Dobrynin, David Patterson and Niall Rooney. Contextual Document Clustering. *Lecture Notes in Computer Science. Advances in Information Retrieval*, 2997,167–180, 2004
- David J.C. MacKay. *Information Theory, Inference, and Learning Algorithms.* Cambridge University Press, 2003

-2-

- N. Tishby, F. Pereira and W. Bialek. The Information Bottleneck Method. *Proc. 37th Allerton Conference on Communication and Computation*, 1999
- Noam Slonim. *The Information Bottleneck: Theory and Applications*. PhD thesis, Hebrew University, Israel, 2002