Pp. 52–60 in *Second Workshop on Semantic Web Mining*, edited by B. Berendt, A. Hotho and G. Stumme. KD-Net, 2002.

Multi-faceted Learning for Web Taxonomies

Wray Buntine and Henry Tirri

Helsinki Inst. of Information Technology
HIIT, P.O. Box 9800
FIN-02015 HUT, Finland
{wray.buntine,henry.tirri}@hiit.fi

Abstract. A standard problem for internet commerce is the task of building a product taxonomy from web pages, without access to corporate databases. However, a nasty aspect of the real world is that most web-pages have multiple facets. A web page might contain information about both cameras and computers, as well as having both specification and sale data. We are interested in methods for supervised and unsupervised learning of multiple faceted models. Here we present results for multi-faceted clustering of bigram words data.

1 Introduction

A recognized problem for internet commerce is the task of building a product taxonomy from web pages, without access to corporate databases, and then populating a database with link information about service, repair, spare parts, reviews, product specifications, product family and company home pages, and purchase information from retailers. A key precursor for this task is the ability to build classification hierarchies in a supervised or unsupervised manner. However, a nasty aspect of the real world is that most web-pages have multiple facets. A web page might contain information about cameras and computers, as well as having both specification and sale data. Whereas another page might mix a product index with partial specification and sales data. We are interested in methods for supervised and unsupervised learning of multi-faceted models.

Clustering or unsupervised learning is now a standard method for analysing discrete data such as documents, and is now being used in industry to create taxonomies from web pages. A rich variety of methods exist borrowing theory and algorithms from a broad spectrum of computer science: spectral (eigenvector) methods [1], kd-trees [2], using existing high-performance graph partitioning algorithms from CAD [3], hierarchical algorithms [4] and data merging algorithms [5], etc.

All these methods, however, have one significant drawback for typical application in areas such as document or image analysis: each item/document is to be classified exclusively to one class. Their models make no allowance for instance, for a product page to have 60% digital camera content and 40% laptop computer content. It is 100% one way or another, and any uncertainty is only about whether to place the 100% into one or the other class. In practice documents invariable mix a few topics, readily seen by inspection of the human-classified Reuters newswire, so the automated construction of topic hierarchies needs to reflect this. One alternative is to make clustering *multi-faceted* whereby a document can be assigned proportionally (i.e., using a convex combination) across a number of clusters rather than uniquely to one cluster.

Authors have recently proposed discrete analogues to principle components analysis (PCA) intended to handle discrete or positive only count data of the kind used in the bag-of-words representation of web pages. Methods include non-negative matrix factorization [6], probabilistic latent semantic analysis [7] latent Dirichlet allocation [8], multinomial PCA [9]. A good discussion of the motivation for these techniques can be found in [7], and an analysis of related reduced dimension models and some of the earlier statistical literature here can be found in [10], and theory and algorithms are presented in [9].

Multinomial PCA by itself does not perform multi-faceted clustering because on average a page/item/document might be composed of up to 50 components in some of our experiments, and this does not reflect the behaviour of a "few different topics" we were looking for. We have recently made modifications to the standard algorithms so that multinomial PCA performs multi-faceted clustering. That is, it performs a clustering whereby some items/documents/pages are assigned with proportion to a few different classes.

In this paper, we first expand more on what we mean by a multi-faceted model, and then we give some examples from our working system. Our experiments are conducted on bigram data for words collected off a good fraction of Google's database of web pages for August 2001. The data sizes allowed us to experiment to understand how many components might be produced.

2 Contrasting Clustering and Multi-Faceted Clustering

For concreteness, consider the problem in terms of the usual "bag of words" representation for a document [11]. Here the items making up the sample are documents and the features are the counts of words in the document. A document is represented as a sparse vector of words and their occurrence counts. All positional information is lost. With J different words, the dimensionality for words/features, each document becomes a vector $\boldsymbol{x} \in \mathcal{Z}^J$, where the total $\sum_j x_j$ might be known. Traditional clustering becomes the problem of forming a mapping $\mathcal{Z}^J \mapsto \{1, \ldots, K\}$, where K is the number of clusters. Whereas techniques such as PCA form a mapping $\mathcal{Z}^J \mapsto \mathcal{R}^K$ where K is considerably less than J.

The problem we consider, however, is to represent the document as a convex combination, thus to form a mapping $\mathcal{Z}^I \mapsto \mathcal{C}^K$ where $\mathcal{C}^K$ denotes the subspace of $\mathcal{R}^K$ where every entry is non-negative and the entries sum to 1 ($\boldsymbol{m} \in \mathcal{C}^K$ implies $0 \leq m_k \leq 1$ and $\sum_k m_k = 1$). Call $\boldsymbol{m}$ the reduced image of a document.

For instance, suppose we are performing a coarse clustering of newswires into topics: the topics found might be "sports", "business", "travel", "international", "politics", "domestic", and "cultural". Consider a document about a major sport-star and the overlap of his honeymoon with a big game. Then traditional clustering might output the following: "the document is about sports". A more refined clustering system that represents uncertainties as well might output: "with 90% probability it is about sports, with 7% probability it is about cultural, and 3% probability about something else". General multinomial PCA considered in this paper might output: "50% of the document is about sports, 35% of the document is about cultural, 7% about business, 5% about international". The supposed business content is really a discussion of the hotel for the honeymoon and the supposed international content comes from the location of the honeymoon. Note here general multinomial PCA plays the role of dimensionality reduction, and places similar kinds of words into the same bucket for compression purposes rather than any real topic identification.

The problem we consider is also to perform multi-faceted clustering, which serves the purpose of extracting multiple mutually occurring topics from a document. Suppose $\boldsymbol{m} \in \mathcal{C}^K$ is the reduced image of a particular document. For multi-faceted clustering, $\boldsymbol{m}$ should have most entries zero, and only a few entries significantly depart from zero. A measure we shall use for this is entropy, $H(\boldsymbol{m}) = \sum_j m_j \log(1/m_j)$. Thus multi-faceted clustering prefers low entropy reduced images from $\mathcal{C}^K$. In the limit, when the average entropy of the reduced images is 0, the mapping becomes equivalent to standard clustering. With the simple honeymoon example above, the output could be reduced to: "70% of the document is about sports, 30% of the document is about cultural". This makes the document have $2^{H(\boldsymbol{m})} = 1.85$ effective topics, as opposed to the original PCA example above with more proportions (0.5,0.35,0.07,0.05) which had $2^{H(\boldsymbol{m})} = 3.17$ topics.

Note that clustering is usually varied using the dimensionality K, their number of components, not the nature of their decomposition, $H(\boldsymbol{m})$. Approaches such as the Information Bottleneck method [12] and hierarchical approaches in general yield clustering at different scales and do not relax the assumption of mutual exclusivity. We make the distinction here between the term *component* which is a derived feature discovered for dimensionality reduction, and a *facet*, which is similar but is intended instead to be a relaxation of a topic. Documents should have a few facets for good multi-faceted clustering but many components for effective dimensionality reduction.

3 Theory

We briefly review the theory of the multinomial version of PCA, and discuss the extensions. More details of the basic theory appear in [9].

Given a document, we first to sample a K-dimensional probability vector $\boldsymbol{m}$ that represents the proportional weighting of components, and then to mix it with a $K \times J$ matrix $\boldsymbol{\Omega}$ whose k-th row represents a word probability vector for the k-th component. For a document with a total count of L words in its bag-of-words representation $\boldsymbol{x}$, this is modelled as:

$$\boldsymbol{m} \sim Dirichlet(\boldsymbol{\alpha}) \ ,$$

$$\boldsymbol{x} \sim \mathit{Multinomial}(\boldsymbol{m}\boldsymbol{\Omega}, L) \ ,$$

where $\boldsymbol{\alpha}$ is a vector of K-dimensional parameters to the Dirichlet. Thus, the mean of each entry x_j is a convex combination of a column of $\boldsymbol{\Omega}$, the probabilities for the j-th word for different components.

This probability model does not readily yield an algorithm. The proportion vector $\boldsymbol{m}$ is a hidden variable but it cannot be treated with the standard EM algorithm for hidden variables. However, if we can introduce a second hidden variable for each document $\boldsymbol{w}$ which is the word counts now broken out by word index j and topic index k as a $J \times K$ matrix. This matrix has row totals equal the bag of words data $\boldsymbol{x}$. An algorithm can be derived which iteratively recomputes an expected value for both the topic proportions $\boldsymbol{m}$ and the word counts broken out by topic $\boldsymbol{w}$.

The following iterative algorithm can be derived using variational methods [9].

Theorem 1. *Given the hidden variable model above, and the priors:* $\boldsymbol{m} \sim \mathit{Dirichlet}(\boldsymbol{\alpha})$ *and* $\boldsymbol{\Omega}_{k_l,\cdot} \sim \mathit{Dirichlet}(2\boldsymbol{f})$*, where* $\boldsymbol{f}$ *is an empirical word probability vector and* $\boldsymbol{\alpha}$ *is some other vector giving priors for the first Dirichlet. The following updates converge to a lower bound of* $\log p(\boldsymbol{x}, \boldsymbol{\Omega}, \boldsymbol{\alpha})$ *that is optimal for all product approximations* $q(\boldsymbol{m})q(\boldsymbol{w})$ *for the hidden value posterior* $p(\boldsymbol{m}, \boldsymbol{w} \,|\, \boldsymbol{\Omega}, \boldsymbol{\alpha}, \boldsymbol{x})$*. The subscript* $[i]$ *indicates values from the i-th document. For this the* K*-dimensional vector* $\boldsymbol{\beta}$ *is an intermediate variable representing a Dirichlet approximation to the posterior distribution for the i-th documents proportions* $\boldsymbol{m}_{[i]}$ *and the* $J \times K \times I$*-dimensional array* $\boldsymbol{\gamma}$ *is an intermediate variable representing the multinomial probabilities for an approximation to the posterior distribution for the rows of the i-th documents word matrix* $\boldsymbol{w}_{[i]}$*.*

$$\begin{aligned}
\gamma_{j,k,[i]} &\longleftarrow \frac{1}{Z_{1,j,[i]}} \Omega_{k,j} \exp\left(\Psi_0(\beta_{k,[i]}) - \Psi_0\left(\sum_k \beta_{k,[i]}\right)\right) \ , \\
\beta_{k,[i]} &\longleftarrow \alpha_k + \sum_j r_{j,[i]} \gamma_{j,k,[i]} \ , \\
\Omega_{k,j} &\longleftarrow \frac{1}{Z_{2,k}} \left(2 f_j + \sum_i r_{j,[i]} \gamma_{j,k,[i]}\right) \ ,
\end{aligned}$$

where $\Psi_0()$ *is the digamma function, and* $Z_{1,j,[i]}$ *and* $Z_{2,k}$ *are some normalizing constants.*

The exponential in the first rewrite rule is an estimate of $m_{k,[i]})$ as $\exp(E_q\{\log m_{k,[i]})\})$ which tends to reduce the component entropy $H(\boldsymbol{m}_{[i]})$. Note the last rewrite rule is the standard MAP estimate for a multinomial parameter vector.

To reduces the entropies of the component proportions $\boldsymbol{m}_{[i]}$ even further, we use the additional updates:

$$\gamma_{j,k,[i]} \longleftarrow \frac{1}{Z_{1,j,[i]}} \Omega_{k,j} m_{k,[i]}$$

$$m_{k,[i]} \longleftarrow \frac{n_{k,[i]}}{\sum_k n_{k,[i]} - \lambda\left(H(\boldsymbol{m}_{[\boldsymbol{i}]}) - \log 1/m_{k,[i]}\right)} \ ,$$

where λ is a Lagrange multiplier that can be increased to decrease the entropy $H(\boldsymbol{m}_{[i]})$. Note this replaces the above update for $\gamma_{j,k,[i]}$. These modified updates correspond to using an entropic prior $pr(\boldsymbol{m}_{[i]}) \propto \exp\left(-\lambda H(\boldsymbol{m}_{[i]})\right)$, which is a weighted version of Brand's [13].

4 Experimental Setup

Data was collected about word occurrences from a significant portion of the English language documents in Google's August 2001 crawl. After HTML and other tokens are removed, the basic text is processed to determine the most frequent 5000 words consisting only of letters 'a'-'z' ignoring case. Their co-occurrence data, the so-called bigram data was also collected. Bigrams are only counted for contiguous words in the same phrase: not broken by punctuation (excepting '-'), line breaks or other formatting tokens. The large number of documents used allows the bigram data to be 17% non-zero for bigrams of the top 5000 words. Note, some web pages contain seemingly random text and more than enough jargon. The top word "to" has $139,597,023$ occurrences and the $5,000$-th word "charity" has $920,343$ occurrences. The most frequent bigram is "to be" with $20,971,200$ occurrences, while the $1,000$-th most frequent is "included in" at $2,333,447$ occurrences.

In this case, the role of document in the theory is played by a word, and the role of word, is played by the words appearing after this word.

The code for our system is 1300 documented lines of C, with error checking, input parsing, diagnostic reporting, and component display. The code runs comparably to a PCA algorithm, converging in maybe 10-30 iterations, depending on the accuracy required. It outputs a HTML page with internal links representing the different aspects of the multi-faceted model constructed.

To measure the component entropies, we use $2^{H(\boldsymbol{m}\,|\,d)}$ where $H(\boldsymbol{m}\,|\,d)$ is the mean of the individual entropies $H(\boldsymbol{m}_{[i]})$ for each document (which is a conditional entropy, hence the notation).

5 Experimental Results

We conducted a number of experiments as listed below.

5.1 Basic Illustration

We clustered the 5000 most frequent words on the web into 1500 different multi-faceted classes based on their occurrence in bigrams. The average effective number of components per word (measured by $2^{\langle H(\boldsymbol{m}\,|\,d)\rangle}$) using standard multinomial PCA is about 30 and the distinctions are difficult to interpret in many cases. Using the modified version of multinomial PCA which reduces the entropy of the component vector $\boldsymbol{m}$, we got this to the more manageable figure of about 3 effective components per word. Many words had a majority component with probability over 0.7, while a few had up to 20 different components.

Below we give some examples of the different facets for different words. These are presented here to illustrate the method. Note the clusters here represent *word use* which is subtlety different to *word meaning.* Look at the examples for “wedding” below to see this. Our interpretation of each facet is given in italic prior to the list of words included in the facet,

“wedding”:
- *occasion:* birthday, christmas, romantic, holiday, holidays, vacation, wedding, anniversary, happy,;
- *jewelry:* rush, mini, gold, silver, jewelry, diamond, bell, wedding,

“love”:
- *affection:* kiss, love
- *preference:* prefer, expect, recommend, need, like, probably, suggest, love, want, mean, say, require, never, think
- *emotion:* confusion, pride, stress, pleasure, danger, fear, depression, honor, pain, comfort, britain, suffering, passion, joy, glory, concern, desire, wealth, beauty, strength, escape, feeling, insight, promise, satisfaction, peace, respect, love

“four”:
- *number:* seven, eight, five, nine, six, twelve, ten, four, three, twenty, two, one
- *some/few:* few, several, five, ten, six, four, couple, three, half

“scene”:
- *event:* universe, situation, era, meal, scene, incident, lesson, province, instance, issue, game, case, event, series, state, class, mission, project, school, sale, unit
- *play/performance:* festival, scene

“efforts”:
- *group work:* initiative, initiatives, projects, proposals, collaboration, efforts, effort, programs, activities, strategies, work
- *attempts:* attempts, aim, attempt, efforts, effort
- *relationships:* minds, hearts, lives, voices, bodies, families, efforts, commitment, attention, relationship, original, work

For instance, “love” is broken into an affection term, an preference term, and an emotion term, whereas “efforts” is broken into a group work term, an attempts term, and a relationship term.

5.2 Explaining Component Dimensionality

Why does standard multinomial PCA produce different effective number of components? We varied the input in a number of ways to explore this question.

First, we ran the system with different starting dimensions for the number of components allowed. Given I documents (i.e., words in the Google data) and J words/features per document (again, words in the Google data), then with K starting components, we are attempting to reduce the $I \times J$ word count matrix to the product of a $I \times K$ document topic matrix and a $K \times J$ topic to word mapping. Some of the K topics may be rarely used and contribute little, thus the effective number of total components can be much less. We measure this as $2^{H(\boldsymbol{p})}$ where $\boldsymbol{p}$ is the K-dimensional vector of mean proportion for a topic in all documents (i.e., the mean of the rows of the $I \times K$ document topic matrix). This contrasts with the effective number of topics/components per document $2^{H(\boldsymbol{m}\,|\,d)}$ which is the conditional entropy on the $I \times K$ document topic matrix, or the mean of the entropies of the rows of the $I \times K$ document topic matrix.

Second, we down-sampled the data. We sampled without replacement from the data vector to reduce the total size of each "document" by subsampling factors of 100, $1,000$, $10,000$, $100,000$ respectively. Note the bigram word data had huge starting counts. This was done to produce "documents" of different sizes. The characteristics of the data sets produced are given in Figure 1. The X-axis is the subsampling factor. The solid line (axis on left) represents the proportion of non-zeros in the resultant data, and the dotted line (axis on right) represents the count of words in the resultant "document".

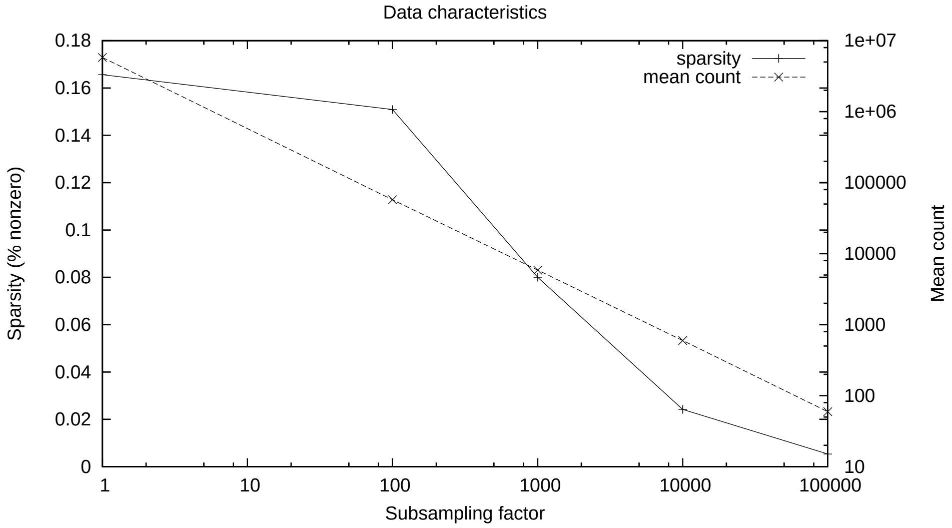

Fig. 1. Data characteristics for subsampled bigram data

The results from these experiments are reported in Figure 2. Each curve represents the results for one subsampling factor, i.e., an X value on Figure 1. So the top curve, which has no subsampling and where documents are $8,000,000$ words on average, the mean effective number of components per document increases upto about 40. The second from the bottom curve (/10000) has about 80 words per document and typically 3-4 mean effective number of components per document, no matter how many components are inferred from the data (from 20 upto about 800). The third from the bottom curve (/1000) has about 800

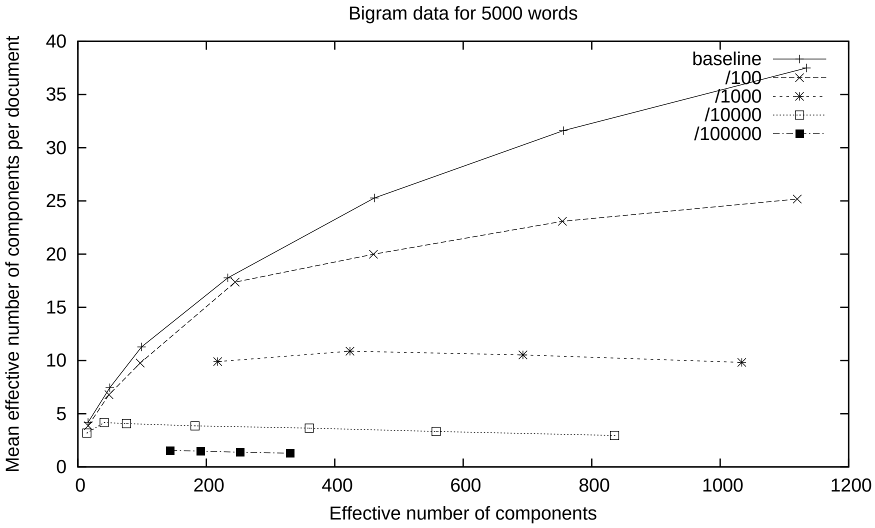

Fig. 2. Component dimensions

words per document and typically 10 mean effective number of components per document, no matter how many components are inferred from the data (from 20 upto about 1000).

From these experiments, we can conclude that the mean effective number of components is largely influenced by the document size. This would be result of the statistical capacity of the document to support a number of components. Small newswires and web pages can be 100 words, and thus could support a few topics statistically. Larger ones are about 1000 words and could support upto about 10 topics statistically. Web pages larger again that correspond to spec sheets, long details of factual data, etc., could support even more topics.

6 Conclusion

We have demonstrated that recent extensions to PCA for multinomial data are inadequate for multi-faceted clustering and given results for a modification to the basic algorithm that performs better in this regard. We argued that multinomial PCA is really a dimensionality reduction algorithm, and not designed for multi-faceted clustering.

It remains to be seen how the modified algorithm will perform on the suggested task of performing multi-faceted clustering of web-pages as a pre-processing step to data mining for the semantic web. For this, we would need at least the ability to generate full topic hierarchies, and to perform automatic labelling/naming of topics in the hierarchy. We expect that by doing this for multiple companies at once, useful hierarchies could be established.

Another research direction is to modify these algorithms to create supervised versions of them, whereby each item/document/page is tagged with multiple topics (as the Reuters and AP news-wires can be), and the task is to learn a model for the component topics and the proportions for each.

References

1. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence **22** (2000) 888–905
2. Moore, A.: Very fast EM-based mixture model clustering using multiresolution kd-tree. In: Neural Information Processing Systems, Denver (1998)
3. Han, E.H., Karypis, G., Kumar, V., Mobasher, B.: Clustering based on association rule hypergraphs. In: SIGMOD'97 Workshop on Research Issues on Data Mining and Knowledge. (1997)
4. Vaithyanathan, S., Dom, B.: Model-based hierarchical clustering. In: UAI-2000, Stanford (2000)
5. Bradley, P., Fayyad, U., Reina, C.: Scaling clustering algorithms to large databases. In: Proc. KDD'98. (1998)
6. Lee, D., Seung, H.: Learning the parts of objects by non-negative matrix factorization. Nature **401** (1999) 788–791
7. Hofmann, T.: Probabilistic latent semantic indexing. In: Research and Development in Information Retrieval. (1999) 50–57
8. Blei, D., Ng, A., Jordan, M.: Latent Dirichlet allocation. In: NIPS*14. (2002) to appear.
9. Buntine, W.L.: Variational extensions to EM and multinomial PCA. In: ECML 2002. (2002)
10. Hall, K., Hofmann, T.: Learning curved multinomial subfamilies for natural language processing and information retrieval. In: ICML 2000. (2000)
11. Baeza-Yates, R., Ribeiro-Neto, B.: Modern Information Retrieval. Addison Wesley (1999)
12. Tishby, N., Pereira, F., Bialek, W.: The information bottleneck method. In: 37-th Annual Allerton Conference on Communication, Control and Computing. (1999) 368–377
13. Brand, M.: Structure learning in conditional probability models via an entropic prior and parameter extinction. Neural Computation **11** (1999) 1155–1182