TopCat: Data Mining for Topic Identification in a Text Corpus*

Chris Clifton[1] and Robert Cooley[2] **

[1] The MITRE Corporation, 202 Burlington Rd, Bedford, MA 01730-1420 USA
clifton@mitre.org
[2] University of Minnesota, 6-225D EE/CS Building, Minneapolis, MN 55455 USA
cooley@cs.umn.edu

Abstract. TopCat (Topic Categories) is a technique for identifying topics that recur in articles in a text corpus. Natural language processing techniques are used to identify key entities in individual articles, allowing us to represent an article as a set of items. This allows us to view the problem in a database/data mining context: Identifying related groups of items. This paper presents a novel method for identifying related items based on "traditional" data mining techniques. Frequent itemsets are generated from the groups of items, followed by clusters formed with a hypergraph partitioning scheme. We present an evaluation against a manually-categorized "ground truth" news corpus showing this technique is effective in identifying topics in collections of news articles.

1 Introduction

Data mining has emerged to address problems of understanding ever-growing volumes of information for structured data, finding patterns within the data that are used to develop useful knowledge. On-line textual data is also growing rapidly, creating needs for automated analysis. There has been some work in this area [14,10,16], focusing on tasks such as: association rules among items in text [9], rules from semi-structured documents [18], and understanding use of language [5,15]. In this paper the desired knowledge is major topics in a collection; data mining is used to discover patterns that disclose those topics.

The basic problem is as follows: Given a collection of documents, what topics are frequently discussed in the collection? The goal is to help a human understand the collection, so a good solution must identify topics in some manner that is meaningful to a human. In addition, we want results that can be used for further exploration. This gives a requirement that we be able to identify source texts relevant to a given topic. This is related to document clustering [21], but the requirement for a topic *identifier* brings it closer to rule discovery mechanisms.

The way we apply data mining technology on this problem is to treat a document as a "collection of entities", allowing us to map this into a *market*

* This work supported by the Community Management Staff's Massive Digital Data Systems Program.

** This work was performed while the author was at the MITRE Corporation.

J.M. Żytkow and J. Rauch (Eds.): PKDD'99, LNAI 1704, pp. 174–183, 1999.

basket problem. We use natural language technology to extract named entities from a document. We then look for *frequent itemsets*: groups of named entities that commonly occurred together. Next, we further cluster on the groups of named entities; capturing closely-related entities that may not actually occur in the same document. The result is a refined set of clusters. Each cluster is represented as a set of named entities and corresponds to an ongoing topic in the corpus. An example topic is: ORGANIZATION Justice Department, PERSON Janet Reno, ORGANIZATION Microsoft. This is recognizable as the U.S. antitrust case against Microsoft. Although not as informative as a narrative description of the topic, it is a compact, human-understandable representation. It also meets our "find the original documents" criteria, as the topic can used as a query to find documents containing some or all of the extracted named entities (see Section 3.4).

2 Problem Statement

The TopCat project started with a specific user need. The GeoNODE project at MITRE [12] is developing a system for analysis of news in a geographic context. One goal is to visualize ongoing topics in a geographic context; this requires *identifying* ongoing topics. We had experience with identifying association rules among entities/concepts in text, and noticed that *some* of the rules were recognizable as belonging to major news topics. This led to the effort to develop a topic identification mechanism based on data mining techniques.

There are related topic-based problems being addressed. The Topic Detection and Tracking (TDT) project [1] looks at clustering and classifying news articles. Our problem is similar to the Topic Detection (clustering) problem, except that we must generate a *human-understandable* "label" for a topic: a compact identifier that allows a person to quickly see what the topic is about. Even though our goals are slightly different, the test corpus developed for the TDT project (a collection of news articles manually classified into topics) provides a basis for us to evaluate our work. A full description of the corpus can be found in [1]. For this evaluation, we use the topic detection criteria developed for TDT2 (described in Section 4). This requires that we go beyond identifying topics, and also match documents to a topic.

One key item missing from the TDT2 evaluation criteria is that the $TopicID$ must be *useful to a human*. This is harder to evaluate, as not only is it subjective, but there are many notions of "useful". We later argue that the $TopicID$ produced by TopCat is useful to and understandable by a human.

3 Process

TopCat follows a multi-stage process, first identifying key concepts within a document, then grouping these to find topics, and finally mapping the topics back to documents and using the mapping to find higher-level groupings. We identify key concepts within a document by using natural language techniques to extract

named people, places, and organizations. This gives us a structure that can be mapped into a *market basket* style mining problem.[1] We then generate *frequent itemsets*, or groups of named entities that commonly appear together. Further clustering is done using a hypergraph splitting technique to identify groups of frequent itemsets that contain considerable overlap, even though not all of the items may appear together often enough to qualify as a frequent itemset.

The generated topics, a set of named entities, can be used as a query to find documents related to the topic (Section 3.4). Using this, we can identify topics that frequently occur in the same document to perform a further clustering step (identifying not only topics, but also topic/subtopic relationships).

We will use the following cluster, capturing professional tennis stories, as an example throughout this section.

PERSON Andre Agassi	PERSON Martina Hingis	PERSON Mary Pierce
PERSON Pete Sampras	PERSON Venus Williams	PERSON Serena
PERSON Marcelo Rios	PERSON Anna Kournikova	

This is a typical cluster (in terms of size, support, etc.) and allows us to illustrate many of the details of the TopCat process. It comes from merging two subsidiary clusters (described in Section 3.5), formed from clustering seven frequent itemsets (Section 3.3).

3.1 Data Preparation

TopCat starts by identifying *named entities* in each article (using the Alembic[7] system). This serves several purposes. First, it shrinks the data set for further processing. It also gives *structure* to the data, allowing us to treat documents as a set of typed and named entities. This gives us a natural database schema for documents that maps into the traditional market basket data mining problem. Third, and perhaps most important, it means that *from the start* we are working with data that is rich in meaning, improving our chances of getting human understandable results. We eliminate frequently occurring terms (those occurring in over 10% of the articles, such as United States), as these are used across too many topics to be useful in discriminating between topics.

We also face a problem with multiple names for the same entity (e.g., Marcelo Rios and Rios). We make use of *coreference* information from Alembic to identify different references to the same entity *within* a document. From the group of references for an entity within a document, we use the globally most common version of the name *where most groups containing that name contain at least one other name within the current group*. Although not perfect, this does give a *global identifier* for an entity that is both reasonably global and reasonably unique.

We eliminate composite articles (those about multiple unrelated topics, such as daily news summaries). We found most composite articles could be identified

[1] Treating a document as a "basket of words" did not produce as meaningful topics. Named entities stand alone, but raw words need sequence.

by periodic recurrence of the same headline; we ignore any article with a headline that occurs at least monthly.

3.2 Frequent Itemsets

The foundation of the topic identification process is *frequent itemsets*. In our case, a frequent itemset is a group of named entities that occur together in multiple articles. What this really gives us is correlated items, rather than any notion of a topic. However, we found that correlated named entities frequently occurred within a recognizable topic.

Discovery of frequent itemsets is a well-understood data mining problem, arising in the *market basket* association rule problem [4]. A document can be viewed as a market basket of named entities; existing research in this area applies directly to our problem. (We use the *query flocks* technology of [20] for finding frequent itemsets using the filtering criteria below). One problem with frequent itemsets is that the items must co-occur *frequently*, causing us to ignore topics that occur in only a few articles. To deal with this, we use a low support threshold of 0.05% (25 occurrences in the TDT corpus). Since we are working with multiple sources, any topic of importance is mentioned multiple times; this level of support captures all topics of any ongoing significance. However, this gives too many frequent itemsets (6028 2-itemsets in the TDT corpus). We need additional filtering criteria to get just the "important" itemsets.[2]

We use *interest*[6], a measure of correlation strength (specifically, the ratio of the probability of a frequent itemset occurring in a document to the multiple of the independent probabilities of occurrence of the individual items) as an additional filter. This emphasizes relatively rare items that generally occur together, and de-emphasizes common items. We select all frequent itemsets where either the support or interest are at least one standard deviation above the average, or where both support and interest are above average (note that this is computed independently for 2-itemsets, 3-itemsets, etc.) For 2-itemsets, this brings us from 6028 to 1033.

We also use interest to choose between "contained" and "containing" itemsets (i.e., any 3-itemset contains three 2-itemsets with the required support.) An $n-1$-itemset is used only if it has greater interest than the corresponding n-itemset, and an n-itemset is used only if it has greater interest than at least one of its contained $n-1$-itemsets. This brings us to 416 (instead of 1033) 2-itemsets.

The difficulty with using frequent itemsets for topic identification is that they tend to be over-specific. For example, the "tennis player" frequent itemsets consist of the following:

[2] The problems with traditional data mining measures for use with text corpuses have been noted elsewhere as well, see [8] for another approach.

Type1	*Value1*	*Type2*	*Value2*	*Support*	*Interest*
PERSON	Andre Agassi	PERSON	Marcelo Rios	.00063	261
PERSON	Andre Agassi	PERSON	Pete Sampras	.00100	190
PERSON	Anna Kournikova	PERSON	Martina Hingis	.00070	283
PERSON	Marcelo Rios	PERSON	Pete Sampras	.00076	265
PERSON	Martina Hingis	PERSON	Mary Pierce	.00057	227
PERSON	Martina Hingis	PERSON	Serena	.00054	228
PERSON	Martina Hingis	PERSON	Venus Williams	.00063	183

These capture individual matches of significance, but not the topic of "championship tennis" as a whole.

3.3 Clustering

We experimented with different frequent itemset filtering techniques, but were always faced with an unacceptable tradeoff between the number of itemsets and our ability to capture a reasonable breadth of topics. Further investigation showed that some named entities we should group as a topic would not show up as a frequent itemset under *any* measure; no article contained **all** of the entities. Therefore, we chose to perform clustering of the named entities in addition to the discovery of frequent itemsets. The *hypergraph clustering* method of [11] takes a set of association rules and declares the items in the rules to be vertices, and the rules themselves to be hyperedges. Clusters can be quickly found by using a hypergraph partitioning algorithm such as hMETIS [13].

We adapted the hypergraph clustering algorithm described in [11] in several ways to fit our particular domain. Because TopCat discovers frequent itemsets instead of association rules, the rules do not have any directionality and therefore do not need to be combined prior to being used in a hypergraph. The interest of each itemset was used for the weight of each edge. Since interest tends to increase dramatically as the number of items in a frequent itemset increases, the log of the interest was used in the clustering algorithm to prevent the larger itemsets from completely dominating the process.

Upon investigation, we found that the stopping criteria presented in [11] only works for domains that form very highly connected hypergraphs. Their algorithm continues to recursively partition a hypergraph until the weight of the edges cut compared to the weight of the edges left in either partition falls below a set ratio (referred to as *fitness*). This criteria has two fundamental problems: it will never divide a loosely connected hypergraph into the appropriate number of clusters, as it stops *as soon as* if finds a partition that meets the fitness criteria; and it always performs at least one partition (even if the entire hypergraph should be left together.)

To solve these problems, we use the *cut-weight* ratio (the weight of the cut edges divided by the weight of the uncut edges in a given partition). This is defined as follows. Let P be a partition with a set of m edges e, and c the set of n edges cut in the previous split of the hypergraph:

$$cutweight(P) = \frac{\Sigma_{i=1}^{n} Weight(c_i)}{\Sigma_{j=1}^{m} Weight(e_j)}$$

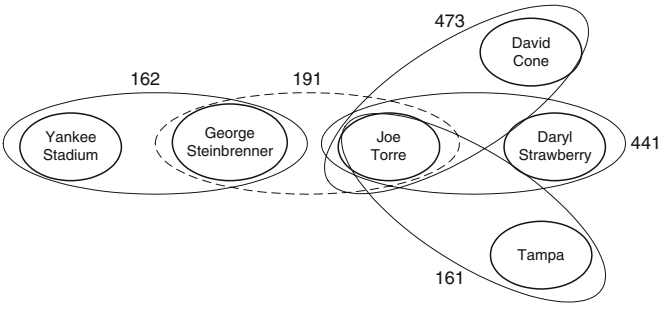

Fig. 1. Hypergraph of New York Yankees Baseball Frequent Itemsets

A hyperedge remains in a partition if 2 or more vertices from the original edge are in the partition. For example, a cut-weight ratio of 0.5 means that the weight of the cut edges is half of the weight of the remaining edges. The algorithm assumes that natural clusters will be highly connected by edges. Therefore, a low cut-weight ratio indicates that hMETIS made what should be a natural split between the vertices in the hypergraph. A high cut-weight ratio indicates that the hypergraph was a natural cluster of items and should not have been split. Once the stopping criteria has been reached, vertices are "added back in" to clusters if they are contained in an edge that "overlaps" to a significant degree with the vertices in the cluster. The minimum amount of overlap required is defined by the user. This allows items to appear in multiple clusters.

For our domain, we found that the results were fairly insensitive to the cutoff criteria. Cut-weight ratios from 0.3 to 0.8 produced similar clusters, with the higher ratios partitioning the data into a few more clusters than the lower ratios.

The TDT data produced one huge hypergraph containing half the clusters. Most of the rest are independent hypergraphs that become single clusters. One that does not become a single cluster is shown in Figure 1. Here, the link between Joe Torre and George Steinbrenner (shown dashed) is cut. Even though this is not the weakest link, the attempt to balance the graphs causes this link to be cut, rather than producing a singleton set by cutting a weaker link. This is a sensible distinction. During spring 1999, the Yankees manager (Torre) and players were in Tampa, Florida for spring training, while the owner (Steinbrenner) was handling repairs to a crumbling Yankee Stadium in New York.

3.4 Mapping to Documents

The preceding process gives us reasonable topics. However, to evaluate this with respect to the TDT2 instrumented corpus, we must map the identified topics back to a set of documents. We use the fact that the topic itself, a set of named entities, looks much like a boolean query. We use the TFIDF metric[17] to

generate a distance measure between a document and a topic, then choose the closest topic for each document. This is a flexible measure; if desired, we can use cutoffs (a document isn't close to any topic), or allow multiple mappings.

3.5 Combining Clusters Based on Document Mapping

Although the clustered topics appeared reasonable, we were over-segmenting with respect to the TDT "ground truth" criteria. For example, we separated men's and women's tennis; the TDT human-defined topics had this as a single topic.

We found that the topic-to-document mapping provided a means to deal with this. Many documents were close to multiple topics. In some cases, this overlap was common and repeated; many documents referenced both topics (the tennis example was one of these). We used this to merge topics, giving the final "tennis" topic shown in Section 1.

There are two types of merge. In the first (*marriage*), the majority of documents similar to either topic are similar to both. In the second (*parent/child*), the documents similar to the child are also similar to the parent, but the reverse does not necessarily hold. (The tennis clusters were a *marriage* merge.)

The marriage similarity between clusters a and b is defined as:

$$Marriage_{ab} = \frac{\sum_{i \in documents} TFIDF_{ia} * TFIDF_{ib}/N}{\sum_{i \in documents} TFIDF_{ia}/N * \sum_{i \in documents} TFIDF_{ib}/N}$$

Based on the TDT2 training set, we chose a cutoff of 30 ($Marriage_{ab} \geq 30$) for merging clusters. Similar clusters are merged by taking a union of their named entities.

The parent child relationship is calculated as follows:

$$ParentChild_{pc} = \frac{\sum_{i \in documents} TFIDF_{ip} * TFIDF_{ic}/N}{\sum_{i \in documents} TFIDF_{ic}/N}$$

We calculate the parent/child relationship after the marriage clusters have been merged. In this case, we used a cutoff of 0.3. Merging the groups is again accomplished through a union of the named entities.

Note that there is nothing **document**-specific about these methods. The same approach could be applied to any market basket problem.

4 Experimental Results

The TDT2 evaluation criteria is based on the probability of failing to retrieve a document that belongs with the topic, and the probability of erroneously matching a document to the topic. These are combined to a single number C_{Det} as describe in [3]. The mapping between TopCat-identified topics and reference topics is defined to be the mapping that minimizes C_{Det} for that topic (as specified by the TDT2 evaluation process).

Using the TDT2 evaluation data (May and June 1998), the C_{Det} score was 0.0055. This was comparable to the results from the TDT2 topic detection participants[2], which ranged from 0.0040 to 0.0129, although they are not directly comparable (as the TDT2 topic detection is on-line, rather than retrospective). Of note is the low false alarm probability we achieved (0.002); further improvement here would be difficult. The primary impediment to a better overall score is the miss probability of 0.17.

The primary reason for the high miss probability is the difference in specificity between the human-defined topics and the TopCat-discovered topics. (Only two topics were missed entirely; one contained a single document, the other three documents.) Many TDT2-defined topics matched multiple TopCat topics. Since the TDT2 evaluation process only allows a single system-defined topic to be mapped to the human-defined topic, over half the TopCat-discovered topics were not used (and any document associated with those topics was counted as a "miss" in the scoring). TopCat often identified separate topics, such as (for the conflict with Iraq) Madeleine Albright/Iraq/Middle East/State, in addition to the "best" topic (lowest C_{Det} score) shown at the top of Table 1. Although various TopCat parameters could be changed to merge these, many similar topics that the "ground truth" set considers separate (such as the world ice skating championships and the winter Olympics) would be merged as well.

The miss probability is a minor issue for our problem. Our goal is to *identify important topics*, and to give a user the means to follow up on that topic. The low false alarm probability means that a story selected for follow-up *will* give good information on the topic. For the purpose of understanding general topics and trends in a corpus, it is more important to get all topics and a few good articles for each topic than to get all articles for a topic.

5 Conclusions and Future Work

We find the identified topics both reasonable in terms of the TDT2 defined accuracy, and understandable identifiers for the subject. For example, the most important three topics (based on the support of the frequent itemsets used to generate the topics) are shown in Table 1. The first (Iraqi arms inspections) also gives information on who is involved (although knowing that Richard Butler was head of the arms inspection team, Bill Richardson is the U.S. Ambassador to the UN, and Saddam Hussein is the leader of Iraq may require looking at the documents; this shows the usefulness of mapping the topic identifier to documents.) The third is also reasonably understandable: Events in and around Yugoslavia. The second is an amusing proof of the first half of the adage "Everybody talks about the weather, but nobody does anything about it."

The clustering methods of TopCat are not limited to topics in text, any market basket style problem is amenable to the same approach. For example, we could use the hypergraph clustering and relationship clustering on mail-order purchase data. This extends association rules to higher-level "related purchase" groups. Association rules provide a few highly-specific *actionable items*, but are

Table 1. Top 3 Topics for January through June 1998

Topic 1		*Topic 2*		*Topic 3*	
LOCATION	Baghdad	LOCATION	Alaska	LOCATION	Albania
LOCATION	Britain	LOCATION	Anchorage	LOCATION	Macedonia
LOCATION	China	LOCATION	Caribbean	LOCATION	Belgrade
LOCATION	Iraq	LOCATION	Great Lakes	LOCATION	Bosnia
ORG.	Security Council	LOCATION	Gulf Coast	LOCATION	Pristina
ORG.	United Nations	LOCATION	Hawaii	LOCATION	Yugoslavia
PERSON	Kofi Annan	LOCATION	New England	LOCATION	Serbia
PERSON	Saddam Hussein	LOCATION	Northeast	PERSON	Slobodan Milosevic
PERSON	Richard Butler	LOCATION	Northwest	PERSON	Ibrahim Rugova
PERSON	Bill Richardson	LOCATION	Ohio Valley	ORG.	Nato
LOCATION	Russia	LOCATION	Pacific Northwest	ORG.	Kosovo Liberation
LOCATION	Kuwait	LOCATION	Plains		Army
LOCATION	France	LOCATION	Southeast		
ORG.	U.N.	LOCATION	West		
		PERSON	Byron Miranda		
		PERSON	Karen Mcginnis		
		PERSON	Meteorologist Dave Hennen		
		PERSON	Valerie Voss		

not as useful for high-level understanding of general patterns. The methods presented here can be used to give an overview of patterns and trends of related purchases, to use (for example) in assembling a targeted specialty catalog.

The cluster merging of Section 3.5 defines a topic relationship. We are exploring how this can be used to browse news sources by topic. Another issue is the use of information other than named entities to identify topics. One possibility is to add *actions* (e.g., particularly meaningful verbs such as "elected").

We have made little use of the type of named entity. However, what the named entity processing really gives us is a *typed* market basket (e.g., LOCATION or PERSON as types.) Another possibility is to use generalizations (e.g., a geographic "thesaurus" equating Prague and Brno with the Czech Republic) in the mining process[19]. Further work on expanded models for data mining could have significant impact on data mining of text.

References

1. 1998 topic detection and tracking project (TDT-2). http://www.nist.gov/speech/tdt98/tdt98.htm.
2. The topic detection and tracking phase 2 (TDT2) evaluation. ftp://jaguar.ncsl.nist.gov/tdt98/tdt2_dec98_official_results_19990204/index.htm.
3. The topic detection and tracking phase 2 (TDT2) evaluation plan. http://www.nist.gov/speech/tdt98/doc/tdt2.eval.plan.98.v3.7.pdf.
4. Rakesh Agrawal, Tomasz Imielinski, and Arun N. Swami. Mining association rules between sets of items in large databases. In Peter Buneman and Sushil Jajodia, editors, *Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, pages 207–216, Washington, D.C., May 26–28 1993.
5. Helena Ahonen, Oskari Heinonen, Mika Klemettinen, and Inkeri Verkamo. Mining in the phrasal frontier. In *1st European Symposium on Principles of Data Mining and Knowledge Discovery (PKDD'97)*, Trondheim, Norway, June 25–27 1997.

6. Sergey Brin, Rajeev Motwani, and Craig Silverstein. Beyond market baskets: Generalizing association rules to correlations. In *Proceedings of the 1997 ACM SIGMOD Conference on Management of Data*, Tucson, AZ, May 13-15 1997.
7. David Day, John Aberdeen, Lynette Hirschman, Robyn Kozierok, Patricia Robinson, and Marc Vilain. Mixed initiative development of language processing systems. In *Proceedings of the Fifth Conference on Applied Natural Language Processing*, Washington, D.C., March 1997.
8. Ronen Feldman, Yonatan Aumann, Amihood Amir, Amir Zilberstein, and Wiolli Kloesgen. Maximal association rules: a new tool for mining for keyword co-occurrences in document collections. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pages 167–170, August 14–17 1997.
9. Ronen Feldman and Haym Hirsh. Exploiting background information in knowledge discovery from text. *Journal of Intelligent Information Systems*, 9(1):83–97, July 1998.
10. Ronen Feldman and Haym Hirsh, editors. *IJCAI'99 Workshop on Text Mining*, Stockholm, Sweden, August 2 1999.
11. Eui-Hong (Sam) Han, George Karypis, and Vipin Kumar. Clustering based on association rule hypergraphs. In *Proceedings of the SIGMOD'97 Workshop on Research Issues in Data Mining and Knowledge Discovery*. ACM, 1997.
12. Rob Hyland, Chris Clifton, and Rod Holland. GeoNODE: Visualizing news in geospatial context. In *Proceedings of the Federal Data Mining Symposium and Exposition '99*, Washington, D.C., March 9-10 1999. AFCEA.
13. George Karypis, Rajat Aggarwal, Vipin Kumar, and Shashi Shekar. Multilevel hypergraph partitioning: Applications in VLSI domain. In *Proceedings of the ACM/IEEE Design Automation Conference*, 1997.
14. Yves Kodratoff, editor. *European Conference on Machine Learning Workshop on Text Mining*, Chemnitz, Germany, April 1998.
15. Brian Lent, Rakesh Agrawal, and Ramakrishnan Srikant. Discovering trends in text databases. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pages 227–230, August 14–17 1997.
16. Dunja Mladenić and Marko Grobelnik, editors. *ICML-99 Workshop on Machine Learning in Text Data Analysis*, Bled, Slovenia, June 30 1999.
17. Gerard Salton, James Allan, and Chris Buckley. Automatic structuring and retrieval of large text files. *Communications of the ACM*, 37(2):97–108, February 1994.
18. Lisa Singh, Peter Scheuermann, and Bin Chen. Generating association rules from semi-structured documents using an extended concept hierarchy. In *Proceedings of the Sixth International Conference on Information and Knowledge Management*, Las Vegas, Nevada, November 1997.
19. Ramakrishnan Srikant and Rakesh Agrawal. Mining generalized association rules. In *Proceedings of the 21st International Conference on Very Large Databases*, Zurich, Switzerland, September 23-25 1995.
20. Dick Tsur, Jeffrey D. Ullman, Serge Abiteboul, Chris Clifton, Rajeev Motwani, Svetlozar Nestorov, and Arnon Rosenthal. Query flocks: A generalization of association rule mining. In *Proceedings of the 1998 ACM SIGMOD Conference on Management of Data*, pages 1–12, Seattle, WA, June 2-4 1998.
21. Oren Zamir, Oren Etzioni, Omid Madan, and Richard M. Karp. Fast and intuitive clustering of web documents. In *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, pages 287–290, August 14–17 1997.