Paper 086-2007

Making Web Results Relevant with SAS®

Russell Albright, Jake Bartlett, and David Bultman, SAS Institute Inc., Cary, NC

ABSTRACT

Many companies search the Web to learn about their competition and understand their potential customers. But how accurate are these search results? For instance, have you ever submitted the query "SAS", only to get results back about "Scandinavian Airline Systems"? This paper presents a SAS-based solution to accessing and clustering Yahoo! search engine results by using SAS® Text Miner. We demonstrate how to use matrix factorization techniques, clustering algorithms, and visualizations to discriminate between subsets of documents that are returned as the result of a query.

INTRODUCTION

Searches, whether they are applied to Web pages, customer comments, or employee surveys, are the primary way for users to understand and navigate large, unexplored document collections. Search results obtained from submitted key words and phrases provide a perspective on the collection from which we can learn and discover the content that is available. However, these queries can only go so far in returning documents that are relevant to the researcher. The number of documents returned can be more than what is manageable, and the result set provided by the search engine may still be largely heterogeneous even though each returned document contains the query term(s).

Most search engines such as Yahoo! and Google return the results of a query as a list of Web pages that are ranked by their relevancy to the query. No information is provided about the relationships between the elements in the list, and no feedback is provided about the scope of the content contained in the result set. Because of this, many attempts have been made to cluster the search results and label each group in a useful way. The advantages of clustering the search results have been documented (Zamir and Etzioni 1998) and they include:

1. Clusters provide users with an immediate global view of the result set.
2. They allow users to navigate the information in a reasonable way.
3. They provide immediate negative feedback if the query missed the mark retrieving relevant documents.

Refining a search by adding keywords does not eliminate the benefit of clustering the search results. For instance, when a user submits a general term such as “sas” to a search engine, results can include SAS Institute, Scandinavian Airlines, and Radisson SAS Hotels, just to name a few. The results seem very diverse. If the user refines the query by searching on the two terms “sas” and “software”, the results will be more focused but, relative to the new result set, ambiguity will remain. The results of the refined query can be segmented into those that are related to the sale and marketing of SAS software, those that are related to how to program in SAS, those that have to do with real-world applications of SAS software, and so on.

There have been some successes in building Web search engines that automatically cluster and report on the results. The Northern Light Group (www.northernlight.com) has provided categorized searches of for-fee content for over 10 years, and Vivisimo (clusty.com) has an online version that is becoming an increasingly popular search engine.

In this paper, we demonstrate how the functionality available in SAS Text Miner 3.1 and the new technology available with SAS Text Miner 4.1 (SAS® 9.2) enables users to access, understand, and analyze search results. Our focus is separated into the following three parts:

1. Accessing and retrieving the text we wish to analyze.
2. Clustering and labeling search results using a statistical clustering methodology.
3. Clustering and labeling search results using a factorization-based clustering approach.

In the next section, we present some background information about SAS Text Miner. Following the background information, we demonstrate how to access query results from the Yahoo! Web Search service using Base SAS® code. The code submits the query, retrieves the search results, and creates a data set out of the query results. In this section, we also create an extension node in SAS® Enterprise Miner™, which serves as a user interface for the Yahoo! search engine and outputs a data set that can be used directly by the Text Miner node.

Once the search results are retrieved from Yahoo!, we parse the results and cluster them using two different unsupervised techniques. The first approach uses the default clustering in SAS Text Miner 3.1 and then improves on these results by post-processing the cluster results.

The second approach allows for the possibility for a single document to appear in multiple clusters. For this approach, we use a new topic factorization methodology, which will be new in SAS Text Miner 4.1. It produces a factorization that can be used both to identify the clusters and to obtain cluster labels.

Each approach presented provides precision and recall scores (discussed in the next section) so that a comparison between the approaches can be quantified.

BACKGROUND

SAS Text Miner is able to accomplish tasks either by exploratory methods, such as clustering documents, or by confirmatory methods, such as classifying documents and combining structured information to do predictive modeling. It does this by first transforming the text found in documents into a numeric representation and then applying data mining algorithms to this representation. In SAS terminology, unstructured documents are converted to observations and the variables belonging to the observations correspond to frequencies (or weighted frequencies) of the kept terms in the documents. Because the number of variables is very large (the number of distinct kept terms in the collection) and sparse (mostly zeroes), typically some kind of data reduction is done on these variables prior to mining.

SAS TEXT MINER FUNCTIONALITY

At the heart of the SAS Text Miner 3.1 node are three key components: parsing, transformation, and clustering. (Predictive modeling, another important aspect of text mining, is done in the broader context of SAS Enterprise Miner.) The key components are summarized here:

- **Parsing –** SAS Text Miner parsing translates the textual contents of a data set variable (or the contents of a file reference to the text) into a quantitative representation. It identifies the terms found in the text and assigns each term a unique key. It then outputs a data set containing all the terms in a collection and a data set identifying how frequently each term occurred in each document. SAS Text Miner not only parses text, but it also can identify entities such as COMPANY, VEHICLE, and ADDRESS as well as noun groups, stems, and parts of speech.

- **Transformation** – Following parsing, the Singular Value Decomposition (SVD) is used on the data as a dimension reduction technique. Details of the approach are available (Albright 2004). The term-by-document frequency data set produced by the parsing is transformed into a data set that characterizes every document as a single *k*-dimensional point in space, where *k* is chosen by the user or set by the software.

- **Clustering** – SAS Text Miner uses the expectation-maximization algorithm in order to fit *k* probability distributions to the document data set produced by the transformation listed previously. Just as in the *k*-means algorithm, this technique requires *k* seeds to begin. But unlike *k*-means, the algorithm attempts to find the optimal mixture of *k* distributions that best predict the data. Each distribution is then interpreted as a cluster.

PRECISION AND RECALL

Precision and recall are the standard statistics used in information retrieval and text mining to quantify the accuracy of a query result or a binary categorization (Chakrabarti 2002). The calculations follow directly from the entries of a 2-by-2 contingency table, which can be calculated with the FREQ procedure.

If we define the following:

- *A* – the number of documents that are predicted to be in a category and that actually belong to the category,
- *B* – the number of documents that are predicted to be in a category, and
- *C* – the number of documents that actually belong to the category.

Then precision (prec), prec=*A*/*B*, gives an indication of the quality of the contents assigned to the category and recall (rec), rec=*A*/*C*, gives an indication of the quantity (as a percentage) of correct documents that were assigned to the category.

When precision and recall information is gathered on several categories, a single precision and recall value can be derived by computing the microaveraged precision and microaveraged recall. Because some categories may be much smaller than others, rather than average the precision and recall across the categories, the microaveraged precision and recall essentially treats the computation as one big contingency table over all the categories. The result can be interpreted as a weighted average based on the size of each category.

ACCESSING THE WEB WITH A SEARCH API

Many Web sites now provide Web services as a way to enable users to make use of their content and to encourage developers to build new applications with the data. Yahoo! provides many such Web services including services for photos, shopping, travel, and Web search. We will be using their Web Search Application Programming Interface (API) to access their Web Services.

THE YAHOO! SEARCH API

The Yahoo! Web Service uses Representational State Transfer (REST) style API via HTTP GET requests. The parameters of the request are encoded into the submitted URL. For example, if we wanted to search for PDF files that contain the phrase “finance”, the following URL would be submitted:

```
http://search.yahooapis.com/WebSearchService/V1/webSearch?appid=YahooDemo&query=fina
nces&format=pdf
```

The URL consists of several components:

- **http://search.yahooapis.com** – The host name of the service.
- **/WebSearchService/V1/** – The particular service of interest, Yahoo! Web Search (as opposed to Yahoo! Photo Search or Yahoo! Map Search).
- **webSearch?** – The method that is being called.
- **appid=YahooDemo** – The unique application ID for the user. While Yahoo! does not charge for this service, they do monitor its use. The ID can be obtained by visiting Yahoo!’s Web site and registering for one.
- **&query=finances&format=pdf** – The actual query, which takes the form *argument*=*value* (where the arguments and values are URL-encoded). In this case, the query term is “finances” and we are restricting the search to PDF files.

For further information, see the Yahoo! API documentation, which lists all of the mandatory and optional parameters, in addition to describing the response structure, rate limits, and so on.

The results of the URL submission are returned as XML. A sample of the contents of the XML is included:

```
<?xml version="1.0" encoding="UTF-8"?>
<ResultSet ...>
    <Result>
        <Title>
           GOVERNMENT FINANCES
        </Title>
        <Summary>
           Finances of Individual City-Counties Having 100,000 Population or ... of
           Government Finances as authorized ... government finances, are in terms
           of current ...
        </Summary>
        <Url>
           http://www.census.gov/prod/1/gov/gf92x8.pdf
        </Url>
        ...
        ...
    </Result>

    <Result>
        <Title>
           Government Finances, Public Education Finances
        </Title>
        <Summary>
           census of governments reports that focus on the finances ... requested
           that respondents for elementary-secondary finances ...
        </Summary>
```

```
        <Url>
           http://www.census.gov/ftp/pub/prod/1/gov/gc924x1.pdf
        </Url>
        ...
        ...
    </Result>
    ...
```

The XML contains the results of the query in ranked order and includes information about our query, such as the number of results returned, the title of each result, the URL for each result, a summary of the content found in that URL, and so on.

USING SAS TO RETRIEVE RESULTS

How can we fetch the XML results from within SAS? Since the results of the HTTP GET are returned as XML, fetching the results is as simple as using a filename statement in conjunction with the XML libname engine, as follows:

```
filename my_yahoo_results
"http://search.yahooapis.com/WebSearchService/V1/webSearch?appid=YahooDemo&query=fin
ances&format=pdf";
libname yahooxml xml fileref=my_yahoo_results xmlmap='C:\temp\YahooSearch.map';
```

`yahooxml.results` is now the data set that holds all of our search results. The Base SAS XML Mapper was used to create YahooSearch.map, which simply selects the XML categories of interest (Title, Summary, and URL) and maps them to variables in the generated data set.

EM EXTENSION NODES

To make our new code more accessible, we bundled it up into a SAS Enterprise Miner node by creating a new node using the Enterprise Miner extension node facility. The different parameters to our Yahoo! Search extension node are:

- **Query** – Specifies what is to be searched, for example ”finance PDF”.
- **Result start** – Specifies the Yahoo! rank of the first result you wish to retrieve. For instance, if you want to retrieve the 100th ranking document, use 100 for this parameter.
- **Result end** – Specifies the Yahoo! rank of the last result you wish to retrieve.
- **API Key** – Specifies where the app ID key goes. This identifies you to Yahoo!

The Yahoo! Search extension node then feeds its results into the Text Miner node where further analysis can be done. The screen capture of the Yahoo! search extension is presented in Figure 1. Users can receive the code to create this extension node by contacting the author listed in the "Contact Information" section.

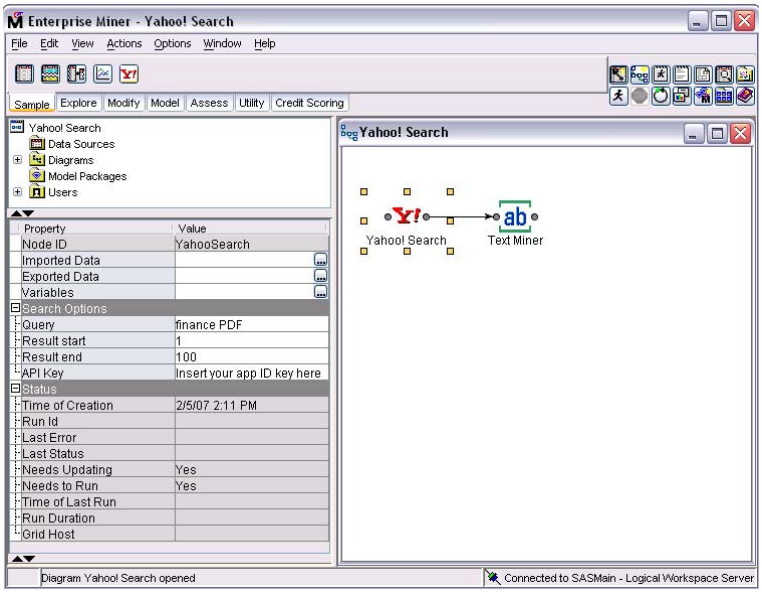


Figure 1. The Yahoo! Search Extension Node

IMPROVING THE SEARCH WITH AUTOMATIC CLUSTERING

In the following sections, we cluster and label the Web results with three different techniques. The first two are presented in the next section and are based on clustering available in the interactive Text Miner node. The final method is based on a matrix factorization. In all cases, our analysis has several key requirements:

1. For performance reasons, the clustering must only use the information returned in the initial query from Yahoo!: the URL, the summary, the title, and the rank. We did not consider analyzing the actual Web pages themselves because it would slow the performance down dramatically, and studies have shown that including the full Web page does not necessarily improve the clustering (Zamir and Etzioni 1998).
2. Clusters should have meaningful labels that assist the user in navigating the query results.
3. The analysis is limited to the first 500 results of the query.
4. The number of clusters formed should be between 8 and 12 for usability. All results that do not belong to one of the created clusters will be assigned to an “Other” cluster.

To analyze and report on the success of the approaches, we compared the result of the three approaches with detailed results for the query “sas”. Once the clusters were formed for the query results, a reviewer was asked to label each cluster created by the algorithms (without being shown the labels created by SAS Text Miner). Most clusters were easily labeled by the reviewer, but there were always a couple of clusters formed that were difficult to label. Some clusters had to be labeled N/A by the reviewer because either they actually appeared to contain two or more smaller clusters or because the clusters were simply a broad mix of topics. Because the clusters were not easily labeled, it made it impossible for the reviewer to assume the clusters were categories and to make judgments about whether a document should be assigned to a specific cluster or not.

As a result, rather than score every cluster for precision and recall as interpreted by the reviewer, the reviewer identified several key categories that were in the result set. These are listed in Table 1. Precision and recall scores were calculated for each of these categories. Since the clustering algorithms always identified several SAS Institute-related categories, these were grouped together for the purposes of computing the precision and recall.

Category	Freq.
SAS Institute	263
Special Air Services	23
Radisson SAS Hotels	11
Serial Attached SCSI	40
Scandinavian Air	18

Table 1. Frequencies for Identified Categories

CLUSTERING SEARCH RESULTS WITH THE TEXT MINER NODE

The interactive Text Miner node makes the process of clustering the result snippets straightforward. The node encapsulates parsing, the SVD transformation, and the expectation-maximization clustering. Each of these can be controlled with the settings on the Text Miner node. Figure 2 shows a screen shot of the Text Miner Interactive Window with the Documents, Terms, and Clusters tables visible. In order to get these results, we modified the default stop list to include terms that are overrepresented in any collection of Web snippets, such as "home”, “page”, “homepage”, and “web". We also turned off part-of-speech tagging because the text being analyzed did not always contain complete sentences. In these situations, terms are often assigned incorrect part-of-speech tags.

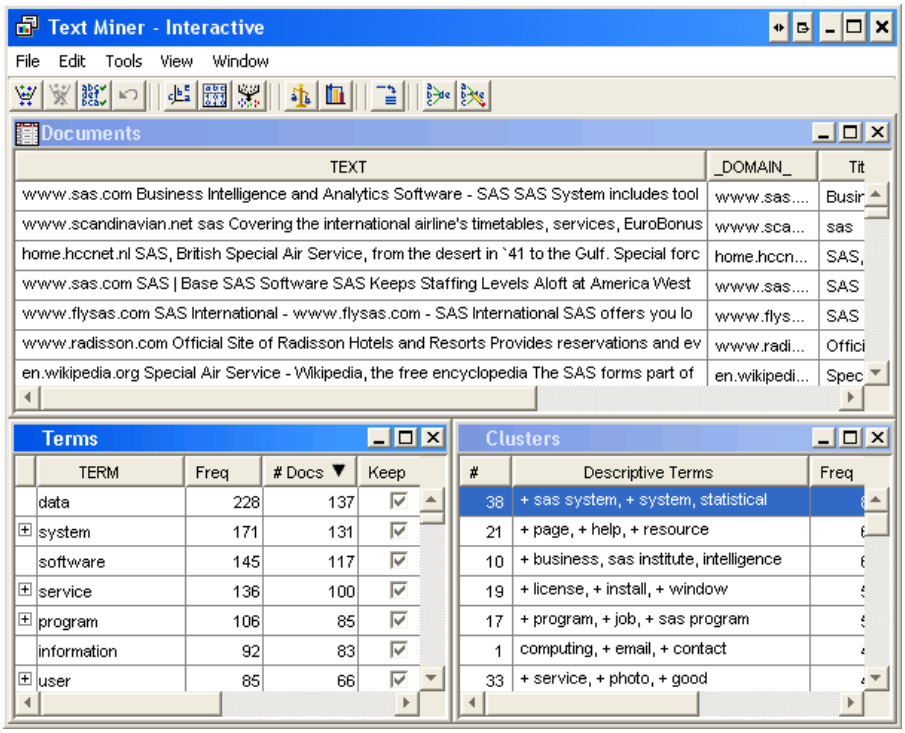


Figure 2: Text Miner Interactive Window

DEFAULT RESULTS

The default run of Text Miner produced the following 10 clusters on the top 500 results from a query of "sas". The clusters are labeled automatically by the Text Miner node. The user inputs the number of terms to use in describing the cluster and a binomial probability model to assign the terms to the cluster. In this analysis, the number of descriptive terms was set to 3.

The three-term labels from the Text Miner node are listed in Figure 3, along with a reviewer's label for each cluster.

Obs	*_CLUSTER_*	*TM_Label*	*Reviewers_Label*	*Size*	*RMS_Std*
1	1	sas institute, state, + alert	SAS Software	74	0.126
2	2	statistical, + analysis, + program	SAS Software Usage	48	0.115
3	3	air, + exam, regiment	SAS Certification Exam	52	0.117
4	4	+ hotel, radisson, + radisson sas hotel	Radisson SAS Hotels	19	0.096
5	5	+ airline, society, astronomical	N/A	84	0.127
6	6	serial, +drive, +serial attach scsi	SCSI Devices	43	0.121
7	7	+audit, clan, +service organization	N/A	30	0.127
8	8	data, +file, +sas data set	SAS Programming	49	0.114
9	9	pc, function, +window	N/A	82	0.128
10	10	fork, +sas release, +face	N/A	19	0.120

Figure 3. Default Clustering Results

We mapped the SAS Institute-related clusters to a single cluster, and used the clusters that applied to Special Air Services, Radisson SAS Hotels, Serial Attached SCSI, and Scandinavian Airlines (there weren't any that satisfied that criteria) to calculate precision and recall. The values are in Table 2.

Category	Precision	Recall
SAS Institute	.77	.65
Special Air Services	0	0
Radisson SAS Hotel	.58	1
Serial Attached SCSI	.76	.82
Scandinavian Air	0	0
Microaverage	**.67**	**.61**

Table 2. Precision and Recall for the Default Clustering

IMPROVING THE RESULTS BY TRIMMING CLUSTERS

After experimenting with several cluster runs and several queries, we discovered that forcing a large number of clusters in SAS Text Miner appeared to clarify and focus many of the obvious clusters at the expense of also forming many clusters that appeared to be noise. Because of the additional noise cluster, this approach requires a method for selecting the relevant clusters and is presented here.

For the analysis, 20 clusters were requested from the Text Miner tool. The algorithm presented here is then used after the Text Miner node to automatically select between 8 and 12 clusters for use in forming the search engine result output. Any query results contained in any of the unselected clusters were then assigned to the "Other" cluster. The criteria to select the "best" clusters were based on the following relevant information:

1. **The size and root mean square standard deviation (RMS std.) of each cluster.** The RMS std. is a measure of the goodness of the fit among observations in a cluster. A small RMS std. means that a cluster is well-defined, but our experience with text and PROC EMCLUS has shown that smaller clusters are biased toward a small RMS std. To ensure that we did not discard large clusters that have a good RMS std. relative to their size, we converted the values of each variable to ranks and then averaged these ranks to get an overall size/compactness ranking for the clusters. This serves as a good compromise between size and RMS std. This method produces ties between clusters and ties were broken based on the Yahoo! rank, presented next.

2. **The Yahoo! search result rank of the documents in the cluster.** As with most search engines, the Yahoo! engine returns the documents in a ranked order. This ranking, created by Yahoo!, is based on the Web page's content, as well as the page's connectivity to the rest of the Web. Since these measures have been shown to be good indicators of the relevance of the document, the search result rank was used to break the ties in the size/RMS std. ranking. When ties occurred between clusters at the 10th position, clusters that contained documents from the top 20 Yahoo! rank were kept and clusters that did not contain top 20 results where removed.

The variable freqrmsrank, in Figure 4, shows the result of the rank of the average rankings for _FREQ_ and_RMSSTD_. The toptwenty variable indicates which clusters contain documents that were in the top 20 results. This variable was used to decide which clusters to keep with a freqmsrrank of 10 or higher.

Obs	_CLUSTER_	_FREQ_	_RMSSTD_	clus_desc	toptwenty	freqrmsrank
1	9	30	0.09858	air, regiment, british	1	1
2	1	40	0.11506	intelligence, + business, sas institute	1	2
3	3	39	0.11395	+ analysis, data, statistical	.	2
4	8	18	0.09013	+ train, + exam, + certification	1	2
5	7	43	0.11632	+ read, + file, data	.	5
6	12	45	0.11773	+ window, unix, + run	1	5
7	20	30	0.11106	society, astronomical, club	.	5
8	18	14	0.06636	radisson, + radisson sas hotel, + hotel	1	8
9	17	17	0.09417	+ attach, + serial attach scsi, + analyzer	.	9
10	14	54	0.12587	+ airline, + pc, software	1	10
11	4	17	0.10488	storage, + drive, hard	.	11
12	19	4	0.06643	eoat, automation, gripper	.	11
13	6	18	0.11525	+ sas service, + service, management	.	13
14	13	34	0.12388	+ technic, + sample, + support	1	13
15	2	28	0.12062	+ resource, + job, group	1	15
16	11	6	0.09890	adventure, adventure travel agency, peru	.	15
17	15	7	0.09954	und, für, das	.	15
18	16	22	0.12151	+ spur, clan, san antonio	1	18
19	10	24	0.12500	+ organization, + audit, + price	.	19
20	5	10	0.11940	andrew, + school, yahoo!	.	20

Figure 4. The Initial 20 Clusters and the Top-Ranking 10 Clusters (separated by a horizontal bar)

EVALUATING THE TRIMMING RESULTS

Again, we mapped the SAS Institute-related clusters to a single cluster and gathered statistics on the other categories to compute precision and recall in Table 3. The selection method made a good improvement in precision at the expense of recall. Because increasing precision can always be done by increasing recall, the improvement was not as significant as we were hoping for. In this run, the reviewer was able to label all the clusters, without seeing the software labels. The added precision made this easier.

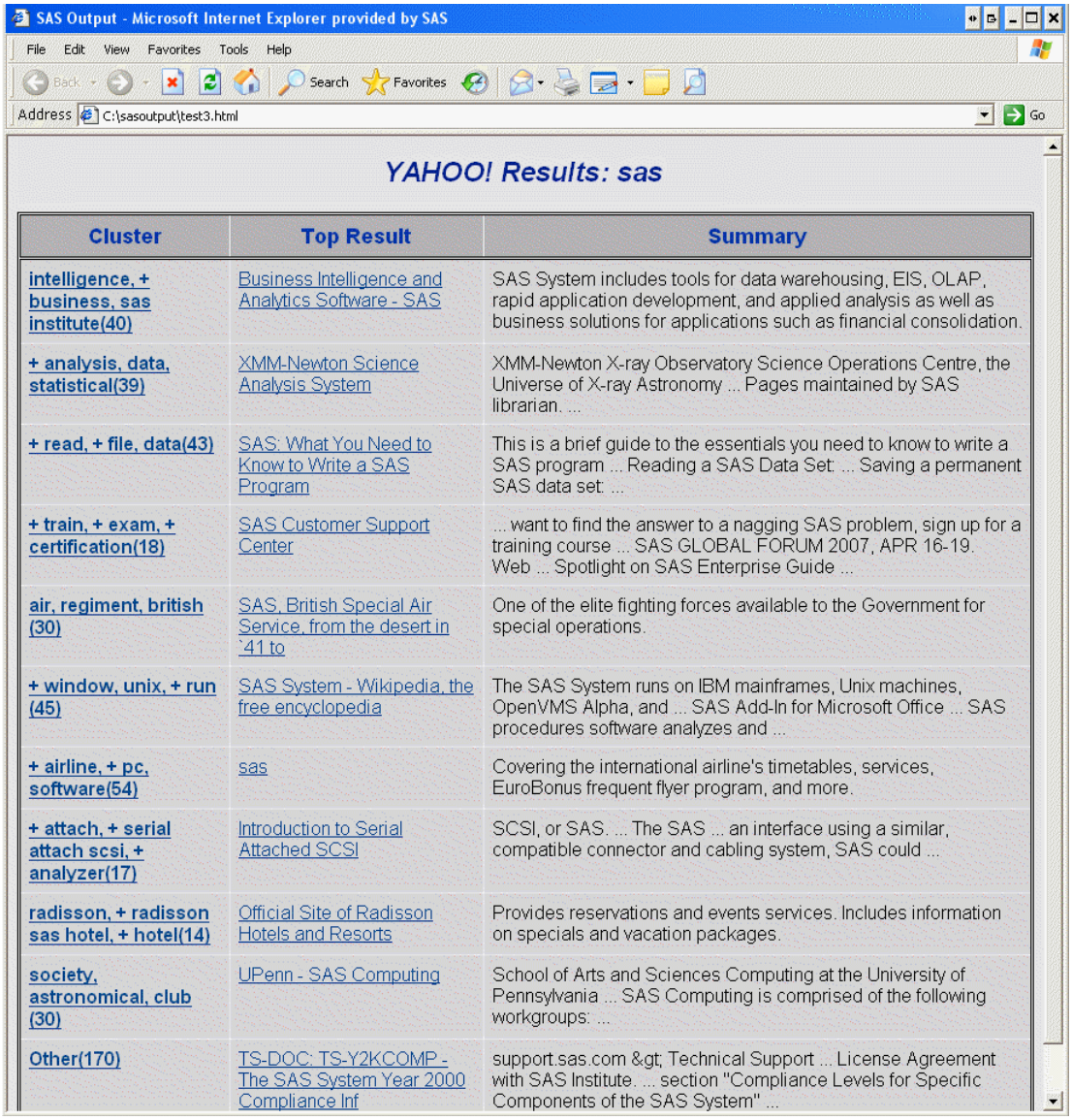
Category	Precision	Recall
SAS Institute	.86	.46
Special Air Services	.7	.91
Radisson SAS Hotel	.78	1
Serial Attached SCSI	.94	.4
Scandinavian Air	0	0
Microaverage	**.81**	**.49**

Table 3. Precision and Recall for the Cluster Reduction Method

DISPLAYING THE QUERY RESULTS WITH THE CLUSTER INFORMATION

The HTML output of the cluster results is shown in Figure 5. The page gives the user a quick summary of the main characteristics of the results by browsing the cluster labels and sizes and allows the user to navigate the results in a meaningful way. After the result data set was clustered, the HTML presented in the figure was created with the

Output Delivery System (ODS) and the REPORT procedure. If the code from the Text Miner diagram was exported, the entire process—from querying Yahoo!, to clustering, to formatting the Web pages of results—could be used in a real-time SAS application.



YAHOO! Results: sas

Cluster	Top Result	Summary
intelligence, + business, sas institute(40)	Business Intelligence and Analytics Software - SAS	SAS System includes tools for data warehousing, EIS, OLAP, rapid application development, and applied analysis as well as business solutions for applications such as financial consolidation.
+ analysis, data, statistical(39)	XMM-Newton Science Analysis System	XMM-Newton X-ray Observatory Science Operations Centre, the Universe of X-ray Astronomy ... Pages maintained by SAS librarian. ...
+ read, + file, data(43)	SAS: What You Need to Know to Write a SAS Program	This is a brief guide to the essentials you need to know to write a SAS program ... Reading a SAS Data Set: ... Saving a permanent SAS data set: ...
+ train, + exam, + certification(18)	SAS Customer Support Center	... want to find the answer to a nagging SAS problem, sign up for a training course ... SAS GLOBAL FORUM 2007, APR 16-19. Web ... Spotlight on SAS Enterprise Guide ...
air, regiment, british (30)	SAS, British Special Air Service, from the desert in `41 to	One of the elite fighting forces available to the Government for special operations.
+ window, unix, + run (45)	SAS System - Wikipedia, the free encyclopedia	The SAS System runs on IBM mainframes, Unix machines, OpenVMS Alpha, and ... SAS Add-In for Microsoft Office ... SAS procedures software analyzes and ...
+ airline, + pc, software(54)	sas	Covering the international airline's timetables, services, EuroBonus frequent flyer program, and more.
+ attach, + serial attach scsi, + analyzer(17)	Introduction to Serial Attached SCSI	SCSI, or SAS. ... The SAS ... an interface using a similar, compatible connector and cabling system, SAS could ...
radisson, + radisson sas hotel, + hotel(14)	Official Site of Radisson Hotels and Resorts	Provides reservations and events services. Includes information on specials and vacation packages.
society, astronomical, club (30)	UPenn - SAS Computing	School of Arts and Sciences Computing at the University of Pennsylvania ... SAS Computing is comprised of the following workgroups: ...
Other(170)	TS-DOC: TS-Y2KCOMP - The SAS System Year 2000 Compliance Inf	support.sas.com > Technical Support ... License Agreement with SAS Institute. ... section "Compliance Levels for Specific Components of the SAS System" ...

Figure 5. Web Results Page Generated from Cluster Results

CLUSTERING WITH A TOPIC FACTORIZATION

Up to this point in the paper, everything has been accomplished with the current production version of SAS Text Miner 3.1. In this section, we discuss new functionality that is planned in the next version, SAS Text Miner 4.1. One of the planned new features is a technique for discovering topics in a collection of documents. The approach is known as a “topic factorization” because it retrieves the topics by factorizing the term-by-document frequency data set that was created by the parsing. In this section, we explain how a matrix factorization can produce these topics and how these topics can be interpreted as clusters. We then use the topic factorization to solve the same Web result clustering problem presented earlier.

THE TOPIC FACTORIZATION

A topic factorization of a document collection is produced with a matrix factorization of a term-document frequency matrix, *A*. An example of a fictitious 8-term by 6-document matrix is shown on the left side of Figure 6. The two factors on the right side of the figure, L_3 and R_3, represent the factors that are produced. In the figure, *k* is set to 3 and is a value that is set by the user. The factors are found with algorithms that optimize the distance between the original matrix, *A*, and the product of the two factors.

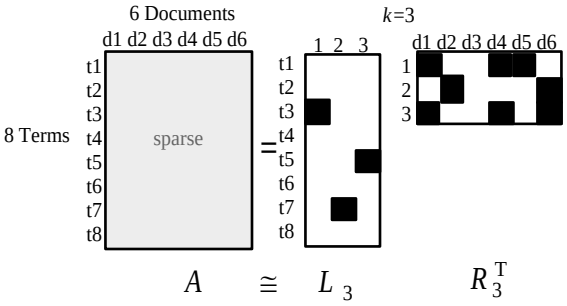


Figure 6. An Example of a Topic Factorization

TOPIC FACTORIZATION AS A CLUSTERING ALGORITHM

The artificial example in Figure 6 also demonstrates how a matrix factorization can both produce the clusters and label them. The factors generated on the right hand side are very sparse, and it is their nonzero patterns that serve to cluster the collection.

Notice that the dimensions of the two factors on the right side of Figure 6 correspond to the number of terms by *k* and the number of documents by *k*, respectively. The L_3 matrix shows the terms as observations mapped to 3 dimensions and the matrix R_3, the transpose of R^T_3 which is shown in Figure 6, gives the documents as observations mapped to 3 dimensions.

A dark box in the factors of Figure 6 indicates that a positive value exists in the cell. All white areas imply a zero value is in the cell. In L_3, each of the 3 dimensions has a single term assigned to it. The terms t3, t7, and t5 are assigned to dimensions1, 2, and 3, respectively, and each of these terms serves as a label for the corresponding topic. Each of the 3 dimensions can be described completely by its corresponding term. The factor R^T_3 reveals how the documents are assigned to the three topics based on the nonzero patterns there. The clusters are shown in Figure 7.

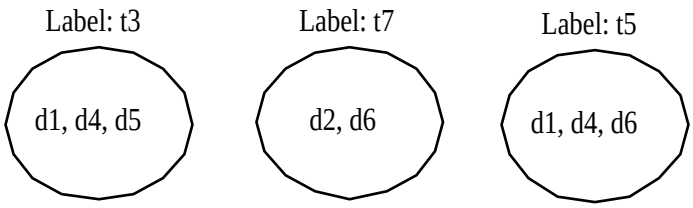


Figure 7. Three Clusters Formed by the Topic Factorization

Obviously, some documents belong to more than one cluster, but this is a strength of topic factorization because usually a document will contain more than one topic. In this case, not all the documents are clustered. Document d3 has no representation in this reduced space. When several documents have no intersection with the created clusters, they can form their own “Other” cluster. The one disadvantage to the topic factor technique is that, like most other clustering algorithms, the value for *k* must be determined by the user.

In general the solution to the problem of assigning documents to clusters is more complex than Figure 6 indicates because the factors produced are much larger and the zero patterns are not immediately obvious. When SAS Text Miner 4.1 is released, the user will be able to control how the software makes the decisions to assign the documents.

EVALUATING THE RESULTS

For our results, we chose the highest three terms in each column of the left matrix for the cluster labels, and we used a threshold on the right matrix to assign documents to clusters. A threshold value of 1.1 was chosen for the right factor so that approximately half the documents would be assigned to the “Other” cluster by default. Documents with a value greater than 1.1 in column i of the right matrix were assigned to cluster i. Documents that did not have a

value larger than the threshold were assigned to the “Other” cluster. Investigation on the optimal value for the threshold is a future research interest. Its selection is an important criterion in determining the best precision and recall outcomes. The results for the threshold value of 1.1 are given in Table 4.

Category	Precision	Recall
SAS Institute	.79	.55
Special Air Services	.55	.65
Radisson SAS Hotels	.84	1
Serial Attached SCSI	.94	.40
Scandinavian Air	0	0
Microaverage	**.76**	**.52**

Table 4. Precision and Recall for the Topic Factor Method

The clusters for the “sas” query are visualized in Figure 8. We used the constellation chart from SAS Enterprise Miner 6.1 to produce a plot of the clusters. The size of each node indicates the relative size of each cluster. Edges between clusters indicate that the clusters share many of the same documents. The width of the edges indicates the strength of that relationship. In this diagram, we included edges for any two clusters that shared at least two documents. Potentially, in real time, our solution could be used to process query results and display the results in a Web page containing this image. The Web page would allow users to navigate the results by clicking on the clusters.

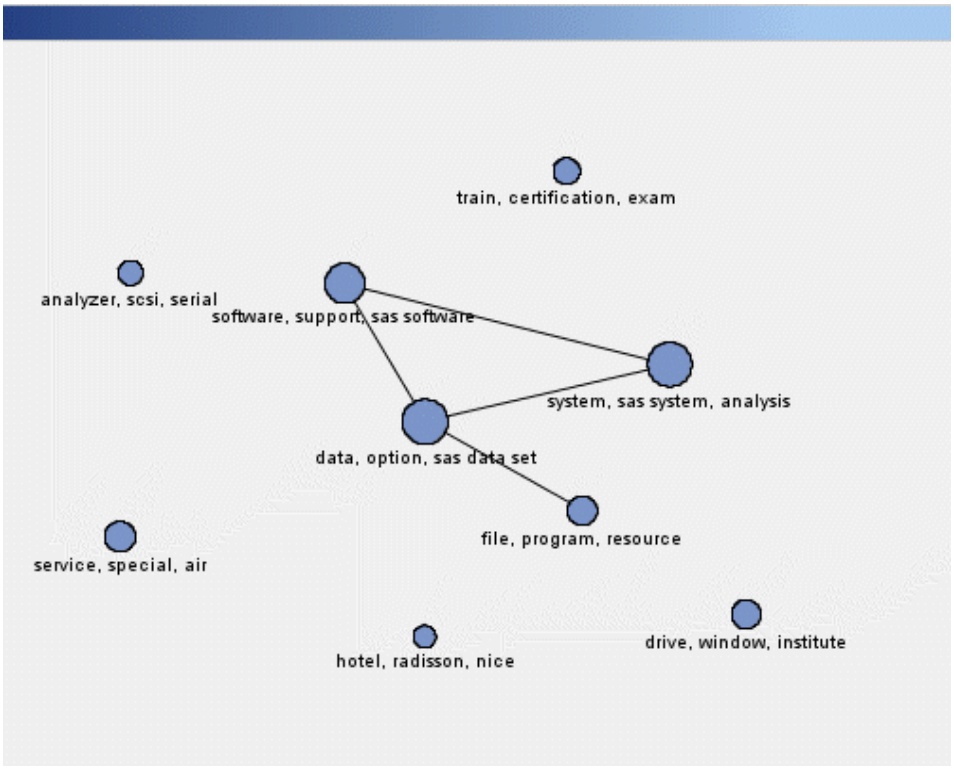


Figure 8. Constellation Chart for the Topic Factor Clusters

CONCLUSION

Although this paper focuses on analyzing Web search results, we hope that the techniques demonstrated here will be applicable for use in a broad spectrum of areas where users seek to understand the contents of large document collections. Web-based databases are becoming increasingly common and many have interfaces that are similar to the Yahoo! Search API that was used in this paper. You may not need to process Web results, but there may be a specialized database available to you that you do want to process.

These clustering techniques demonstrate how different approaches can affect your precision and recall and, depending on your needs, one of these criteria may be more important than the other. If labeling the clusters and providing only the first few related documents for each query is all that is required, then sacrificing recall for precision is acceptable. On the other hand, if you need high-quality groups for all the documents in the result set, then recall is important.

Our first two approaches based on the default clustering and on trimming the clusters and generating the HTML results to navigate the collection are available now to SAS Text Miner users. The approach shows how sometimes a custom solution can be derived from standard behavior. We used information about the clusters (the size and RMS. Std.) and information about the problem domain (the rank of each result) to improve the clusters that were derived.

In our last example, we compared the topic factor approach, which is soon to come in SAS Text Miner, with our clustering approach. The factorization method showed it has some strengths. The precision and recall scores were on par with the default and modified clustering methods. Further research into optimizing precision and recall for a given threshold should provide even more significant gains. The ability to assign documents to more than one topic is also a feature users have requested. Further, since the topic factor method both factors and clusters simultaneously, it is a very fast method for performing clustering.

None of the algorithms were able to capture the relatively large (18 results) cluster of Scandinavian Airline results that was evident in the collection to humans. After investigating the clustering results further, some of the results were understandably confused by the algorithm for the British Air Services cluster, some of the results were confused for Radisson SAS Hotels because of the travel relationships that exist between pages, and some were confused with SAS Institute because SAS has both Web pages indicating a presence in Scandinavia and Web pages touting Scandinavian Airline's use of SAS software!

One last noticeable direction for research is on improving the labels for the clusters. Both techniques in this paper relied on terms within the collection to label the clusters. While the individual terms provide some direction on the content of the cluster, they aren't able to really capture the nature of the cluster. We would like to pursue other ideas for labeling, such as using entities, noun groups, or other co-occurring terms or methods for allowing the user to input taxonomies or dictionaries to guide the labeling process.

REFERENCES

Albright, R. 2004. *Taming Text with the SVD.* Available at ftp://ftp.sas.com/techsup/download/EMiner/Taming Text with the SVD.pdf.

Chakrabarti, Soumen. 2002. *Mining the Web - Discovering Knowledge from Hypertext Data*. San Francisco, CA: Morgan Kaufmann – Elsevier.

Zamir, O., and Etzioni, O. 1998. "Web Document Clustering: a Feasibility Demonstration". *Proceedings of the 21st Annual international ACM SIGIR Conference on Research and Development in Information Retrieval* (Melbourne, Australia, August 24 - 28, 1998). SIGIR '98. New York, NY: ACM Press, 46-54.

ACKNOWLEDGMENTS

The authors thank Tonya Balan and James Cox for reviewing drafts of this paper and Dom Latour for his assistance with aspects of SAS Enterprise Miner coding.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Russ Albright
SAS Institute Inc.
russell.albright@sas.com