A CLUSTER-BASED APPROACH TO THESAURUS CONSTRUCTION

Carolyn J. Crouch
Department of Computer Science
Tulane University
New Orleans, LA 70118

The importance of a thesaurus in the successful operation of an information retrieval system is well recognized. Yet techniques which support the automatic generation of thesauri remain largely undiscovered. This paper describes one approach to the automatic generation of global thesauri, based on the discrimination value model of Salton, Yang, and Yu and on an appropriate clustering algorithm. This method has been implemented and applied to two document collections. Preliminary results indicate that this method, which produces improvements in retrieval performance in excess of 10 and 15 percent in the test collections, is viable and worthy of continued investigation.

INTRODUCTION

A major factor in the successful operation of any information retrieval system is the set of dictionaries available for use by the system. Of these dictionaries (e.g., thesauri, statistical and/or syntactic phrase dictionaries, term hierarchies, etc.), the dictionary having the greatest potential impact on system performance is undoubtedly the thesaurus. Although the benefits accruing from the use of a well constructed thesaurus in terms of increased system performance are well recognized, the methodology for automatically creating such a thesaurus remains unspecified. In fact, virtually all thesauri presently in use are idiosyncratic.

Thus a topic of considerable interest to researchers aiming to improve the overall performance of information retrieval systems is automatic thesaurus construction. This paper describes an approach to the automatic construction of a global thesaurus based on the discrimination value model of Salton, Yang, and Yu [SALTON75a] and on an appropriate clustering algorithm. The discrimination value model itself is based on the vector space model described below.

C 1988 ACM 0-89791-274-8 88 0600 0309 $ 1,50

The Vector Space Model

One of the major models in information retrieval is the vector space model. This model views each document in the document collection as a set of unique words or word types. Each document may then be regarded as a term vector, and the complete document collection becomes a vector space of dimension m, where m is the number of word types in the collection. In the vector space model, a document vector, $\mathbf{d_j}$, is represented by a set of terms, d_{jk}, $1 \leq k \leq m$, where d_{jk} represents the frequency (or weight) of term k in document j (i.e., the number of times term k appears in document j). If $d_{jk} = 0$, term k does not appear in document $\mathbf{d_j}$. Queries, like documents, are represented by weighted term vectors.

Given any two term vectors, the similarity between the vectors may be assumed to be inversely related to the angle between them. If the two vectors coincide, the angle between them is zero, and the vectors are identical. In two dimensions, the vector space may be represented by its envelope. The (normalized) vectors are then viewed as points in the vector space, and the distance between any two points is inversely related to the similarity of the corresponding document vectors. The smaller the distance between two points, the smaller the angle between the corresponding vectors, and the greater the similarity of the vectors in terms of the number of word types they have in common.

Salton *et al* [SALTON75a, SALTON75b, SALTON76] have shown that the best document space for retrieval purposes is one which maximizes the average separation between documents in the document space. In this space, it is easier to distinguish between documents and thus easier to retrieve documents which are most similar to a given query. The model which allows the terms in a collection to be ranked in order of their effect on space density is called the discrimination value model.

The Discrimination Value Model

The discrimination value model [SALTON75a] assigns specific roles to single terms, term phrases, and term classes for content analysis purposes and provides a framework within which each potential index term in a collection can be ranked in accordance with its usefulness as a document discriminator. It also offers a reasonable physical interpretation of the indexing process.

If we consider a collection of documents, each represented by a set of weighted m-dimensional vectors, then the similarity coefficient computed between any two term vectors can be interpreted as a measure of the closeness or relatedness between the vectors in m-space. If the similarity coefficient is large, the documents are very similar and appear in close proximity to each other in the document space. And if the similarity coefficient is small, the documents exhibit little similarity and are widely separated in the document space.

The discrimination value of a term is then defined as a measure of the change in space separation which occurs when a given term is assigned to the document collection. A *good discriminator* is a term which, when assigned to a document, *decreases the space density* (*i.e., renders the documents less similar to each* other). Conversely, the assignment of a *poor discriminator* increases the space density. By computing the density of the document space before and after the assignment of each term, the discrimination value of the term can be determined. The terms can then be ranked in decreasing order of their discrimination values.

Salton, Yang, and Yu [SALTON75a] have used discrimination value to determine three categories of discriminators, namely, good, poor, and indifferent discriminators. A term with a positive discrimination value has been found to be a good discriminator. Salton *et al* suggest that these terms be used directly as index terms. Those terms with negative discrimination values are poor discriminators; the retrieval properties of such terms can be transformed by including them in appropriate phrases. The majority of terms are indifferent discriminators with near-zero discrimination values. The retrieval capabilities of these terms can be enhanced by means of their incorporation in appropriate thesaurus classes.

Thus the discrimination value model presents a criterion for the formation of global thesauri. According to this model, a thesaurus is composed of a set of *thesaurus classes. A thesaurus class is composed of a group of terms or word* types. The terms within a class should be indifferent discriminators (i.e., those with near-zero discrimination values). Thus in order to use the criterion suggested by the discrimination model, the discrimination value of each term in the collection must be calculated and the terms ranked as good, indifferent and poor discriminators according to their discrimination values.

But the calculation of discrimination value is normally expensive. Two different approaches have been used. One approach, the so-called *exact* method, involves the calculation of all pairwise similarities between the document vectors of the collection. For a collection of n documents and m word types, the complexity of this algorithm is $O(mn^2)$. The second or *approximate* approach to the calculation of discrimination value involves the construction of an artificial, *average* document, the centroid, and computes the sum of the similarities of each document with the centroid. The centroid algorithm is O(mn).

Modifications have been suggested which improve the execution times associated with both the exact and the approximate methods of calculating discrimination value [CRAWFORD75, WILLET85, CROUCH88]. Although the discrimination values produced by these two approaches differ significantly for a particular collection, it has been shown that the *rankings* of the terms are in fact highly compatible [CROUCH88]. Thus of these two methods, the more efficient, centroid approach is the obvious method of choice when discrimination values must be calculated.

But for a document collection of any size, even the centroid approach to the calculation of discrimination value may be expensive. A reasonable alternative has been provided by Salton, Yang, and Yu, who suggest the use of document frequency as an approximation to discrimination value. For any term k in a document collection, the document frequency of term k, d_k, is defined as the number of documents in which term k appears. Empirical results indicate that document frequency and discrimination value are strongly correlated. Let n represent the number of documents in a collection whose terms are ranked by increasing document frequency. According to [SALTON75a], those terms whose document frequency is less than n/100 may be considered *low frequency* terms. The discrimination values of these terms are normally near-zero, and these terms as a whole may be considered indifferent discriminators. Likewise, terms whose document frequencies are greater than n/10 may be considered *high frequency* terms. These terms normally have negative discrimination values and are considered poor discriminators. The remaining terms (i.e., $n/10 \leq d_k \leq n/100$) make up the set of good discriminators. The discrimination values of these terms are positive.

Thus document frequency may be used as an approximation to discrimination value. Thesaurus classes, which theoretically should consist of groupings of terms with near-zero discrimination values, may instead be constructed of sets of low frequency terms. Since document frequency is readily available for every term in a collection, the cost associated with this approach is minimal.

AN APPROACH TO THESAURUS CONSTRUCTION

An experiment was designed to investigate the feasibility of constructing a global thesaurus based on low frequency terms. The term "global thesaurus" is used to differentiate this type of thesaurus from the "local thesaurus" described by Attar and Fraenkel [ATTAR77]. In a global approach, thesaurus classes, once constructed, are used to index both documents and queries. The local thesaurus, in contrast, uses information obtained from the documents retrieved in response to a particular query to modify that query, which is then resubmitted to the retrieval system for processing in lieu of the original. Thus a global thesaurus is constructed prior to the indexing process and the thesaurus classes are used to index both documents and queries, whereas a local thesaurus is constructed dynamically during query processing and uses information retrieved in response to a specific query to modify only that query.

Constructing Thesaurus Classes

Constructing a global thesaurus based on the discrimination value model calls for the generation of thesaurus classes consisting of indifferent discriminators or (as a viable alternative) low frequency terms. The question of how the classes themselves are to be constructed remains open. Intuitively, a thesaurus class should consist of terms which are *closely related* in the context of the current collection. (Such terms are not necessarily synonyms in the conventional sense.) One approach to generating groups of closely related terms is to cluster all the terms in the collection. The low frequency terms which cluster together might then be considered a thesaurus class. Unfortunately, in an environment where there is little information to exploit, the resultant clusters are seldom meaningful. This is the case in the low frequency domain, where terms are contained in only a small number of the documents in the collection.

An alternative approach is to cluster the documents of the collection and to generate thesaurus classes from the low frequency terms contained in the document clusters. A key question then arises, namely, what type of clustering algorithm should be used? The choice is dictated largely by the type of clusters an algorithm produces. In order to generate meaningful thesaurus classes, the low frequency terms in a class should come from closely related documents. This implies that the document clusters themselves should be small and tight. An algorithm which produces clusters of this type is the complete-link clustering algorithm, one of a class of agglomerative, hierarchical clustering algorithms that has received some attention in the literature [VANRIJSB79, VOORHEES85, VOORHEES86]. Consequently, this was the algorithm used to cluster the documents in our test collections.

Constructing a Global Thesaurus

The following procedure has been utilized to construct a global thesaurus:

1. The document collection is clustered via the complete-link algorithm.
2. The resultant hierarchy is traversed and thesaurus classes are generated, based on specified, user-supplied parameters.
3. The documents and queries are indexed by the thesaurus classes.

The characteristics of the thesaurus classes generated in step 2 are determined by the following, user-supplied parameters:

(a) *THRESHOLD VALUE*

Application of the complete-link clustering algorithm produces a hierarchy in which the tightest clusters (i.e., those which cluster at the highest threshold values) lie at the bottom of the cluster tree. These nodes are the leaves of the tree. For example, consider Fig. 1. The squares represent documents and the numbers in the circles represent the levels at which the documents cluster. Documents A and B cluster at a threshold value of 0.089, D and E cluster at a level of 0.149, and document C clusters with the D-E subtree at a level of 0.077. The A-B subtree and the C-D-E subtree cluster at a threshold value of 0.029.

The user-supplied threshold value largely determines the documents from which terms are selected for inclusion in a thesaurus class. In Fig. 1, a threshold value of 0.090 would return only the D-E document cluster, since

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**