Using Statistical Properties of Text to Create Metadata

Grace Crowder * — *crowder@cs.umbc.edu*
Charles Nicholas — *nicholas@cs.umbc.edu*

Computer Science and Electrical Engineering Department
University of Maryland Baltimore County
Baltimore, MD 21228-5398
March 26, 1996

1 Problem Statement

Given a large, dynamic, distributed text corpus, traditional Information Retrieval (IR) techniques do not handle the growth in data volume, user base, or data diversity, see Bowman [1]. If data is distributed, then a description of data residing at each location needs to be maintained. Such data would be similar to the data used for traditional IR searches of memory prior to searching secondary storage, e.g. an index as suggested by Frakes [6]. However, the amount of memory needed to hold an inverted file index and the time required to compute results for a query become unmanageable as the corpus grows into the gigabytes, see Witten [13].

2 Proposed Solution

We propose a solution making use of a mediated agent architecture, in which metadata is used to direct queries to appropriate agents. A mediated agent

*This work was supported in part by the United States Department of Defense.

architecture consists of application agents managing local corpora and brokers using metadata to direct searches. In this architecture, metadata must be automatically generated; effective for query, search, browse, and retrieval of the data; and much smaller than the data.

In our approach, metadata is the key to scalability. Metadata is used to characterize the data for several purposes, including query processing, browsing, and retrieval. Metadata may take different forms. We require that metadata have the following properties:

- Effective - if the metadata says the information is [not] relevant, it's probably [not] relevant
- Concise - much smaller than the data it describes
- Generated automatically - little or no human intervention required
- Abstractable - can create metadata of metadata
- Interchangeable - the form must allow queries, documents, and metadata to be compared.

3 CAFE

We call our mediated architecture CAFE for Cooperating Agents Find Everything. In CAFE there are application agents and brokers. Application agents communicate with other entities in the system using KQML (Knowledge Query and Manipulation Language) as described in Mayfield [5]. In addition, each application agent is responsible for its own data space, which means it will provide access to that data by generating and registering metadata as well as performing more traditional IR tasks: handling queries, corpus updates, etc.

Brokers are similar to application agents. However, the data they manage is metadata, supplied to them by application agents and other brokers. Brokers are not necessarily organized in a strict hierarchy and sibling communication is allowed.

4 Centroids as Metadata

As first introduced in Shannon [12], an n-gram is a sequence of n characters. Example 5-grams taken from this sentence: examp, xampl, ample. Pearce [8, 9], Church [2], and Damashek [4] have shown that IR systems based on n-grams as terms have desirable properties of language independence, garble resistance, and effective retrieval.

A frequency distribution of the n-grams in a document is called its n-gram profile. Pearce and Nicholas [9] showed that a document's n-gram profile serves to characterize its content. The mean of the collected n-gram profiles of a set of documents is the centroid. Note that the centroid document isn't a real document. The whole corpus has a centroid as does any subset of member documents.

Using centroids as metadata is appealing because, through our work with Telltale, an n-gram based IR system supplied by our sponsor, we've seen that n-grams satisfy the five properties mentioned above:

- Effective - as demonstrated through experiments with Telltale.
- Concise - Example: 5-gram centroid for a 40 megabyte *Wall Street Journal* corpus requires 6.5 megabytes.
- Generated automatically - a single pass through a set of documents suffices to calculate the centroid.
- Abstractable - a collection of centroids has a centroid in its own right.
- Interchangeable - n-gram profiles and centroids can be calculated for queries, documents, and corpora in the same manner.

Our experiments show that the size of a centroid depends on the choice of n: as n grows, so does the number of unique n-grams in the centroid. Figure 1 shows that as n grows from 3 to 8 the number of unique n-grams in a 40 megabyte *Wall Street Journal* corpus grows from 10,040 to 3,495,627.

Furthermore, for each value of n, as a corpus grows, the number of unique n-grams in the corpus grows and appears to reach a plateau, as shown in Figure 1.

Once metadata has been generated, it is to be used in CAFE, by a broker running Telltale to manage metadata, to direct queries to appropriate agents.

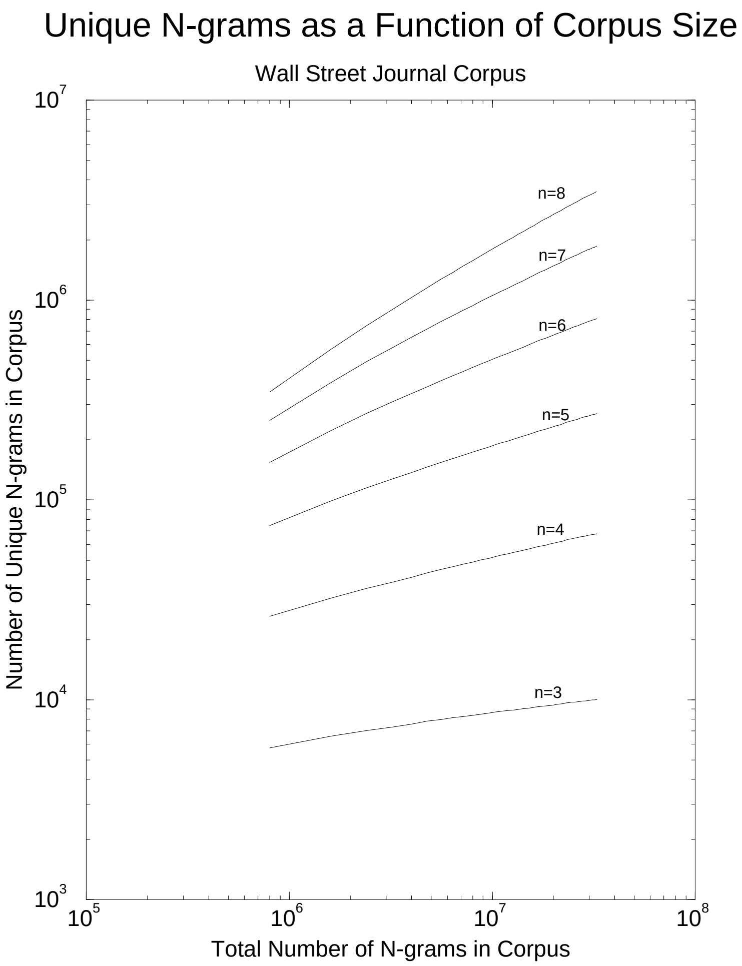

Figure 1: Number of Unique N-grams Reaches a Plateau as Corpus Grows

To assess the effectiveness of such retrieval, we conducted experiments to compare the effectiveness of Telltale against SMART.

SMART is an information retrieval system based on the vector space model using words as terms. SMART was developed at Cornell under the direction of Gerard Salton [10]. SMART is a language-dependent IR system in that it performs stop-listing and stemming on input. SMART is a successful participant in the Text Retrieval Conference (TREC) managed by Harman [7] at NIST.

Scored collections are used to compare IR systems. A scored collection consists of a set of documents, a set of queries, and relevance judgements for the queries. In a scored collection every document in the corpus has been judged as either relevant or not relevant for every query. There are several

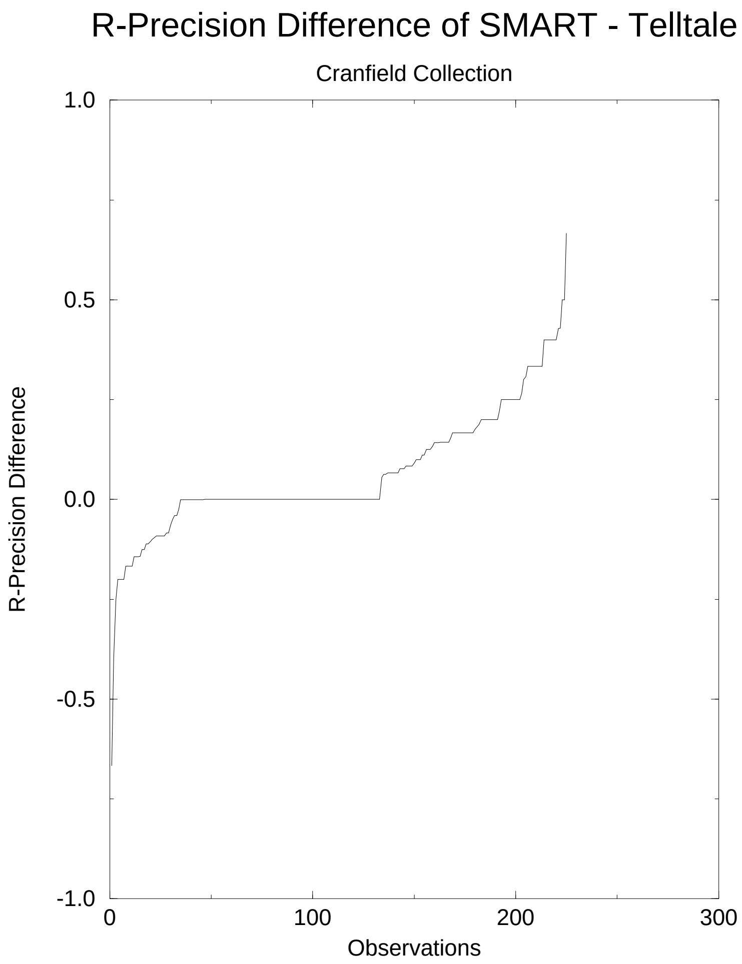

Figure 2: R-Precision Difference of SMART and Telltale

scored collections readily available from Cornell.

We have used the Cranfield collection of 1400 documents and 225 queries on aerodynamics to compare Telltale to SMART. Precision and recall measures are easily computed when working with a scored collection because relevance judgements (supplied by human experts) exist for every document with respect to every query. R-precision is a metric that can be used to compare two information retrieval engines. It is defined as the precision value for the j-th document returned for a query that has j documents in its relevant set. In a perfect system, the r-precision value would be 1.0 meaning all relevant documents and no irrelevant documents had been found. In our comparison, we found the r-precision difference of SMART and Telltale averages 0.06 in a possible range of -1.0 to +1.0. See Figure 2. This lets us conclude that for the Cranfield collection, Telltale does not perform significantly worse than SMART performs.

5 Conclusion

In conclusion, CAFE is a large-scale document information retrieval system that is:

- able to handle large corpora,
- dynamic: application agents can update their corpora and associated metadata,
- scalable: the choice of n-gram centroids for metadata further removes the traditional IR distinctions between metadata and corpus and document and query.

We plan to continue comparing SMART and Telltale using other scored collections and further studying the characteristics of the queries and result sets where SMART and Telltale performed differentyl

We have implemented user interface agents and back-end server agents. A broker that uses metadata to route queries is nearing completion. We are also investigating even more concise forms of metadata based on centroids and the impact (if any) on scalability and retrieval effectiveness.

References

[1] C. Mic Bowman, Peter Danzig, Udi Manber, and Michael Schwartz. Scalable Internet Resource Discovery: Research Problems and Approaches. *Communications of the ACM*, Vol. 37, no. 8, pp. 98-107, 114(Aug 1994).

[2] Kenneth Ward Church. *Ngrams* Talk given at UMBC, Fall 1995.

[3] Grace Crowder and Charles Nicholas. An Approach to Large Scale Distributed Information Systems Using Statistical Properties of Text to Guide Agent Search. *CIKM Workshop on Intelligent Information Agents*, 1995.

[4] Marc Damashek. Gauging Similarity with N-Grams: Language-Independent Categorization of Text. *Science*, Vol. 267, pp. 843-848, 10 February 1995.

[5] James Mayfield, Yannis Labrou, and Tim Finin. Evaluating KQML as an Agent Communication Language. in Michael Wooldridge, Joerg P. Mueller, and Milind Tambe(Ed.), *Intelligent Agents II: Agent Theories, Architectures, and Languages*, Springer-Verlag Lecture Notes in AI - Volume 1037, 1996.

[6] William B. Frakes and Ricardo Baeza-Yates. Information Retrieval: Data Structures and Algorithms. Prentice-Hall, 1992.

[7] Donna Harman. The Third Text REtrieval Conference. NIST Special Publication, PB95-216883, 1995.

[8] Claudia Pearce. A Dynamic Hypertext Environment Through N-gram Analysis. Ph. D. thesis, University of Maryland Baltimore County, 1994.

[9] Claudia Pearce and Charles Nicholas. TELLTALE: Experiments in a Dynamic Hypertext Environment for Degraded and Multilingual Data. *Journal of the American Society for Informaiton Science (JASIS)*, April 1996.

[10] Gerard Salton and Michael J. McGill. Introduction to Modern Information Retrieval. McGraw-Hill, 1983.

[11] Gerard Salton. Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer. Addison-Wesley, 1989.

[12] Claude Shannon. A Mathematical Theory of Communication. *The Bell System Technical Journal*, Vol 27, pp. 379-423, 623-656, July, October 1948.

[13] Ian H. Witten, Alistair Moffat, Timothy C. Bell. *Managing Gigabytes: Compressing and Indexing Documents and Images*. Van Nostrand Reinhold, New York, 1994.