# Exploring the use of natural language systems for fact identification: Towards the automatic construction of healthcare portals

**Frederick A. Peck[1], Suresh K. Bhavnani[2], Marilyn H. Blackmon[3], Dragomir R. Radev[4]**

School of Information, University of Michigan[1, 2]
School of Information and Department of EECS, University of Michigan[4]
Ann Arbor, MI 48109-1092
Email: {peckf[1], bhavnani[2], radev[4]}@umich.edu

Institute of Cognitive Science, University of Colorado[3]
Boulder, CO 80309-0344
Email: blackmon@psych.colorado.edu

**In prior work we observed that expert searchers follow well-defined search procedures in order to obtain comprehensive information on the Web. Motivated by that observation, we developed a prototype domain portal called the Strategy Hub that provides expert search procedures to benefit novice searchers. The search procedures in the prototype were entirely handcrafted by search experts, making further expansion of the Strategy Hub cost-prohibitive. However, a recent study on the distribution of healthcare information on the web suggested that search procedures can be automatically generated from pages that have been rated based on the extent to which they cover facts relevant to a topic.**

**This paper presents the results of experiments designed to automate the process of rating the extent to which a page covers relevant facts. To automatically generate these ratings, we used two natural language systems, Latent Semantic Analysis and MEAD, to compute the similarity between sentences on the page and each fact. We then used an algorithm to convert these similarity scores to a single rating that represents the extent to which the page covered each fact.**

**These automatic ratings are compared with manual ratings using inter-rater reliability statistics. Analysis of these statistics reveals the strengths and weaknesses of each tool, and suggests avenues for improvement.**

## Introduction and motivation

Healthcare information has become one of the most popular search topics on the Web (Fox & Fallows, 2003; Madden & Rainie, 2003). Correspondingly, healthcare organizations, such as The National Cancer Institute (NCI), have spent millions of dollars creating high-quality healthcare domain portals. These portals are written in lay terminology and are targeted at the health consumer, rather than the health professional. The information contained in these portals is extensive – NCI's site, for example, contains information on over 118 different cancers across thousands of webpages.

Despite the popularity of healthcare information searching and the amount of high-quality information on the Web, obtaining comprehensive information for a healthcare topic is not a trivial task for a novice searcher. Recent user studies (Bhavnani, 2001; Bhavnani et al., 2003) show that while expert searchers follow specific *search procedures* to find comprehensive information, novice searchers do not follow such search procedures, and retrieve incomplete, unreliable sources.

To address this situation, we prototyped a new type of domain portal, called a *Strategy Hub*, which provides search procedures to novice users, with high-quality links to satisfy each subgoal. In a user study, the Strategy Hub was shown to improve the performance of novice searchers (Bhavnani et al., 2003).

While there is little question of the utility of the Strategy Hub, the prototype was very costly to build, largely because the search procedures were elicited from search experts. Such handcrafting makes it cost-prohibitive to build a large-scale version of the Strategy Hub that covers many diseases and topics. We therefore set out to automate the process of creating search procedures.

In the next section we will discuss our prior study on the distribution of healthcare on the Web. This study provides insights into why search procedures are so important on the

Web, and informs the automatic generation of search procedures for specific healthcare topics. We then describe two tools, Latent Semantic Analysis (LSA) and MEAD, which could be useful for automating the search procedures. Then, we present the results of two experiments designed to automatically rate the extent to which a fact is covered on a webpage. In the first experiment, ratings were automatically generated using LSA, and in the second experiment, they were generated using MEAD. Finally, we discuss how these two experiments reveal the strengths and weaknesses of each tool in the automatic generation of search procedures, providing a valuable first step towards full automation of the Strategy Hub.

## Previous research on search procedures

*Distribution of healthcare information on the Web*

Why is it so hard for novice searchers to find comprehensive information about a healthcare topic, even when given a list of high quality healthcare sites (Bhavnani et al., 2003)? To address this question, a prior study focused on a single healthcare topic, *melanoma risk and prevention*, and examined how facts related to this topic were distributed across 10 high-quality healthcare websites. This study was conducted in three parts: (1) identify facts necessary for a comprehensive understanding of the topic, (2) generate a corpus of high-quality pages that contained the facts, and (3) determine the extent to which each fact was covered on each page in the corpus.

*1. Identify facts necessary for a comprehensive understanding of the topic.* In the first phase of the study, two skin cancer physicians identified 14 facts that they felt were necessary for a patient to know in order to have a comprehensive understanding of melanoma risk and prevention. Each fact was a single sentence with optional synonyms. The doctors rated each fact for importance on a 1-5 scale (1=not important (and will be dropped from the study), 5=extremely important). Table 1 shows two example facts, and their mean importance.

*2. Generate a corpus of high-quality healthcare pages that contained the facts.* To avoid noise introduced by inaccuracies in healthcare pages (see e.g., Biermann et al., 1999; Griffiths & Christensen, 2000), the corpus used in the study was restricted to pages from only the most reliable sites. This was defined as the union of all sites pointed to on the MEDLINEplus melanoma page[1], and the top sites identified in a recent study of melanoma information on the Web (Bichakjian et al., 2002). After dropping 2 sites that were no longer online, the union resulted in 10 high-quality websites that contained melanoma information.

Table 1. Two example facts related to melanoma risk/prevention.

| Fact | Mean importance |
|---|---|
| Having dysplastic nevi [or atypical moles] increases your risk of getting melanoma [or skin cancer] | 5 |
| Wearing sunscreen can help to prevent melanoma | 4.5 |

Google was used to search within each of these 10 sites for pages that contained at least one of the 14 facts identified by the physicians. This resulted in a corpus of 189 high quality healthcare pages that had a high likelihood of containing information related to melanoma risk and prevention.

*3. Determine the extent to which each fact was covered on each page in the corpus.* Given the facts and pages, the next step was to rate the extent to which each fact was covered on each page in the corpus. These ratings were done by a graduate student at the School of Information, University of Michigan, using a 5-point scale:

0. Fact not covered on page
1. Fact covered in less than one paragraph
2. Fact covered in one paragraph
3. Fact covered in more than one paragraph
4. Page mostly devoted to fact (although the page could cover other facts as well).

To test the reliability of the ratings, a second graduate student was given a random sample of 25% of the pages (the *inter-rater set*), and asked to perform the same ratings. The agreement between the two raters was assessed in two ways. To assess whether the raters agreed on the *presence or absence of a fact on a page*, Cohen's kappa was used, which measures the amount that the agreement between the raters exceeds chance. To measure the agreement on the *extent to which a fact was covered on a page*, Cohen's weighted kappa was used, which gives "partial credit" for ratings that are close to each other (e.g., a disagreement of 3 vs. 4 is not treated the same as a disagreement of 3 vs. 0).

The raters had high agreement on both the presence or absence of a fact on a page (kappa=.806) and the extent to which a fact occurred on a page (weighted kappa=.731). These agreements are considered very good and good, respectively (Altman, 1990).

1 MEDLINEplus is a healthcare domain portal maintained by the National Libraries of Medicine and the National Institutes of Health. The melanoma page (December, 2003) can be accessed at: http://www.nlm.nih.gov/medlineplus/melanoma.html.

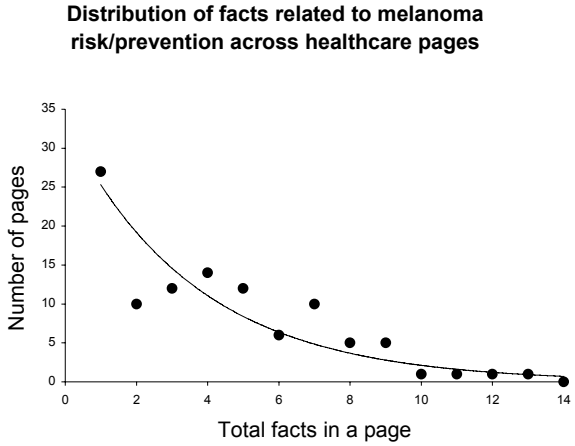


Figure 1. The distribution of facts related to melanoma risk/prevention is highly skewed, and no one page contains all 14 facts.

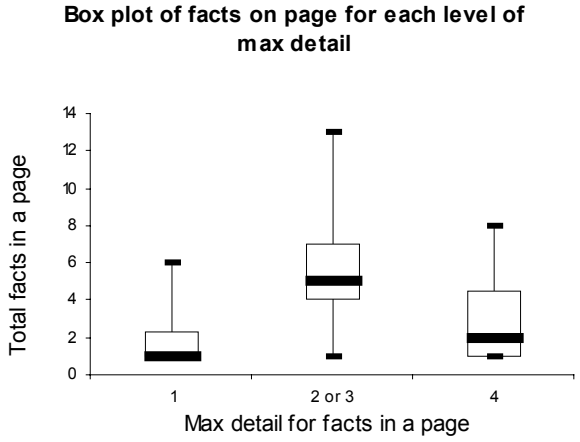


Figure 2. Pages with a maximum detail of 2 or 3 had significantly more facts than pages with a maximum detail of 1 or 4. This suggests the existence of three page-types: *general* pages cover many facts in a medium amount of detail, *specific* pages cover few facts in high detail, and *sparse* pages cover few facts in low detail.

Analysis of these ratings painted an interesting picture of the distribution of healthcare information on the web. As shown in Figure 1, the distribution is highly skewed, meaning that many pages contain few facts while few pages contain many facts. Furthermore, no one page contains all 14 relevant facts.

What is causing this skewed distribution? An exploratory analysis of pages revealed that pages with many facts appeared to provide information in less detail than pages with few facts. As shown in Figure 2, pages with a maximum detail level of 2 or 3 had a significantly higher number of facts ($p<.001$, mean number of facts=5.89, SD=2.63) than pages that had a maximum detail level of 4 (mean=2.87, SD=2.12), or a maximum detail level of 1 (mean=1.86, SD=1.21). This suggests the existence of three page types. *General* pages are written to cover an entire topic, such as melanoma risk/prevention. They occur in the tail of the distribution, and cover many facts in a medium amount of detail. *Specific* pages are written to cover a single fact. They occur in the head of the distribution and cover few facts in high detail. Finally, *sparse* pages cover other topics outside of melanoma risk/prevention, and happen to mention one or two risk/prevention facts. These pages also occur in the head of the distribution, and have few facts with low detail.

The above analysis helps to explain why search procedures are so important to finding comprehensive information on the Web. First, because no one page contains all of the facts related to melanoma risk/prevention, users must visit more than one page to get all of the facts. Second, the page-types suggest a specific order in which the pages should be visited. For example, users should first read general pages to get a broad overview of the topic, followed by specific pages to obtain depth information about specific facts. This *general to specific* strategy is well known by librarians (Kirk, 1974), and is similar to the expert search procedures provided in the Strategy Hub.

This study was therefore an important first step in determining what the automated Strategy Hub must do: accurately classify webpages into general, specific, and sparse. Given a corpus of high-quality pages, a list of facts related to a topic, and the extent to which each page covers each fact, a tool can be developed to automatically classify pages into these three page-types. These classified pages can then be sequenced in a search procedure. However, although the corpus of high-quality pages can be automatically generated, and the list of relevant facts can be elicited from experts with very little cost, the process of rating the pages for each fact is still a time consuming, manual process. Therefore, in order to automatically generate search procedures, we must first automate this sub-process of rating the degree to which a page contains the relevant facts.

## Two natural language tools for automating fact coverage

There exist several tools that can be used to analyze natural language text. Below we describe two well-known

tools that are adept at rating the similarity between two texts.

*Latent Semantic Analysis*

Latent Semantic Analysis (LSA) is designed to capture the semantic similarity (similarity in meaning) between any pair of texts (Landauer et al., 1998). For example, consider the two sentences in Table 2A. These sentences are lexically similar (they have a 75% word overlap) but they are semantically different. That is, they contain similar words, but have an entirely different meaning. The sentences in Table 2B are exactly the opposite – they are lexically different, but semantically similar. That is, they contain different words, but have a similar meaning. Indeed LSA rates the sentence pair in 2B far higher in semantic similarity than the sentence pair in 2A.

Table 2. Examples of (A) sentences that are syntactically similar but semantically different, and (B) sentences that are syntactically different but semantically similar.

| | |
|---|---|
| A | I kicked the ball. |
| | I kicked the bucket. |
| B | I kicked the bucket. |
| | I died. |

In order to extract semantics, LSA must be trained on a large corpus of documents. From this corpus, LSA builds an extensive matrix in which each unique word in the training corpus appears as a row in the matrix and each document appears as a column. Finally, singular value decomposition is used to transform this matrix into a high-dimensional *semantic space*. Each word in the training corpus can be represented as a vector in this semantic space; this vector can be thought of as the "average" meaning of a word.

The similarity of any two words in the training corpus can then be expressed as the cosine of the angle between the two vectors. Importantly, this similarity calculation is not limited to single words. In fact, *any combination* of words in the training corpus can be represented as a vector in the semantic space. Therefore, LSA can calculate the similarity between pairs of phrases, sentences, paragraphs, or documents, as long as they contain words in the training corpus.

Because a semantic space can only represent words in the original training corpus, the calculated similarity of two inputs is highly dependent on whether or not the inputs contain words from the training corpus. Two phrases may be semantically similar, but LSA will compute an inaccurate similarity if either or both phrases contain any words that LSA does not recognize (e.g., words with zero frequency in the semantic space). Avoiding such inaccuracies is an important constraint on proper use of LSA (Landauer 2002).

However, given the right semantic space, LSA has been shown to be remarkably effective. For example, Landauer et al (1998) created several semantic spaces from the set of scientifically sampled documents used by Touchstone Applies Sciences Association (TASA) to create *The Educator's guide to word frequency* (Zeno et al., 1995). The most advanced space simulates the general reading knowledge of college freshmen. Using appropriate semantic spaces LSA was able to do reasonably well on a variety of tasks, such as the TOEFL test (Landauer et al., 1998). In other experiments, a semantic space was created using three college-level psychology textbooks, and tested using multiple-choice exams given in introductory psychology courses at two universities. Although LSA did not score as well as students in the classes, it did well enough to pass the class according to the professor's grading criteria. Other uses of LSA include modeling a user's behavior at on a website (Blackmon et al, 2002, 2003).

LSA also has a clear application to Information Retrieval (IR). For example, Google, the most popular IR system on the Web today, is *lexically-based*, meaning that it returns pages that contain words in the user's query. In contrast, LSA has been used to create *semantic-based* IR systems, which can return pages with words that have similar *meanings* to the user's query (Dumais et al., 1988; Deerwester et al, 1990; Dumais, 1994), even if the lexical overlap between the page and the query is quite small.

The utility of LSA has therefore been demonstrated from a theoretical and practical point of view. In our experiments, we used LSA in a novel way to score the extent to which a fact is covered by a webpage.

*MEAD*

MEAD is a tool developed at the University of Michigan to automatically summarize multiple document clusters (Radev et al, 2000; Radev et al., 2001b)[2]. To create a summary, MEAD first determines the *cluster centroid*, which is a group of words that are central to all documents in the cluster. MEAD then computes the similarity of each sentence in the cluster to the centroid, and chooses the most salient sentences for use in the summary. This process is called *centroid-based summarization*.

In addition to centroid-based summaries, MEAD can create *query-based* summaries as well. Rather than scoring each sentence based on its similarity to a centroid, this method scores each sentence based on its similarity to a

2 MEAD is public-domain software, see http://www.summarization.com/mead (accessed December, 2003). The results reported in this paper were obtained using MEAD version 3.07.

$$\text{Similarity} = \frac{\text{(Total overlapping words)}}{\text{sqrt((total words in sentence) * (total words in target))}}$$

Figure 3.The function used by MEAD to calculate the similarity between a sentence and the target.

given query. The query is simply a sentence or two that describes the topic on which the user wishes to summarize.

Whether MEAD is computing a centroid-based summary or a query-based summary, it first computes the *similarity score* between each sentence and the *target*, which is either the cluster centroid, or the query. As shown in Figure 3, the similarity score is based entirely on the word overlap between the sentence and the target. Because the MEAD similarity score is computed using only the lexical composition of the sentence and the target, there is no analog to a semantic space in MEAD.

MEAD has been used to create two web-based summarization tools, *NewsInEssence* (Radev et al., 2001a), and *WebInEssence* (Radev et al., 2001c). NewsInEssence[3] summarizes related news articles from multiple online news sources, including www.cnn.com and www.msnbc.com. WebInEssence is an IR system that uses MEAD to automatically cluster and summarize similar webpages returned from a user's query.

Both of these example systems use MEAD to extract relevant *sentences* based on their salience to a centroid or user-defined query. However, in our experiments, we used MEAD to score the extent to which a *page* covers a set of facts, based on the salience of each sentence in the page to each fact.

## Experiment to automatically rate pages

### *Overview of experiments*

MEAD and LSA each provide a method of computing the similarity between a fact and a sentence. However, they are not immune to common problems of natural language systems, including the extent to which the system must be trained, and the extent to which the input must be pre-processed prior to using the tool. In general, these problems are approached incrementally, where the results of one experiment suggest ways to modify the training or input in order to improve the results of the next experiment. Therefore, the experiments presented in this section are only the first in a planned *series* of experiments designed to learn the strengths and weaknesses of LSA and MEAD, and ultimately to exploit the strengths to automatically rate the extent to which a fact is covered on a page.

3 NewsInEssence can be accessed at http://www.newsinessence.com (accessed December, 2003)

### *Experiments to automatically rate pages*

In our experiments, each tool was used to rate the similarity between a fact and each sentence page. An algorithm was then used to convert these *similarity scores* into a single rating denoting the extent to which the fact was covered on the page. These ratings were then compared with the manual ratings in the distribution study described above.

### *Data*

Determining the extent to which a fact is covered on a page requires two inputs: (1) a list of facts, and (2) a corpus of pages. Both of these inputs were identified in the manual distribution study discussed above, and were kept the same for this study.

However, the webpages required some pre-processing in order to convert them into a format that each tool could understand. First, the webpages were stripped of their HTML content using a custom script. This content was stored in a database, and subsequently parsed into paragraphs and sentences. Finally, the content was converted into XML files for use in MEAD.

After pre-processing, the corpus was split into two sets. The *inter-rater set* contained 25% of the pages, and was the same inter-rater set used in the manual distribution study described earlier. The *non-interrater set* contained the remaining 75% of the corpus. We split the pages into these two sets because only the pages in the inter-rater set were rated by both raters in the distribution study. Therefore, in order to compare the automatic ratings with both manual raters, we tested the automatic tools using only pages in the inter-rater set.

### *Method*

LSA requires a semantic space in order to compute similarity scores. In our experiments, we used two separate semantic spaces. First, in order to obtain a baseline metric, we used the college-level semantic space (described in the description of LSA). We chose this space knowing fully well that melanoma is a specialized topic, and that this general space might not contain several crucial words and phrases related to melanoma. We therefore also created a *melanoma-specific* semantic space. This space was created using 128 webpages related to melanoma, obtained from the 10 high-quality sites used in the distribution study. Because the pages in the inter-rater set were used to the test the tools, to avoid circularity we did not use these pages to create the space. In order to keep separate the ratings in the general semantic space from the ratings in the

```
For a given fact and page:

For each paragraph, determine the percentage of
sentences that "match" the fact - sentences
whose similarity with the fact is greater than
some threshold, $similarity_threshold.

       If more than 2/3 of the sentences
       in a paragraph match the fact,
       then the paragraph is devoted to
       the fact.

Rating=0 if 0 sentences match the fact

Rating=1 if at least 1 sentence matches the
fact, but 0 paragraphs match the fact

Rating=2 if only one paragraph matches
the fact, and no other sentences match
the fact

Rating=3 if more than one paragraph
matches the fact

Rating=4 if more than 2/3 of the paragraphs in a
page match the fact
```

Figure 4. Brief description of the algorithm used to calculate the extent to which a fact is covered on a page, given similarity scores between the fact and each sentence on the page. The complete algorithm is given in Appendix 2.

melanoma space, we will use the term *LSA_general* to refer to LSA ratings in the general space, and *LSA_melanoma* to refer to ratings in the melanoma space.

As discussed in the LSA section above, these semantic spaces are necessary for LSA to determine the *meaning* of a phrase. On the other hand, as shown in Figure 3, MEAD simply computes the word overlap between two phrases, and therefore does not require a semantic space.

After choosing the semantic spaces for LSA, we used each tool to compute the similarity between each fact and each sentence in the inter-rater set, resulting in over 65,000 similarity scores for each tool. We then used these similarity scores to rate the extent to which a fact was covered on a page according the 5-point scale used in the distribution study

0. Fact not covered on page
1. Fact covered in less than one paragraph
2. Fact covered in one paragraph
3. Fact covered in more than one paragraph
4. Page mostly devoted to fact (although the page could cover other facts as well)

Figure 4 gives a brief description of the algorithm used to convert the similarity scores into ratings. The complete algorithm is shown in the appendix.

As shown in Figure 4, `$similarity_threshold` is the value at which we consider a sentence to match a fact, and is a key input to the algorithm. Therefore, before we could use the algorithm to rate pages, we had to learn the optimal `$similarity_threshold` for each tool. To learn this optimal value, we conducted a small *learning experiment*.

To conduct the learning experiment, we first selected 10 random pages from the non-interrater set to be used as a *learning set*. We then used the above algorithm to rate each of the pages in the learning set for various values of `$similarity_threshold`. The optimal value for `$similarity_threshold` is the value at which the automatic ratings best match the manual ratings. Thus, we chose the value that maximized the weighted kappa between the automatic ratings and the manual ratings for each tool.

Given this optimal value for `$similarity_threshold`, we ran the algorithm for each tool, and computed two statistics to assess the agreement between the automatic ratings and the manual ratings. (1) To assess whether the raters agreed on the *presence or absence of a fact on a page*, we used Cohen's kappa. (2) To measure the agreement on the *extent to which a fact was covered on a page*, we used Cohen's weighted kappa, which gives "partial credit" for ratings that are close to each other. The above kappa statistics are the standard method to calculate inter-rater reliability. They are a better estimate of the true reliability between two raters than simply the percentage agreement because kappa takes into account the probability that the raters will agree by chance.

*Results*

*Learning experiment.* As determined by the learning experiment, the optimal `$similarity_thresholds` were: LSA_general: .55; LSA_melanoma: .45; MEAD: .25.

*Agreement on whether or not a fact was covered on the page.* Table 5 shows the agreement between the automatic and the manual ratings for whether or not a fact was covered on a page, assessed using Cohen's kappa (percentage agreements are in parentheses). As shown, LSA_melanoma had the highest agreement with the manual raters, with kappa values close to .3. However, this agreement was considerably lower then the agreement between the two manual raters (kappa=.806). The agreement for MEAD was lower than LSA_melanoma, but approximately double that of LSA_general.

*Agreement on the extent to which a fact is covered on a page.* Table 6 shows the agreement between the automatic and the manual ratings for the extent to which a fact was covered on a page, assessed using Cohen's weighted kappa (percentage agreements are in parentheses). Again, LSA_melanoma had the highest agreement with the manual raters with weighted kappa

Table 5. The agreement for fact presence/absence on a page. The table shows the kappa values for inter-rater reliability, with the observed percentage agreement in parentheses.

| | Manual-1 | Manual-2 |
|---|---|---|
| **Manual-1** | 1 | .806 (96%) |
| **Manual-2** | .806 (96%) | 1 |
| **LSA_general** | .102 (60%) | .096 (60%) |
| **LSA_melanoma** | .322 (87%) | .295 (85%) |
| **Mead** | .245 (83%) | .221 (81%) |

Table 6. The agreement for the extent to which a fact was covered on a page. The table shows the weighted kappa values for inter-rater reliability, with the observed percentage agreement in parentheses.

| | Manual-1 | Manual-2 |
|---|---|---|
| **Manual-1** | 1 | .734 (93%) |
| **Manual-2** | .734 (93%) | 1 |
| **LSA_general** | .093 (56%) | .072 (54%) |
| **LSA_melanoma** | .319 (84%) | .235 (81%) |
| **Mead** | .197 (80%) | .151 (77%) |

values of .235 and .319. Again, however, this agreement was considerably lower then the agreement between the two manual raters (weighted kappa = .731). Finally, the weighted kappa for MEAD was approximately double that of LSA_general.

*Discussion*

Kappa measures the extent to which the agreement between two raters exceeds chance agreement. Therefore, if kappa = 0, the agreement is simply that which would be expected by chance. In our experiments, all of the kappa and weighted kappa values were greater then 0, meaning that the agreement between the automatic ratings and the manual ratings is at least better than the agreement that we would expect by chance.

Furthermore, it is pertinent to note that even the manual raters did not agree 100%. In fact, the kappa values for the manual raters range between .734 and .806. Because it is unrealistic to expect that an automatic tool will perform as well as a human, even this level of agreement is probably unattainable. In this light, the agreement between LSA_melanoma and the manual raters is actually quite encouraging, and suggests that automatically rating the extent to which a fact is covered on a page is indeed possible.

However, using the current tools and algorithm, the agreement between the automatic ratings and the manual ratings is still too low. Although there is no significance test for kappa, Altman (1990) suggests that kappa values between 0 and .2 should be interpreted as *poor*, while kappa values between .2 and .4 should be interpreted as *fair*. According to these guidelines, the agreement between the LSA_melanoma ratings and the manual ratings can be interpreted as "fair" in both Tables 5 and 6, as can the agreements between MEAD and the manual raters shown in Table 5.

Although we expected low agreement for LSA_general, we were surprised that the agreement for MEAD was so much higher than the LSA_general. After all, LSA employs a complicated algorithm to get at passage meaning, rather than relying solely on lexical composition as MEAD does. However, a brief example will illustrate why this occurred. For clarity, we will use an example of a single word, however the discussion applies to phrases as well.

As discussed in the prior work section, LSA represents any word or collection of words as a vector in a semantic space. This vector is supposed to embody the *meaning* of the word, and can only be computed if the word itself is contained in the space. Thus, the similarity between two words can only be computed if both words are contained in the semantic space.

For example, consider the word "dysplastic." A dysplastic, or atypical, mole is a mole that has many of the same physical characteristics as a melanoma, but is not cancerous. However, dysplastic moles seem to have a greater chance than regular moles of becoming melanomas, so having dysplastic moles is a risk factor for melanoma (see Table 1).

But the word dysplastic is not in the college-level general semantic space. This means that LSA_general cannot compute the similarity between "dysplastic" and "atypical", because it is unable to determine the meaning of "dysplastic" (Landauer, 2002). Even though the two words are semantically very close, the similarity score between the two will be NULL (considered 0 by our algorithm). In fact, the similarity between "dysplastic" and "dysplastic" is NULL in LSA_general, even though they are the same word!

Now consider the same example in MEAD (refer to the formula in Figure 3). Again the similarity between "dysplastic" and "atypical" would be 0, because even though the words are semantically similar, the two phrases have no words in common. However, "dysplastic" and "dysplastic" would have a similarity score of 1, because every word in one phrase is contained in the other phrase.

This suggests that MEAD will outperform LSA when many of the input words are not represented in the semantic space. This seems to be the case with LSA_general and our corpus. Even though the input pages were written for consumers, they still contained language

specialized to melanoma (e.g., "dysplastic"). Because this language was not in the college-level space, LSA_general ignored it, leading to low similarity scores even for closely matching sentences.

This situation should have been addressed by the melanoma-specific semantic space. Indeed, the agreements between LSA_melanoma and the manual raters were considerably higher than the agreements between the manual raters and LSA_general or MEAD. However, even the highest agreement between LSA_melanoma and the manual raters was still relatively low. To help understand why this agreement was so low, we examined the frequency table of ratings between LSA_melanoma and Manual-2.

The frequency table is shown in Table 7. Each cell in the table displays the number of times that a pair of ratings occurred. For example, the top-left cell shows the number of times that both raters rated a 0. Cells along the main diagonal represent perfect agreement between the raters. The greater the distance a cell is from the main diagonal, the greater the disagreement between the raters. Cells below the main diagonal are cases where LSA_melanoma *underrated* the page, and cells above the diagonal are cases where LSA_melanoma *overrated* the page, compared to Manual-2.

Table 7. The frequency table for LSA_melanoma and Manual-2 shows that the majority of the disagreements were on whether or not a fact was covered on a page.

| | LSA_melanoma | | | | |
|---|---|---|---|---|---|
| **Manual-2** | 0 | 1 | 2 | 3 | 4 |
| 0 | 473 | 27 | 16 | 3 | 0 |
| 1 | 24 | 1 | 12 | 3 | 0 |
| 2 | 8 | 1 | 3 | 1 | 0 |
| 3 | 8 | 1 | 1 | 0 | 0 |
| 4 | 2 | 0 | 3 | 1 | 0 |

The frequency table therefore allows us to examine the nature of the disagreements in more detail. For example, as shown in the first row and first column of the table, the majority of the disagreements were on the presence or absence of a fact on the page. The first row shows that there were 46 instances where LSA_melanoma rated a page as containing a fact (rating > 0) when the manual rater rated the page as not containing the fact (rating = 0). Similarly, the first column shows that there were 42 instances where the manual rater rated the page as containing a fact when LSA_melanoma rated a page as not containing the fact.

The remainder of the table shows the frequency of ratings when the raters agreed that a fact was covered on a page. As shown, when both raters agreed that a fact was covered on a page, the disagreements on the extent of the coverage tended to be slight. Indeed, 70% of these disagreements were by only 1 point. Because the disagreements tended to be on whether or not a fact was covered on a page rather than on the extent to which a fact was covered on a page, our future research will attempt to improve the agreement on whether or not a fact is covered on a page.

*Future research*

We are currently exploring ways to improve the agreement on whether or not a fact is covered on a page, For example, we are exploring how to combine the scores from LSA_melanoma, LSA_general, and MEAD. These tools had relatively low agreement on whether or not a fact was covered on a page, which implies that when LSA_melanoma rated a fact as covered on a page, LSA_general or MEAD may not have, and vice versa. The ratings might therefore be improved by combining the three scores. One way to combine the scores would be a linear combination of the three, with the optimum weights for each score learned in much the same way as the optimum `$similarity_threshold`. Another way would be to consider a sentence as matching a fact if at least one tool matched the fact (or if at least two tools matched, etc).

In addition to combining the scores, we can also improve the agreement by improving the individual ratings. It is worth noting that the task we gave LSA is difficult. LSA performs best when input texts are long (Landauer et al., 2000), but the facts (see Table 1) contain relatively few words. Furthermore, the facts are also sometimes very similar to each other. While humans can use logic and syntax to distinguish between these similar facts, LSA cannot. Therefore, the LSA ratings may improve if facts are elaborated into one or more paragraphs that make facts more distinct and less similar to each other.

This elaboration should also decrease the importance of *low-frequency words* in the facts. Low frequency words are words in the input text that appear less than 15 times in the semantic space. There were 9 low-frequency words in the LSA_general space: *ABCDs, atypical, dysplastic, melanoma, nevi, Pigmentosum, sunscreen, UV,* and *Xeroderma*. On the other hand, there were 22 low-frequency words in the LSA_melanoma space, including *green, seeking, itching,* and *match*. Thus, the low frequency words in the general space tended to be specific to melanoma, while the low frequency words in the melanoma space tended to be general in nature. To address this situation, we are exploring the creation of a *general healthcare space* created from the medical encyclopedia on MEDLINEplus. We believe that this space will include many general words, as well as many medical terms.

Finally, we may believe that we can improve the MEAD results using a just-released, trainable version of MEAD

that was not available for our original experiments. We plan to train this tool using the same corpus that was used to create the melanoma-specific semantic space. Such training will make the tool more sensitive to the melanoma-specific content in the webpages, allowing it to more accurately distinguish between lexically similar phrases. This should increase the reliability of the ratings.

As discussed in the introduction, our ultimate goal is to automate the creation of the Strategy Hub. When using this automatic tool, a user will first select a healthcare topic (such as melanoma risk/prevention). The system will then: (1) retrieve a list of relevant facts from a physician-supplied database, and a list of high quality websites, (2) use Google to find pages from the high quality sites that contain at least one of the facts, (3) use the tools and algorithm described in this paper to rate the extent to which the facts are covered in the pages, (4) classify pages into general, sparse and specific, and finally, (5) present pages to the user in a pre-determined sequence, such as from general to specific. This tool should help novice searchers to obtain comprehensive healthcare information.

## Conclusion

We began with a description of a prototype system called the Strategy Hub, which provides expert search procedures to novice users. However, because these search procedures were manually determined by experts, the creation of future Strategy Hubs is cost prohibitive. We therefore set out to automatically generate search procedures. A study on the distribution of healthcare information on the Web provided a roadmap towards this automation. The study showed that that there seem to be three types of pages on the Web: general, specific, and sparse. The existence of these page-types suggests a general-to-specific search procedure, in which a user visits general pages to get an overview of the topic, followed by specific pages to get detailed information about specific facts.

Automatically generating search procedures, then, requires automatically identifying general and specific pages, which requires automatically rating the extent to which a fact is covered on a page. Towards this end, we described how we used MEAD and LSA to determine the similarity between a sentence and a fact. Furthermore, we described the algorithm used to covert these similarity scores into a rating of the extent to which the fact was covered on a page.

To test whether these automatic ratings agreed with the manual ratings in the distribution study, we computed inter-rater agreement statistics. At best, the agreements between the automatic ratings and the manual ratings were fair. However, they exceeded chance in all cases, which encourages us that reliable automatic ratings of pages should be possible. Examining these ratings revealed the strengths and weaknesses of each tool, and suggested methods to improve the agreement with the manual raters. Future work will attempt to improve these ratings by combining the sentence similarity scores from each tool, and by modifying the inputs to improve the sentence similarity scores. The ultimate goal is to develop automatic tools that will guide users in finding accurate and comprehensive information in vast, unfamiliar domains such as healthcare.

## ACKNOWLEDGMENTS

This research was supported in part by the National Science Foundation, Award# EIA-9812607. The views and conclusions contained in this document should not be interpreted as representing the official policies, either expressed or implied, of NSF or the U. S. Government. The authors thank G. Furnas, R. Little, D. Mandalia, and R. Thomas for their contributions.

## REFERENCES

Altman D. G. (1990). *Practical statistics for medical research.* London: Chapman and Hall.

Bhavnani, S. K. (2001). Important cognitive components of domain-specific search knowledge. *Proceedings of TREC'01*, NIST, 571-578.

Bhavnani, S.K., Bichakjian, C.K., Johnson, T.M., Little, R.J., Peck, F.A., Schwartz, J.L., & Strecher, V.J. Strategy Hubs: Next-generation domain portals with search procedures. *Proceedings of CHI'03*, ACM Press, 393-400.

Bichakjian, C., Schwartz, J., Wang, T., Hall J., Johnson, T., & Biermann, S. (2002). Melanoma information on the Internet: Often incomplete-a public health opportunity? *Journal of Clinical Oncology, 20*, 1, 134-141.

Biermann, J.S., Golladay, G.J., Greenfield, M.L., & Baker, L.H. (1999). Evaluation of cancer information on the Internet. *Cancer, 86*, 3, 381-90.

Blackmon, M. H., Kitajima, M., & Polson, P. G. (2003). Repairing usability problems identified by the Cognitive Walkthrough for the Web. *Chi Letters, 5: Proceedings of CHI 2003,* 497–503 *(*ACM Press).

Blackmon, M. H., Polson, P. G., Kitajima, M., & Lewis, C. (2002). Cognitive Walkthrough for the Web. *Proceedings of CHI'02*, ACM Press, 463-470.

Deerwester, S., Dumais, S. T., Furnas, G. W., Landauer, T. K., & Harshman, R. (1990). Indexing By Latent Semantic Analysis. *Journal of the American Society for Information Science, 41*, 391-407.

Dumais, S. T., Furnas, G. W., Landauer, T. K. and Deerwester, S. (1988), Using latent semantic analysis to improve information retrieval. In *Proceedings of CHI'88,* ACM Press, 281-285

Dumais, S. T. (1994), Latent Semantic Indexing (LSI) and TREC-2. *Proceedings of TREC'94*, NIST, 105-116

Fox, F., & Fallows, D. (2003). *Internet health resources* (Pew Internet & American Life Project, Report, July 16, 2003.) http://www.pewinternet.org/reports/toc.asp?Report=95.

Griffiths, K.M., & Christensen, H. (2000). Quality of web based information on treatment of depression: cross sectional survey. *BMJ, 321*, 1511-1515

Kirk, T. (1974). Problems in library instruction in four-year colleges. In: Lubans, John, Jr. (ed.), *Educating the library user*, 83-103. New York: R. R. Bowker.

Landauer, T. K. (2002). On the computational basis of learning and cognition: Arguments from LSA. In N. Ross (Ed.), *The psychology of learning and motivation, 41*, 43-84.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). Introduction to Latent Semantic Analysis. *Discourse Processes, 25*, 259-284.

Landauer, T. K., Laham, D., & Foltz, P. W. (2000). The Intelligent Essay Assessor. *IEEE Intelligent Systems, 15*, 5, 27-31.

Madden, M., & Rainie, L. (2003). America's online pursuits: The changing picture of who's online and what they do (Pew Internet & American Life Project, Report, Dec. 22, 2003). http://www.pewinternet.org/reports/toc.asp?Report=106.

Radev, D.R., Blair-Goldensohn, S., & Zhang, Z. (2001a). Experiments in single and multi-document summarization using MEAD. *Proceedings of the* First Document Understanding Conference

Radev, D.R., Blair-Goldensohn, S., Zhang, Z., & Raghavan, R.S. (2001b). Newsinessence: A system for domain-independent, real-time news clustering and multi-document summarization. Proceedings of the Human Language Technology Conference'01

Radev, D.R., Fan, W. & Zhang, Z (2001c). Webinessence: A personalized web-based multi-document summarization and recommendation system. AACL Workshop on Automatic Summarization'01.

Radev, D.R., Jing, H., & Budzikowska, M. (2000) Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies. ANLP/NAACL Workshop on Summarization'00.

Zeno, S. M., Ivens, S. H., Millard, R. T., & Duvvuri, R. (1995). *The educator's word frequency guide.* Brewster, NY: Touchstone Applied Science Associates, Inc.

**Appendix: Algorithm used to rate pages**

- Obtain list of facts necessary for a comprehensive understanding of the topic.

  Let `TotalFacts` denote the total number of facts. The $i^{th}$ fact is referred to as $\texttt{fact}_i$

- Parse each page into paragraphs and sentences.

  Let `TotalSentencesInPage` denote the total number of sentences in the page.

  Let `TotalParagraphsInPage` denote the total number paragraphs in the page

  Let $\texttt{sentence}_{j,k}$. denote the $j^{th}$ sentence in the $k^{th}$ paragraph.

  Let `TotalSentencesInParagraph(k)` denote the number of sentences in $\texttt{paragraph}_k$.

- Let `SematicProximityForSentence(i,j,k)` denote the semantic proximity (cosine angle) between $\texttt{fact}_i$ and $\texttt{sentence}_{j,k}$
- Let `SimilarityThreshold` be the semantic proximity score above which a sentence is considered to match a fact.
- If `SematicProximityForSentence(i,j,k) >= SimilarityThreshold`, then `facti` matches $\texttt{sentence}_{j,k}$.

  Let `TotalMatchedSentencesInParagraph(i,k)` denote the total number of sentences in $\texttt{paragraph}_k$ that matches $\texttt{fact}_i$.

  Let `TotalMatchedSentencesInPage(i)` denote the total number of sentences in the page that match $\texttt{fact}_i$.

  Let `ProportionOfMatchedSentencesInParagraph(i,k)` denote the proportion of sentences in $\texttt{paragraph}_k$ that match $\texttt{fact}_i$. This can be computed by:

  ```
  TotalMatchedSentencesInParagraph(i,k)/
  TotalSentencesInParagraph(k).
  ```

- If `ProportionOfMatchedSentencesInParagraph(i,k) > .66` then $\texttt{Paragraph}_k$ is considered to be devoted to $\texttt{fact}_i$

  Let `TotalMatchedParagraphsInPage(i)` denote the total number of paragraphs in the page that are devoted to $\texttt{fact}_i$.

  Let `ProportionOfMatchedParagrpahsInPage(i)` denote the proportion of paragraphs in the page that are devoted to $\texttt{fact}_i$. This can be computed as:

  ```
  TotalMatchedParagraphsInPage(i)/TotalParagraphsInPage.
  ```

- Let $\texttt{Rating}_i$ denote the detail level at which a page covers a fact, based on the

following scale:

If $\texttt{fact}_i$ is not matched in page, then $\texttt{Rating}_i$=0

If less than 1 paragraph is devoted to $\texttt{fact}_i$, then $\texttt{Rating}_i$=1

If one paragraph is devoted to $\texttt{fact}_i$, then $\texttt{Rating}_i$=2

If more than 1 paragraph is devoted to $\texttt{fact}_i$, then $\texttt{Rating}_i$=3

If entire page is mostly devoted to $\texttt{fact}_i$, then $\texttt{Rating}_i$=4

This can be computed as follows:

If `TotalMatchedSentencesInPage(i)`=0, then $\texttt{Rating}_i$=0
*(There are no sentences in the page that match* $\texttt{fact}_i$*)*

Else, if `TotalMatchedParagraphsInPage(i)`=0, then $\texttt{Rating}_i$=1
*(There is at least one sentence, but not an entire paragraph devoted to* $\texttt{fact}_i$*)*

Else, if `TotalMatchedParagraphsInPage`=1

Let `MatchedParagraph` denote the paragraph that is devoted to $\texttt{fact}_i$

If `TotalMatchedSentencesInParagraph(i,MatchedParagraph)`= `TotalMatchedSentencesInPage(i)`, then $\texttt{Rating}_i$=2
*(There is one paragraph devoted to* $fact_i$ *and no sentences outside of this paragraph match* $\texttt{fact}_i$*)*

Else $\texttt{Rating}_i$=3
*(There is one paragraph devoted to* $\texttt{fact}_i$*, and at least one sentence outside of this paragraph that also mathches* $\texttt{fact}_i$*)*

Else

If `ProportionOfMatchedParagraphsInPage(i)`>.66, then $\texttt{Rating}_i$=4
*(At least 2/3 of the paragraphs in the page are devoted to* $\texttt{fact}_i$*, so the page is mostly devoted to* $\texttt{fact}_i$*)*

Else, $\texttt{Rating}_i$=3
*(At least 2 paragraphs are devoted to* $\texttt{fact}_i$*, but less than 2/3 of the total paragraphs on the page are devoted to* $\texttt{fact}_i$*)*