Giving an Upperbound of the Number of Clusters and Relevant Words in Hierarchical Document Clustering Based on BIC

Yoshio FUKUSHIGE
fukushige.yoshio@jp.panasonic.com
Network Development Center, Matsushita Electric Industrial Co., Ltd.

Abstract

This working note quickly describes a new generative model based approach to automatic document clustering, using the BIC as the model selection criterion. A new method based on a graphical model is proposed to give an upperbound to the numbers of clusters and relevant words. The result of an experiment using the NTCIR web data collection is briefly reported.

Keywords: *generative model, BIC*

1. Introduction

1.1. Overview

When clustering documents, the number of the variables (dimension) is usually very large. In machine learning, high dimensionality is known to lead to high computational cost and low precision. Hence various approaches have been proposed to reduce the number of words used in the clustering; some are based on word frequency, some employ information gain, and some perform data transformation (e.g. LSI [2]). However those approaches lack the direct connection to the nature of clustering task.

And in the clustering problem, the number of the classes has to be estimated. Conventional clustering approaches solve this issue by fixing either the maximum number of the child clusters or minimum number of the elements in each cluster, without theoretical support.

Therefore the following issues are addressed in this work through a probabilistic approach:

- how to determine which words are relevant in clustering
- how to determine the maximum number of child clusters

The result of an experiment using the NTCIR data set is reported.

This working note is a quick report of the outline of the work and details will be described in the full paper which will appear in the formal proceedings.

In this working note, as the problem setting, the generative model based clustering is quickly described in section 2 with a simple example model. Section 3 describes a new model with word selection and section 4 describes the method for giving an upperbound of numbers of clusters. Section 5 describes the experiment with NTCIR data and section 6, the conclusion.

2. Generative model based clustering

2.1. Generative model based clustering

A generative model based clustering suppose a model with unknown parameters which can produce the given data set[3],[1]. Among those parameters are the "cluster indices" which denote which item belongs to which cluster. The clustering problem is then described as the parameter estimation problem or model selection problem. By the Bayes rule, we have: $P(Model|Data) \propto P(Model)P(Data|Model)$, so to maximize the posterior of the model (left hand), we could maximize the right hand instead. If we assume all models are equally probable in their prior, we can get the best model by searching for the best model which maximizes the likelihood $P(Data|Model)$.

To balance the expressibility and generality, the BIC (Bayes Information Criterion)([5])below is often used as the melkmar.

$$BIC \equiv \log P(Data|Model) - \frac{\#of\ \ parameters}{2}\log(\#of\ \ data)$$

In that regard, in order to get "best" clustering result, one can search for the model which maximizes the BIC. Because the cluster indices are "hidden", one has to get the model through some indeterministic approach, for examle, using the EM algorithm.

2.2. A simple example

Here we show a simple example of a generative model, which is called a bag-of-words model, where the whole data set is regarded as a collection of documents and each document is regarded as a bag of words (i.e. word order is neglected), and where each word token in the data set is regarded to have derived in the following manner:

1. choose the document d it appears in
2. choose the cluster (= category) c it belongs to (a word token is considered to belong to a cluster)
3. choose its word type w

(here, "word type" mean the string form of a word and "word token" means its occurrence in a document. We use hereafter “word” for “word type” when there is no confusion)

And we put an assumption that $p(w|d,c=)p(w|c)$

Therefore, "the probability of a word token to appear in document d, belong to cluster c, and have word type w" is $p(d)p(c|d)p(w|c)$

Figure 1 shows the graphical representation (called a Bayesian Network ([3])) of the model here. Each node represents a variable, and a directed edge between two nodes means a (direct) dependency between those variables. Here the node D corresponds to d (document), C to c, W to w .Each node has a CPT (conditional probability table), which describe the conditional probability of the corresponding variable given the variables corresponding to its parent nodes (node A is said to be a parent of node B when there is a directed edge heading to B) Then the joint probability of the data is

$$k!\prod_w \prod_c \prod_d \left\{p(d)p(c|d)p(w|c,d)\right\}^{N_{dwc}}$$

$$= k!\prod_d p(d)^{N_d} \cdot \prod_c \prod_w p(w|c)^{N_{wc}}$$

where k denotes the number of clusters, N_{dwc} denotes the number of tokens in document d, of word type w and belong to cluster c, and $N_d \equiv \sum_c \sum_w N_{dwc}$, $N_{wc} \equiv \sum_d N_{dwc}$

(k! comes from the interchangeability of the hidden variable c's)

With $D \equiv |d|$, $M \equiv |w|$, $N \equiv \sum_d \sum_c \sum_w N_{dwc}$ the number of the variables is $D + D(k-1) + k(M-1) = k(D+M-1)$, yielding

$$BIC = \sum_{i=1}^{k} \log i + \sum_d N_d \log p(d)$$
$$+ \sum_c \sum_w N_{wc} \log p(w|c)$$
$$- \frac{k(D+M-1)}{2} \log N$$

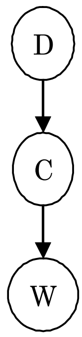

Figure 1 A simple generative model

2.3. Hierarchical clustering

Hierarchical clustering approaches are often claimed to lead to a good result. Our clustering method is also a hierarchical one. However, it is a greedy one, i.e. after choosing the best clustering in a certain level, say $l-1$, we proceed to search for the best clustering of the child clusters in level l. Some other approaches make the set of whole hierarchical models first and then chose the best from them ([6]).

3. A new model with word selection

Here we introduce a new generative model, as an extension to the simple model in 2.2, where a new variable r (relevance flag) which denotes whether a word token is relevant in clustering ($r=1$) or not ($r=0$). Each word token in the data set is regarded to have derived in the following manner:

1. choose the document d it appears in
2. choose the cluster c it belongs to (a word token is considered to belong to a cluster)
3. choose whether it is one of the relevant words (word types) to the clustering
4. choose its word type w

And we assume $p(r|d,c) = p(r|c)$, $p(w|d,c,r) = p(w|c,r)$.

Therefore, "the probability of a word token to appear in document d, belong to cluster c, have relevance flag r, and have word type w" is: $p(d)p(c|d)p(r|c)p(w|c,r)$

Figure 2 shows the graphical representation of the model here.

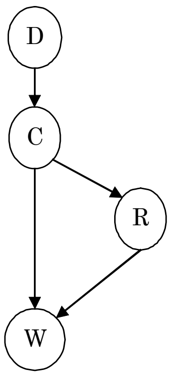

Figure 2: An extended graphical model

We further assume that word tokens of the same word type are either all relevant (saying the word type is relevant) or irrelevant (saying the word type is irrelevant), and w is independent from c given r =0.

Here we introduce some additional notations to those in 2.2:

$$W_1 \equiv \{w | p(r=1|w)=1\},$$

$$W_0 \equiv \{w | p(r=0|w)=1\},$$

$$N_{W_1 c} \equiv \sum_{w \in W_1} N_{wc},$$

$$N_{W_0 c} \equiv \sum_{w \in W_0} N_{wc},$$

$$N_c \equiv \sum_{w} N_{wc},$$

$$N_w \equiv \sum_{c} N_{wc},$$

$$N_{W_0} \equiv \sum_{w \in W_0} N_w,$$

$$M_1 \equiv | W_1 |,$$

$$M_0 \equiv | W_0 |$$

The probability of our having the data set is then,

$$k! \prod_d p(d)^{N_d} \cdot \prod_c \left(\frac{N_{W_1 c}}{N_c} \right)^{N_{W_1 c}} \cdot \prod_c \left(\frac{N_{W_0 c}}{N_c} \right)^{N_{W_0 c}} \cdot \prod_c \prod_{w \in W_1} \left(\frac{N_{wc}}{N_{W_1 c}} \right)^{N_{wc}} \cdot \prod_{w \in W_0} \left(\frac{N_w}{N_{W_0}} \right)^{N_w}$$

Here, the number of the variables is

$$D-1+D(k-1)+k+k(M_1-1)+(M_0-1) = k(D+M_1)+M_0-2$$

therefore

$$\begin{aligned} BIC = & \sum_{i=1}^{k} \log k + \sum_d N_d \log p(d) \\ & + \sum_c N_{W_0 c} \log N_{W_0 c} - \sum_c N_c \log N_c \\ & + \sum_c \sum_{w \in W_1} N_{wc} \log N_{wc} + \sum_{w \in W_0} N_w \log N_w \\ & - N_{W_0} \log N_{W_0} \\ & - \frac{k(D+M_1)+M_0-2}{2} \log N \end{aligned}$$

When we turn a relevant word w^* to irrelevant, the BIC will be

$$\begin{aligned} BIC^* = & \sum_{i=1}^{k} \log i + \sum_d N_d \log p(d) \\ & + N_{W_1'} \log p(W_1') \\ & + \sum_c \sum_{w \in W_1'} N_{wc} \log p(w|c) \\ & - \sum_c N_{W_1' c} \log p(W_1' | c) \\ & + \sum_{w \in W_0 \cup \{w^*\}} N_w \log p(w) \\ & - \frac{k(D+M_1-2)+M_0}{2} \log N \end{aligned}$$

where $W_1' \equiv W_1 - \{w^*\}$,

and the gain is

$$
\begin{aligned}
\Delta BIC &= N_{W_1'} \log p\left(W_1'\right) - N_{W_1} \log p(W_1) \\
&- \sum_c N_{w^* c} \log p\left(w^* \middle| c\right) \\
&- \sum_c N_{W_1' c} \log p\left(W_1' \middle| c\right) \\
&+ \sum_c N_{W_1 c} \log p(W_1 | c) \\
&+ N_{w^*} \log p\left(w^*\right) + \frac{k-1}{2} \log N \\
&= \sum_c N_{W_1 c} \left[- \frac{N_{W_1' c}}{N_{W_1 c}} \log \frac{N_{W_1' c}}{N_{W_1 c}} \right. \\
&\left. - \frac{N_{w^* c}}{N_{W_1 c}} \log \frac{N_{w^* c}}{N_{W_1 c}} \right] \\
&+ N_{W_1'} \log \frac{N_{W_1'}}{N_{W_1}} + N_{w^*} \log \frac{N_{w^*}}{N_{W_1}} \\
&+ \frac{k-1}{2} \log N \\
&\geq \sum_c \sum_{d \in c} N_{W_1 d} \left[- \frac{N_{W_1' d}}{N_{W_1 d}} \log \frac{N_{W_1' d}}{N_{W_1 d}} \right. \\
&\left. - \frac{N_{w^* d}}{N_{W_1 d}} \log \frac{N_{w^* d}}{N_{W_1 d}} \right] \\
&+ N_{W_1'} \log \frac{N_{W_1'}}{N_{W_1}} \\
&+ N_{w^*} \log \frac{N_{w^*}}{N_{W_1}} \\
&+ \frac{k-1}{2} \log N \\
&= \sum_d N_{W_1 d} \left[- \frac{N_{W_1' d}}{N_{W_1 d}} \log \frac{N_{W_1' d}}{N_{W_1 d}} \right. \\
&\left. - \frac{N_{w^* d}}{N_{W_1 d}} \log \frac{N_{w^* d}}{N_{W_1 d}} \right] \\
&+ N_{W_1'} \log \frac{N_{W_1'}}{N_{W_1}} + N_{w^*} \log \frac{N_{w^*}}{N_{W_1}} \\
&+ \frac{k-1}{2} \log N
\end{aligned}
$$

where

$$N_{wd} \equiv \sum_c N_{wdc} \ ,$$

$$N_{W_1 d} \equiv \sum_{w \in W_1} N_{wd} \ ,$$

$$N_{W_1' d} \equiv \sum_{w \in W_1'} N_{wd} \ ,$$

$$N_w \equiv \sum_d N_{wd} \ ,$$

$$N_{W_1} \equiv \sum_d N_{W_1 d} \ ,$$

$$N_{W_1'} \equiv \sum_d N_{W_1' d}$$

Therefore word w^* can be turned to be irrelevant regardless of the clustering result, if

$$
\begin{aligned}
k \geq 1 &+ \frac{2}{\log N} \left[N_{W_1} \left\{ - \frac{N_{W_1'}}{N_{W_1}} \log \frac{N_{W_1'}}{N_{W_1}} - \frac{N_{w^*}}{N_{W_1}} \log \frac{N_{w^*}}{N_{W_1}} \right\} \right. \\
&\left. - \sum_d N_{W_1 d} \left\{ - \frac{N_{W_1' d}}{N_{W_1 d}} \log \frac{N_{W_1' d}}{N_{W_1 d}} - \frac{N_{w^* d}}{N_{W_1 d}} \log \frac{N_{w^* d}}{N_{W_1 d}} \right\} \right]
\end{aligned}
$$

4. Giving an upperbound of numbers of clusters

Now we address the problem of giving an upperbound of numbers of clusters.

When there is no division, the variable r is not introduced and

$$
BIC = \sum_d N_d \log p(d) + \sum_w N_w \log p(w) - \frac{D+M-2}{2} \log N
$$

Therefore, the gain of BIC by dividing the cluster is

$$
\begin{aligned}
\Delta BIC = &\sum_{i=1}^{k} \log i + N_{W_1} \log N_{W_1} \\
&- \sum_{w \in W_1} N_w \log N_w \\
&- \frac{(k-1)(D+M_1-1)}{2} \log N \\
&+ \sum_c \sum_{w \in W_1} N_{wc} \log \frac{N_{wc}}{N_{W_1 c}}
\end{aligned}
$$

What is wanted here is the minimum k s.t. "the gain of BIC is negative when the cluster is divided into $(k+1)$ clusters no matter what the relevant word set is and the clustering is." Let us denote this k by k^*. Then

$$k^* \le \arg\min_k \forall \mathbf{c} \forall W_1 \left[\sum_{i=1}^{k+1} \log i + N_{W_1} \log N_{W_1} - \sum_{w \in W_1} N_w \log N_w - \frac{k(D+M_1-1)}{2} \log N - \sum_c \sum_{d \in c} N_{W_1 d} \left(- \sum_{w \in W_1} \frac{N_{wd}}{N_{W_1 d}} \log \frac{N_{wd}}{N_{W_1 d}} \right) < 0 \right]$$

$$= \arg\min_k \forall W_1 \left[\sum_{i=1}^{k+1} \log i + N_{W_1} \log N_{W_1} - \sum_{w \in W_1} N_w \log N_w - \frac{k(D+M_1-1)}{2} \log N + \sum_d \sum_{w \in W_1} N_{wd} \log N_{wd} - \sum_d N_{W_1 d} \log N_{W_1 d} < 0 \right]$$

where $\mathbf{c}$ denotes a clustering (= set of clusters)

When we introduce the function f, as

$$f(W_1, k) \equiv \sum_{i=1}^{k+1} \log i + N_{W_1} \log N_{W_1} - \sum_{w \in W_1} N_w \log N_w - \frac{k(D+M_1-1)}{2} \log N - N_{W_1} \min_d \left(- \sum_{w \in W_1} \frac{N_{wd}}{N_{W_1 d}} \log \frac{N_{wd}}{N_{W_1 d}} \right)$$

then

$$k^* \le \arg\max_k \exists W_1 \left[f(W_1, k) \ge 0 \right]$$

Letting $W_1(k)$ be the subset of W which doesn't include any w^*'s which satisfies $f\left(W_1', k\right) - f(W_1, k) \ge 0$, from $\forall W_1 \; f(W_1, k) \le f(W_1(k), k)$, we have

$$k^* \le \arg\max_k \left[\sum_{i=1}^{k} \log i + N_{W_1(k)} \log N_{W_1(k)} - \sum_{w \in W_1(k)} N_w \log N_w - \frac{(k-1)(D+M_1(k)-1)}{2} \log N - N_{W_1} \min_d \left(- \sum_{w \in W_1(k)} \frac{N_{wd}}{N_{W_1(k)d}} \right) < 0 \right]$$

where

$$M_1(k) \equiv |W_1(k)|$$

and

$$N_{W_1(k)d} \equiv \sum_{w \in W_1(k)} N_{wd}$$

5. Experiment with NTCIR data

5.1. NTCIR Topical Classification Task

An experiment is carried through participating the NTCIR Topical Classification Task.

Participants are given 47 queries and a set of documents for each query. The document set for each query consists of 200 documents from NTCIR's NW100G-01 data set. Participants are then required to cluster documents in each set into arbitrary numbers of clusters.

5.2. Algorithm

The following is the outline of our algorithm applied to the task. The details of the stop condition and KL-divergence based score are not described here because of space limit.

And clusters with less than 5 documents were discarded

1. Extract words using a dictionary and counting their occurrences in each document.
2. Make the root cluster which includes all the documents and put it in the cluster queue.
3. Repeat while the cluster queue is not empty:
   (a) Pick up the first cluster in the queue and name it the current cluster.
   (b) Get the relevant words to the division of the current cluster into each number of children clusters
   (c) Get the upperbound of the numbers

of clusters ($k_c^{\max}$), where c is the current cluster. Go to next cluster if $k_c^{\max} < 2$

(d) Get the BIC of the current cluster without division ($\equiv BIC^{\max} = BIC_0$).

(e) For each N from 2 to $k_c^{\max}$, repeat until a stop condition is met

① Get N child clusters by estimating the $p(c'|c,d)$'s where d is each document and c' is a child cluster of c using an EM algorithm.

② Calculate the BIC for the resultant model (BIC_1).

③ if $BIC_1 > BIC_0$, (re)mark this result as the best clustering and $BIC^{\max} \leftarrow BIC_1$

(f) If $BIC^{\max} > BIC_0$, put the child clusters in the best clustering to the top of the cluster queue, else mark the current cluster undividable.

4. Output the resultant leaf clusters (= clusters without children) with relevant key words using a KL-based score (the details will be described in the full paper).

5.3. Quick overview of the result

By now the results for 11 queries are evaluated by the NTCIR staffs and the evaluation results are given to each participant. The following is a quick overview of our result. Detailed analyses of the evaluation of our results are yet to be done.

The numbers of the relevant words to each clustering varies from 0 to 39,480 with the mean being 862.1 and standard deviation being 2740. Figure 3 shows the histogram of numbers of relevant words.

The numbers of the children turned out to be considerably smaller than the upperbouds above during preliminary experiments and a heuristic rule was employed to quickly abort the trial. Figure 4 shows the numbers of children found.

The class size (= number of documents in the class) varies from 5 to 61, with the mean being 7.98 and the standard deviation being 6.26. Figure 5 is the histogram of leaf cluster sizes. The minimum 5 is derived from a heuristic restriction setting of the minimum size to 5. Figure 6 is the box and whisker graph of cluster sizes for each query.

The result of the official evaluation (average base) by NTCIR standards are listed in Table 1.

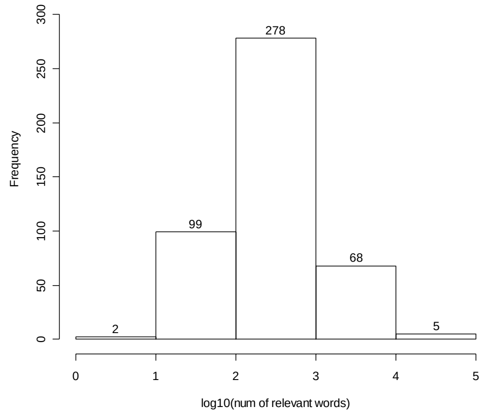

Figure 3: Numbers of relevant words

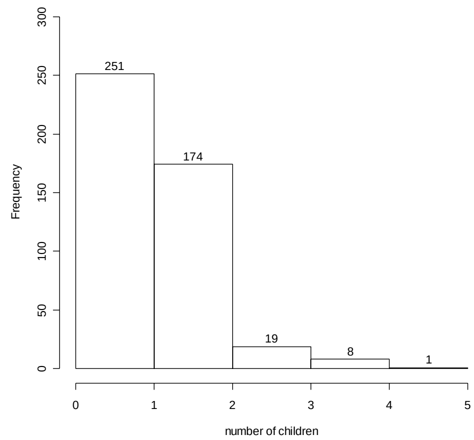

Figure 4 Histogram of numbers of children

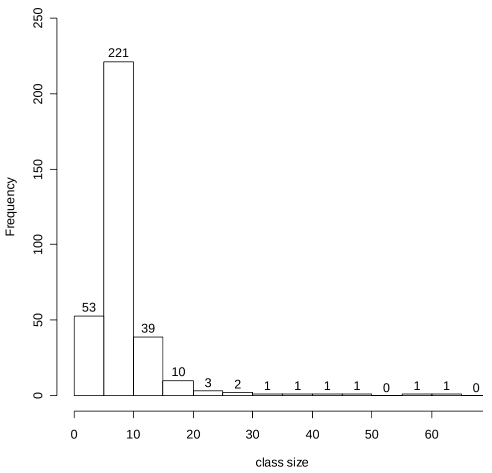

Figure 5 Histogram of (leaf) cluster sizes

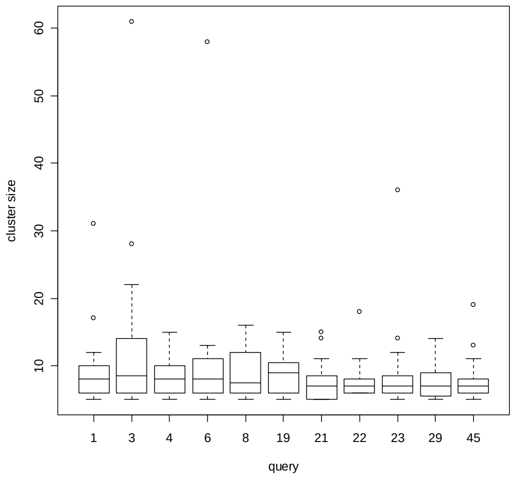

Figure 6 Cluster sizes for each query

Table 1 Evaluation results by NTCIR standards (average)

	relax	rigid
Avg. Precision	0.21	0.22
Precision @20	0.40	0.35
Recall @20	0.48	0.53
F-value@20	0.30	0.27
cg	8.09	6.91
dcg	3.01	2.64
mdcg1	2.80	2.45
mdcg2	7.25	6.14

6. Conclusion

A new generative model based hierarchical clustering method is proposed, where the upperbounds of relevant words and number of clusters are calculated by introducing a class indices and using the BIC for model selection.

Future work includes

- more precise upperbouding of the number of the clusters.
- consideration of interdependence between word types
- use of variational methods
- comparison of BIC to other information criteria

References

[1] Blei, D.M., Ng, A. Y. and Jordan, M. I. : Latent Dirichlet Allocation, *NIPS 14*, 2003

[2] Deerwester, S., Dumais, S.T., Furnas, G.W., Landauer, T.K., and Harshman, R. : Indexing by latent semantic indexing. *Journal of the American Society for Information Science*, 41 (6), pp.391-407, 1990.

[3] Hofmann, T. : Probabilistic latent semantic indexing, *Proceedings of the Twenty-Second Annual International SIGIR Conference*, pp. 50-57, 1999.

[4] Pearl., J: *Probabilistic Reasoning in Intelligent Systems*, Morgan Kaufmann, 1988.

[5] Schwartz, G. : Estimating the dimension of a model, *The Annals of Statistics*, 6, pp.461-464, 1978

[6] Vinokov, A. & Girolami, M. : A Probabilistic Framework for the Hierarchical Organisation and Classification of Document Collections, *Journal of Intelligent Information Systems*, 18 (2), pp.153-172, 2002.