INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN
ICED 99 MUNICH, AUGUST 24-26, 1999

MACHINE GENERATION OF THESAURI: ADAPTING TO EVOLVING VOCABULARIES IN DESIGN DOCUMENTATION

M. C. Yang and M. R. Cutkosky

Keywords: Design information retrieval, electronic design notebook, information capture

1 Introduction

A new breed of engineering design tools are electronic design notebooks, which are electronic versions of the traditional engineer’s logbook. They capture design information as it is generated, providing a rich, unfiltered history of a design project. This presents great potential for accessing past design decisions and rationale. This paper examines ways of searching for design information by enhancing existing information retrieval techniques with design-specific thesauri. Previous work examined the generation of thesauri both manually and automatically. While manual methods are still superior to machine methods for creating thesauri, it was found that thesauri extracted from blocks of text performed better than thesauri generated from whole documents. This paper takes up from that point, examining the promise of drawing out a viable design thesaurus from smaller blocks of text to aid in design search.

2 Background: Design information and information retrieval

Design information retrieval incorporates elements from both design information research and the field of information retrieval. The information captured in electronic design notebooks is informal and unstructured. Research in informal design information includes case-based design studies [7], as well as research in decision making that centers on lightly structuring informal information. Other work focuses on the detailed analysis of design process and information. One such study, known as Dedal [1], serves as part of the basis for this paper.

The second area that contextualizes this work is information retrieval. Examples of information retrieval systems include the Web search engines Hotbot and Excite. There is a great deal of new research in areas such as information filtering, information extraction, and data mining. Data mining deals with discovering patterns in data that can yield rules, classifications, or cluster the data in meaningful groups. Part of what this paper looks at is finding patterns of design terms that appear together in text in order to generate design thesauri.

3 Methods

3.1 The Dedal Framework

The framework for indexing and retrieval in this paper is Dedal [2] which grew out of protocol studies of designers. Through analysis of designer’s questions, certain patterns were detected. The questions were broken down into two parts: a *subject* and a *descriptor*. The subject is a part of the artifact being designed ("solenoid"), and the descriptor is an aspect of the subject, such as its performance ("limits on force and temperature"). For example, "What are the limits on force and temperature of this solenoid?" would be expressed as the subject <solenoid> and the descriptor <performance>. An example showing a car bumper is shown in Figure 1.

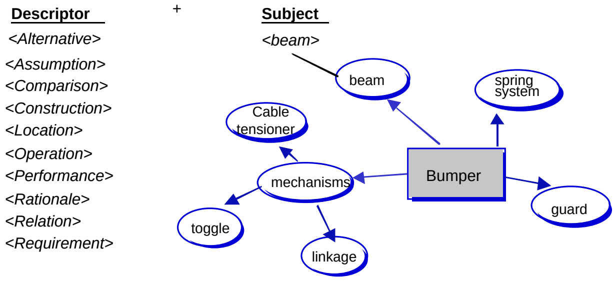

Figure 1. Dedal framework for composition of design questions, including descriptor and subject

3.2 Dedal performance and trade-offs

Before describing the results of Dedal, performance measures in information retrieval must first be defined. Information retrieval is measured in *precision* and *recall* (equations 1, 2). Perfect precision and recall is 100%, but that is rare in real world systems.

$$\text{Precision} = \frac{\text{Relevant retrieved}}{\text{Total retrieved}} \tag{1}$$

$$\text{Recall} = \frac{\text{Relevant retrieved}}{\text{Total relevant in collection}} \tag{2}$$

[2] demonstrated that Dedal could improve the precision of design information retrieval in final project documentation by approximately 70% over subject-only searches. Clearly, the Dedal framework has something to offer in terms of indexing design information.

The caveat of Dedal is its high cost, in terms of usability and maintenance. First, each descriptor and subject in a document must be indexed by hand, which requires both time and knowledge of a project. Second, hand indexed systems are difficult to keep up to date. Maintenance is critical to rapidly changing collections, such as design documents.

3.3 Test collection: Electronic design notebooks

The documents studied are electronic design notebooks generated from a graduate design course at Stanford University called ME210: Team Based Design Development with Corporate Partners. Teams of 3 to 4 graduate students work on an industry-sponsored project for over two quarters. They are encouraged to document work in PENS (Personal Electronic Notebook with Sharing) [4], an Internet-based electronic design notebook. PENS entries are amassed collectively over time, gradually building up the team’s design space. By the end of the project, a broad, unstructured record of the design process is available.

For each of these experiments, the same set of queries was used: nine queries for two notebooks - two sets of three queries each for <*alternative*>, <*construction*>, and <*performance*> Dedal descriptors. The set of relevant documents for each of the 18 queries was determined through an exhaustive search of each notebook.

3.4 Information retrieval methods

The experiments presented in this paper used the SMART information retrieval system [5] as a substrate for testing different strategies. SMART uses the same underlying principles as many search engines to index and retrieve information, called *term frequency-inverse document frequency* (tf-idf) indexing.

In tf-idf, documents in a collection are represented as vectors. The vectors are combined to form a term-document corpus matrix that represents the entire collection. Figure 2 shows an example corpus matrix for a collection of mechanical engineering documents. On the left side are titles of *documents* in a collection, and along the top are *terms* that appear in at least one of the documents in the collection.

$$\begin{array}{r|cccccc|}
 & \text{ÒbeamÓ} & \text{ÒlinkageÓ} & \text{ÒprototypeÓ} & . & . & \text{ÒassemblyÓ} \\
\textit{5/12 Meeting Notes} & c_{11} & c_{12} & c_{13} & . & . & c_{1j} \\
\textit{Executive Summary} & c_{21} & c_{22} & c_{23} & . & . & . \\
\textit{Abstract} & c_{31} & c_{32} & c_{33} & . & . & . \\
. & . & . & . & . & . & . \\
. & . & . & . & . & . & . \\
\textit{NSF report} & c_{i1} & c_{i2} & c_{i3} & . & . & c_{ij}
\end{array}$$

Figure 2. Corpus matrix for term frequency-inverse document frequency indexing

The matrix values c_{ij} (equation 3) are weightings that relate the importance of terms in each document.

$$c_{ij} = \log\left(\frac{1+ tf_{ij}}{df_j}\right) \qquad (3)$$

where *i* is the document index, *j* is the term index, c_{ij} is the weight of term *j* in document *i*, tf_{ij} is the term frequency (number of times term *j* appears in document *i*), and df_j is the document frequency (number of documents term *j* appears in). To retrieve documents, a user poses a query, which is represented by another term-document vector **q**. The cosine *similarity* **s** of **q** to the corpus **C** is determined (equation 4) and the most similar documents are then retrieved for the user.

$$\mathbf{Cq} = \mathbf{s} \qquad (4)$$

3.5 Increasing usability: Automating indexing and retrieval in Dedal

The approach to automating indexing is to use design-specific thesauri for Dedal descriptor and subject terms. Thesauri have been used in many information retrieval schemes to both expand and focus search. They expand search by providing alternative terms to satisfy part of a query, and focus search by limiting these alternative terms to a set of words or domain, such as mechanical design. Automating the indexing of descriptors has already been addressed in [9], so in this paper, we focus on developing thesauri for subjects.

In [8], a variety of approaches for building subject thesauri by hand were explored. It was found that these performed quite well, but were both expensive to develop and difficult to maintain. Machine generated thesauri are an attractive alternative, desirable for their relative ease of generation and maintainability. In [10], we first attempt to use machine methods to draw out “themes” or “modes” of the collection. The technique is Singular Value Decomposition (SVD) [6], which has been used to determine the underlying meaning of documents. The process involves first decomposing the corpus matrix into three matrices:

$$\mathbf{C} = \mathbf{USV}^{T} \tag{5}$$

where **C** is the corpus matrix, **S** is a diagonal matrix, **V** is an orthonormal matrix of term modes in the corpus, and **U** is an orthonormal matrix that reconstructs **C** from the term modes. These themes are then used as queries on the collection itself to generate sets of related words. The retrieved documents are broken down into quartiles based on the relevance of the retrieved synonym terms. The idea was to see if using terms from only the most relevant terms, rather than all relevant quartiles, would improve results.

Themes were extracted from not only whole documents but also documents that had been divided into fifty-word long blocks. The motivation was to try to extract themes that are contextually ‘close’ [3]. To better understand this, imagine that two words appear together repeatedly in separate, full-length documents. There is a reasonable probability that these words are related. However, if two words from a fifty-word block of text appear together often, the likelihood of the two terms being related together increases.

It was found in [10] that thesauri derived from blocks of text perform much better (~70%) than thesauri derived from whole documents. This demonstrates that proximity improves theme extraction. In this new round of tests, we look at the effect of breaking documents down into 50, 25, 10, and 5 word long blocks for generating thesauri.

4 Results

Figure 4 shows the results of using different combinations of quartiles in the machine generated thesauri. The plots show average precision over a range of recall values for 50, 25, 10 and 5 word-long blocks. It was hypothesized that using only terms from the top quartiles (Q1 and Q12) might improve precision, but the graph shows that using terms from all four quartiles (Q1234) produces significantly better precision over all values of recall.

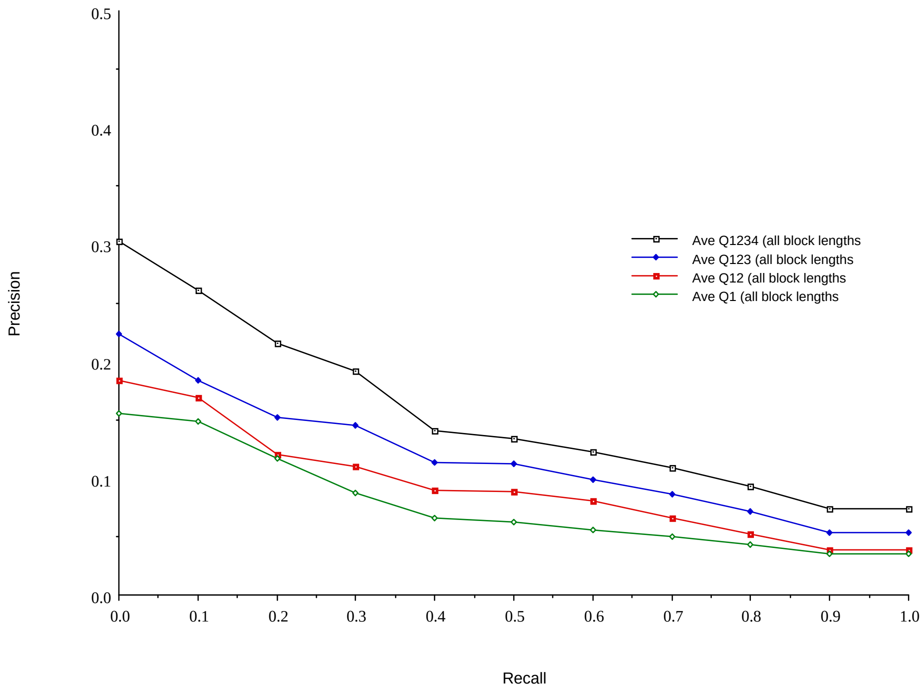

Figure 3. Comparison of the effect of using different combinations of quartiles on performance

Understanding that using all quartiles is the best strategy, we compare the results of testing the various block lengths on retrieval using all quartiles of the thesauri (figure 4). As hypothesized, 50 word long blocks performed the worst, while 25, 5 and 10 word blocks perform comparably well. In particular, 5 and 10 word blocks appear to have very similar precision and recall. In fact, along middle values of recall, 10 word blocks perform noticeably better than the smaller blocks.

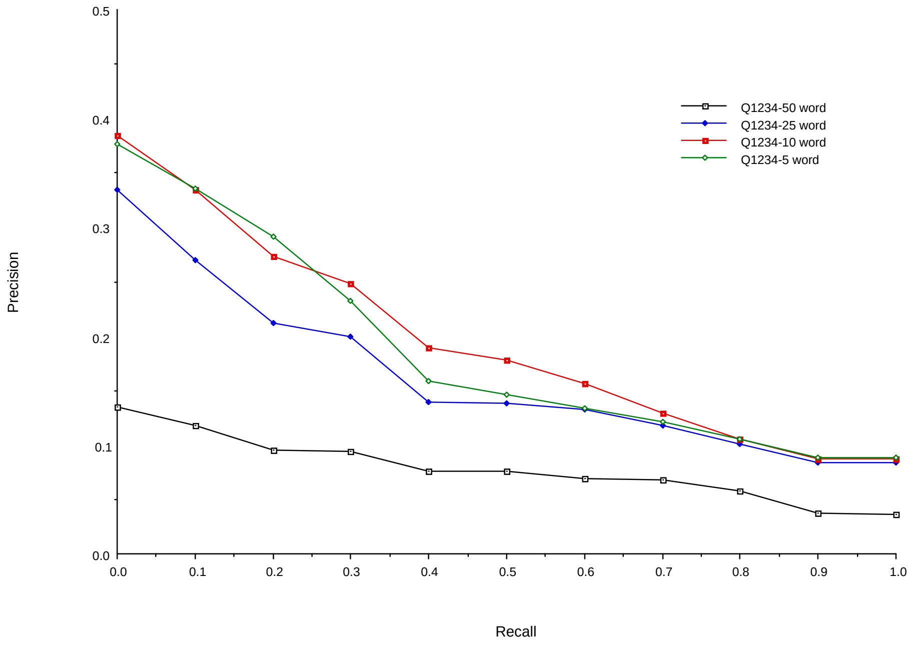

Figure 4. Effect of decreasing block sizes on retrieval performance

5 Conclusions

This paper examined the effect of reducing block size for machine generating thesauri. As expected, decreasing size increases performance. However, 5 and 10 word blocks are quite close, suggesting there is a limit to how small a block can be. In view of these results, future work will concentrate on finding the optimal block size. We will also take other approaches to machine generation of thesauri, using different schemes for finding modes, such as covariance matrices. Future work will also involve user testing of the system

References

[1] Baudin, C., Gevins, J., Baya, V., and Mabogunje, A., "Dedal: Using Domain Concepts to Index Engineering Design Information", *Proceedings of the Ninth National Conference on Artificial Intelligence*, AAAI, Anaheim, CA, July 1991, pp 702-707.

[2] Baudin, C., Underwood, J. G., Baya, V., 1993, "Using Device Models to Facilitate the Retrieval of Multimedia Design Information", *Proceedings of International Joint Conference on Artificial Intelligence*.

[3] Hearst, Marti A., Context and Structure in Automated Full-Text Information Access. PhD Thesis, Computer Science Division, UC Berkeley, UCB/CSD-94/836. 1994.

[4] Hong, J., Toye, G., Leifer, L., "Using the WWW for a Team-Based Engineering Design Class", *Proceedings of the Second WWW Conference*, Chicago, Illinois, October 1994.

[5] Salton, G., 1988, Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer, Addison-Wesley, Reading, MA.

[6] Schutze, H. and Silverstein, C., "Projections for Efficient Document Clustering", *SIGIR* 1997, pp 74-81 (ftp://parcftp.xerox.com/pub/qca/papers)

[7] Wood, W. H., "Supplying Concurrent Engineering Information to the Designer: The Conceptual Design Information Server", PhD Thesis, Dept. of Mech. Eng., UC Berkeley, 1996.

[8] Wood, W. H.; Yang, M. C.; Cutkosky, M. R. and Agogino, A. M.. "Design Information Retrieval: Improving Access to the Informal Side of Design. " *Proceedings of the 1998 ASME 10th International Conference on Design Theory and Methodology*. Atlanta, GA, Sept. 13-16, 1998.

[9] Yang, M. C. and Cutkosky, M. R. "Automated Indexing of Design Concepts for Information Management." *Proceedings of the International Conference in Engineering Design*, Tampere, Finland, Aug. 19-21, 1997.

[10] Yang, M. C.; Wood, W. H., and Cutkosky, M. R. "Data Mining for Thesaurus Generation in Informal Design Information Retrieval." *Proceedings of the 1998 International Congress on Computing in Civil Engineering*. Boston, MA, Oct. 18-21, 1998.

Maria C. Yang
Center for Design Research, Stanford University
Mechanical Engineering Department
560 Panama Mall, Stanford University, Stanford, CA 94305-2232, USA
650-723-7909 TEL
650-725-8475 FAX
mcyang@cdr.stanford.edu