Automatic Labeling of Document Clusters

Alexandrin Popescul
University of Pennsylvania
Moore Building, 200 South 33rd Str.
Philadelphia, PA 19104−6389
(215)898−2716
popescul@unagi.cis.upenn.edu

Lyle H. Ungar
University of Pennsylvania
Moore Building, 200 South 33rd Str.
Philadelphia, PA 19104−6389
(215)898−7449
ungar@cis.upenn.edu

ABSTRACT

Automatically labeling document clusters with words which indicate their topics is difficult to do well. The most commonly used method, labeling with the most frequent words in the clusters, ends up using many words that are virtually void of descriptive power even after traditional stop words are removed. Another method, labeling with the most predictive words, often includes rather obscure words. We present two methods of labeling document clusters motivated by the model that words are generated by a hierarchy of mixture components of varying generality. The first method assumes existence of a document hierarchy (manually constructed or resulting from a hierarchical clustering algorithm) and uses a χ^2 test of significance to detect different word usage across categories in the hierarchy. The second method selects words which both occur frequently in a cluster and effectively discriminate the given cluster from the other clusters. We compare these methods on abstracts of documents selected from a subset of the hierarchy of the Cora search engine for computer science research papers. Labels produced by our methods showed superior results to the commonly employed methods.

Keywords: labeling, document hierarchy, document collection, statistical

1. INTRODUCTION

Rapid growth of the World Wide Web has caused an explosion of research aimed at facilitating retrieval, browsing and organization of on−line text documents. Much of this work was directed towards clustering documents into meaningful groups. Often, given a set or hierarchy of document clusters, a user would prefer to quickly browse through the collection to identify clusters of interest without examining particular documents in detail. E.g., Scatter/Gather is a cluster based approach to browsing large document collections [1].

Such clusters are often labeled by providing their most frequent words after removing the stop words, e.g. *for, the, and* [1−3]. The lists of the most frequent words often reveal the topic at a high level, but can fail to depict cluster−specific details as they are diluted with what we call *collection specific stop words.* E.g., in a collection of computer science research papers, terms such as *paper, method, result, system,* or *present* are very frequent and are common to most computer science subdisciplines, therefore giving no additional information to someone who already knows that all of the documents are computer science research papers. One could use the words that are the most predictive of a given cluster. These are equivalent to what have been called the most salient words in word sense disambiguation [4]. However, relatively infrequent words usually have high salience and are not suggestive of any topic. They are often simply words that are misspelled, if low frequency words are not removed. There exists a promising body of work [5−8] aimed at learning or organizing hierarchical topics of words or document collections, which assumes that words in a document follow distributional patterns and can be ascribed to different word−generating components ranging from the general to specific. This research provides ways of labeling clusters which reduce but do not eliminate the problems mentioned above. Furthermore, such methods can be difficult to implement. If labeling is all that is needed, one would prefer to use simpler methods to label clusters. Clusters can also be labeled using titles of the papers most central to a

cluster [1], or most cited, but these labels can be quite idiosyncratic and also fail to provide insights into the quality of clusters under consideration.

We present two new ways of selecting words for cluster labeling that promise to avoid the aforementioned problems. The first method assumes the existence of a document hierarchy, either manually constructed and/or populated, or a hierarchy resulting from application of a hierarchical clustering algorithm. Using χ^2 tests of independence [9] at each node in the hierarchy starting from the root we determine a set of words that are equally likely to occur in any of the children of a current node. Such words are general for all of the subtrees of a current node, and are excluded form the nodes below. The second method selects words which both occur frequently in a cluster and effectively discriminate the given cluster from the other clusters.

2. METHODS

2.1 χ^2 Method

The χ^2 test is well suited for testing dependencies when count data is available. The main idea of our method is to use χ^2 tests for each word at each node in a hierarchy starting at the root and recursively moving down the hierarchy. If one cannot reject the hypothesis that a word is equally likely to occur in all of the children of a given node, it is marked as general to the current subtree, assigned to the current node's bag of node−specific words and removed from all nodes under the current node.

The detailed description of the algorithm follows:

Input: A hierarchy of documents where the leaves contain bags of words from all of the documents in that leaf unioned together[1].

1 We currently neglect to take into account lengths of documents, but believe that doing so would not significantly affect the results.

1. Populate all internal nodes by unioning the bags of words in its children starting from the leaves and moving up to the root;
2. Start at the root and for each word perform a χ^2 test to discover dependencies:

   a) if a test rejects the independence hypothesis, conclude that the word has different probability of occurring in children and thus is specific to one or more categories down the tree;

   b) if a test fails to reject the independence hypothesis, conclude that the word is equally likely to occur in all of the children[2]. Retain the word at the current node as being general to the subtree rooted at the current node. Remove all such words from all of the nodes below the node at which the test was performed;
3. Repeat step 2 recursively moving down the tree to the leaves.

Output: A hierarchy of words isomorphic to the initial hierarchy of documents, where each node contains words specific to that node and not occurring in the subtree below the current node.

A label is a list of the most frequent words at the node corresponding to a cluster of documents we want to label. Results are presented below.

2.2. Frequent and Predictive Words Method

For the "frequent and predictive words" method , words are selected as labeling based on the product of local frequency and predictiveness:

$$\text{p(word | class)} \times \frac{\text{p(word | class)}}{\text{p(word)}}$$

2 More precisely, fail to conclude that the word has different probability of occuring in children.

This combined use of local frequency and predictiveness was used by Yarowsky [4] to select the most important words in categories for illustrating his approach of word sense disambiguation. As far as we know, this method has not been used to label document clusters.

The formula consists of two parts each having a well defined meaning: the first term, predictiveness, **p(word | class) / p(word)** is similar to a mutual information estimator [10] and TF−IDF [11] measure used in information retrieval in that is distributes more weight to the words occurring frequently in a given cluster and less weight to the words occurring frequently in all of the clusters; **p(word | class)** is frequency of the word in a given cluster and **p(word)** is the word's frequency in a more general category or in the whole collection. Words receiving high predictiveness values are good discriminators in distinguishing one cluster from another.

Words selected by this formula tend to both occur often in a cluster and be specific to the cluster. This avoids the dilution of a label by generally frequent words and by words that are obscure. One might think of selecting the most predictive words, subject to the constraint that they be statistically significant, or appear a minimum number of times, or be also on the list of frequent words. This gives less good results than taking the product of predictiveness and frequency, which does better at selecting words high on both scales.

The next section presents experimental results and comparisons between the methods.

3. EXPERIMENTAL RESULTS

The documents for the experiments were chosen from the abstracts of computer science research papers retrieved from the Cora search service [12]. A subset of the hierarchy consisting of papers on Artificial Intelligence and Information Retrieval was used. The

subhierarchy obtained was four levels deep with the root containing all Artificial Intelligence and Information Retrieval words, see Figure 1.

Figure 1. Cora Hierarchy Subset

- Root
  - Artificial Intelligence
    - Agents
    - Data Mining
    - Expert Systems
    - Games and Search
    - Knowledge Representation
    - Machine Learning
      - Case-Based
      - Genetic Algorithms
      - Neural Networks
      - Probabilistic Methods
      - Reinforcement Learning
      - Rule
      - Theory
    - Natural Language Processing
    - Planning
    - Robotics
    - Speech
    - Theorem Proving
    - Vision and Pattern Recognition
  - Information Retrieval
    - Digital Libraries
    - Extraction
    - Filtering
    - Retrieval

At each of the 22 leaves, 50 abstracts of the "most" seminal papers presented by Cora where unioned together, stemmed using the standard Porter stemmer [13], and stop words and words occurring five times or less were removed. This resulted in 1,022 unique words. This is not a large data set, but served the feasibility demonstration purpose well.

Because the χ^2 method is unreliable for low count data, we follow the frequently used rule of only applying it when the counts are greater than five.

At the root, the χ^2 tests were performed at significance level of 0.95. The tests in 122 cases did not detect dependencies in children and these 122 words were retained at the

root as general to all of the document collection. The twenty most frequent of these are:

> paper, method, result, present, approach, data, set, system, new, knowledg, task, techniqu, agent, domain, search, differ, number, applic, work, develop

All of these words are quite general to all computer science research papers.

At the next level of the hierarchy, the tests retained far fewer words, even when performed at a significance level of 0.99 to relax the dependence rigor. Only nine words were retained at the Information Retrieval node: *perform, base, problem, larg, experi, describ, collec, class, avail*, and none were retained for Artificial Intelligence. The reason fewer words were retained seems to come from the fact that the branching factor is higher − 4 for Information Retrieval and 12 for Artificial Intelligence. Naturally, there are fewer words common to all of Information Retrieval and Artificial Intelligence after removing the most general words at the root, possibly because these disciplines are quite vast, and terminology varies substantially from one subfield of another. It is not even clear if Artificial Intelligence is really a unified discipline anymore, or whether this name is simply used for historical reasons. In any case, the fields of Robotics and Natural Language Processing use very different vocabularies.

By the point the leaves are reached, very general words are excluded and labels, now represented by most frequent words after removal of more general ones, are more meaningful.

Table 1 presents the ten highest rank word labels of the most frequent and predictive words method compared to both the most frequent words method and the most predictive words method[3] for the Information Retrieval node, one of the internal nodes.

3 More than ten words received the highest score of 1 in the most predictive words method. In this table and the other result tables below, ten words serving as a label produced by the most predictive words method were chosen randomly from all the words that received the highest score.

Table 1. Information Retrieval

Most frequent	**Most predictive**	**Most frequent and predictive**
us	alloc	text
inform	area	docum
text	attribut	inform
docum	comput	retriev
retriev	cooper	user
learn	effici	web
perform	environ	extrac
paper	implem	relev
user	increas	us
result	io	librari

The variants of word *use* (stemmed as *us*) occur very frequently in all collections and are successfully downweighted by the most frequent and predictive words method. *Web* is an important word in Information Retrieval and occurs often, but not enough to be among the ten most locally frequent words, but it was upweighted due to its high predictiveness when the most frequent and predictive words method was applied. The method also avoided including words such as *result, method, process, approach* et al. The most predictive words label completely fails to reveal the topic of the cluster.

The results for the other two internal nodes (Artificial Intelligence and Machine Learning, a subcategory of Artificial Intelligence) are not as encouraging. They contain many very general words, which can be attributed to the fact that these two categories are very diverse.

To check our intuition on the relative informativeness of the labels at the leaf nodes, we conducted a small test. There three people (all PhD students in Computer Science) were given randomized lists of four labels per subdiscipline and asked to rank them on a 1 to 4

scale according to how well they believed a label represented topic. The most frequent and predictive words method received the best rankings. The χ^2 method received the second best rankings followed by the most frequent words method. The most predictive words method received the lowest rankings. The average rankings are as follows for the most frequent and predictive words, χ^2, most frequent words, and most predictive words methods respectively: 2.04, 2.21, 2.33, 3.42. Given the tiny sample size, these numbers, of course, are only indicative, not precise.

Examples of some of the leaf node labels (first ten words) are given in Tables 2−9, with each table entitled by the name of the corresponding Cora cluster; the rest of the cluster labels show similar results.

Table 2. Natural Language Processing

Most frequent	**Most predictive**	χ^2	**Most frequent and predictive**
us	bank	us	tag
learn	cooccurr	learn	text
paper	cue	model	linguist
model	flat	word	lexic
approach	grammar	algorithm	pars
word	lexic	inform	corpu
algorithm	resolv	languag	tagger
present	symbol	tag	word
inform	t	corpu	syntact
languag	tagger	research	grammar

Table 3. Robotics

Most frequent	**Most predictive**	χ^2	**Most frequent and predictive**
robot	actuat	robot	robot
us	configur	us	mobil
paper	distinct	environ	sensor
environ	give	plan	commun
task	mobil	behavior	map
plan	multirobot	research	environ
behavior	obstacl	control	manipul
research	rhino	learn	behavior
control	sensor	describ	topolog
learn	sensori	perform	navig

Table 4. Agents

Most frequent	**Most predictive**	χ^2	**Most frequent and predictive**
agent	action	inform	agent
inform	call	us	distribut
paper	capabl	distribut	coordin
us	distribut	model	multiag
task	econom	problem	mechan
distribut	fac	environ	team
present	partial	comput	cooper
model	present	research	negoti
problem	selfinteres	plan	exchang
environ	situat	coordin	ration

Note that the word *agent* is not part of the χ^2 method's list for the "Agents" cluster in

Table 4. This is undesirable, but occurred because *agent* was used similarly often in Information Retrieval and Artificial Intelligence abstracts and thus was retained at the root. This can be regarded as a version of Simpsosn's paradox; it is worth checking whether it would occur often.

Table 5. Data Mining

Most frequent	**Most predictive**	χ^2	**Most frequent and predictive**
data	column	rule	databas
rule	discov	algorithm	mine
databas	interest	mine	rule
algorithm	item	us	data
mine	metaruleguid	larg	associ
us	partition	discoveri	discoveri
associ	pass	problem	discov
method	scan	gener	frequent
larg	specifi	effici	cluster
discoveri	transac	cluster	larg

Table 6. Genetic Algorithms

Most frequent	**Most predictive**	χ^2	**Most frequent and predictive**
genet	balanc	genet	genet
program	binari	program	program
us	block	us	crossov
problem	correl	problem	ga
algorithm	crossov	algorithm	fit
function	implement	function	evolv
crossov	initi	crossov	gp
result	popul	ga	encod
paper	schema	fit	oper
ga	tradition	evolv	popul

Table 7. Rule

Most frequent	**Most predictive**	χ^2	**Most frequent and predictive**
learn	claudien	learn	claus
algorithm	defini	algorithm	logic
us	horn	us	induct
exampl	idea	problem	posit
problem	ilp	theori	exampl
method	liter	claus	special
theori	outlin	program	learn
search	regres	rule	prune
paper	stop	logic	search
result	view	induct	ilp

Table 8. Speech

Most frequent	**Most predictive**	χ^2	**Most frequent and predictive**
speech	asr	speech	speech
recogni	baselin	recogni	recogni
model	filter	model	speaker
us	incorpor	us	word
word	phone	word	pronunci
speaker	phonem	speaker	recogn
perform	reverber	perform	rate
probabl	spectral	probabl	signific
system	syllabl	error	speak
error	utter	recogn	spontan

Table 9. Digital Library

Most frequent	**Most predictive**	χ^2	**Most frequent and predicitve**
server	bandwidth	server	server
librari	coeffici	librari	digit
inform	file	inform	librari
digit	increas	digit	imag
imag	map	imag	distribut
us	materi	us	metadata
perform	multipl	user	search
user	respons	queri	schedul
queri	solution	web	servic
data	www	process	request

4. CONCLUSIONS AND FUTURE WORK

This paper has presented several methods of labeling document clusters by selecting topic−revealing keywords. The most frequent and predicitve words method produced the best labels, capturing the words which both occur frequently in a cluster and effectively discriminate the given cluster from the other clusters, but the χ^2 method also outperformed labeling by either most frequent or most predicitve words. The χ^2 method also successfully identified a set of *collection specific stop words,* words that are common to a given collection of documents, but are not part of the traditional stop word list, and lack any descriptive power to someone browsing the document collection.

The χ^2 method checks to see if word frequencies differ in **any** of the child nodes. This lends to poor performance in hierarchies with high branching factor. The method could be improved by checking for subsets of the child nodes where words have similar frequencies and excluding such words from these children while retaining them in the other children.

Unfortunately, none of the methods gave uniformly satisfactory results at the internal nodes of the hierarchy. This could well be a feature of the document collection, showing that the disciplines corresponding to the internal nodes, Information Retrieval, Artificial Intelligence, and Machine Learning, are very diverse in the vocabulary used and encompass very broad topics.

5. ACKNOWLEDGMENTS

We thank NEC Research Institute for partial support of this work.

REFERENCES

[1] Cutting, D. R., Karger, D. R., Pedersen, J. O., and Tukey, J. W., Scatter/Gather: a Cluster−Based Approach to Browsing Large Document Collections. *In Proceedings of the 15th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 318−29 (1992).

[2] Baker, L. D., and McCallum, A. K., Distributional Clustering of Words for Text Classification, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* , pp. 96 − 103 (1998).

[3] Sahami, M., Hearst, M., and Saund, E., Applying the Multiple Cause Mixture Model to Text Categorization, *In ICML−96: Proceedings of the Thirteenth International Conference on Machine Learning,* pp. 435−443, San Francisco, CA: Morgan Kaufmann (1996).

[4] Yarowsky, D.,"Word−Sense Disambiguation Using Statistical Models of Roget's Categories Trained on Large Corpora ,"*In Proceedings, COLING−92.* Nantes, pp. 454−460 (1992).

[5] Pereira, F., Tishby, N., and Lee, L., Distributional Clustering of English Words,"*30th Annual Meeting of the Association for Computational Linguistics*, pp. 183−190 (1993).

[6] McCallum, A., Rosenfeld, R., Mitchell, T., and Ng, A.Y, Improving Text Classification by Shrinkage in a Hierarchy of Classes, *In Proceedings: the Fifteenth International Conference on Machine Learning,* Morgan Kaufmann (1998).

[7] Hofmann, Th., Learning and Representing Topic: A hierarchical Mixture Model for Word Occurrences in Document Databases, *Conference for Automated Learning and Discovery, Workshop on Learning from Text and the Web, CMU* (1998).

[8] McCallum, A., Nigam, K., Rennie, J., and Seymore, K., Building Domain−Specific Search Engines with Machine Learning Techniques, *AAAI−99 Spring Symposium on Intelligent Agents in Cyberspace* (1999).

[9] Tahmane, A. C., and Dunlop, D. D., Statistics and Data Analysis, *Prentice Hall*, pp. 299−330 (2000).

[10] Cover, T. M., and Thomas, J. A., Elements of Information Theory, *New York :*

Wiley (1991).

[11] Salton, G., Developments in Automatic Text Retrieval, *Science 253*:974−80 (August 30, 1991)

[12] Cora, Computer Science Research Paper Search Engine, http://www.cora.jprc.com/.

[13] Porter, M. F., An Algorithm for Suffix Stripping, *Program*, 14(3):130−137 (1980).