# Making MIRACLEs: Interactive Translingual Search for Cebuano and Hindi

DAQING HE, DOUGLAS W. OARD, JIANQIANG WANG, JUN LUO, DINA DEMNER-FUSHMAN, KAREEM DARWISH, and PHILIP RESNIK
University of Maryland, MD
SANJEEV KHUDANPUR
The John Hopkins University, MD
MICHAEL NOSSAL and MICHAEL SUBOTIN
University of Maryland, MD
and
ANTON LEUSKI
University of Southern California, CA

Searching is inherently a user-centered process; people pose the questions for which machines seek answers, and ultimately people judge the degree to which retrieved documents meet their needs. Rapid development of interactive systems that use queries expressed in one language to search documents written in another poses five key challenges: (1) interaction design, (2) query formulation, (3) cross-language search, (4) construction of translated summaries, and (5) machine translation. This article describes the design of MIRACLE, an easily extensible system based on English queries that has previously been used to search French, German, and Spanish documents, and explains how the capabilities of MIRACLE were rapidly extended to accommodate Cebuano and Hindi. Evaluation results for the cross-language search component are presented for both languages, along with results from a brief full-system interactive experiment with Hindi. The article concludes with some observations on directions for further research on interactive cross-language information retrieval.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information search and retrieval

General Terms: Query formulation, Search process, Selection process

Additional Key Words and Phrases: Cross-language information retrieval, Interactive information retrieval, Machine translation

This work has been supported in part by DARPA cooperative agreement N66001-00-2-8910.
Authors' addresses: He, Oard, Wang, Luo, Demner-Fushman, Darwish, Resnik, Nossal, Subotin, University of Maryland, College Park, MD 20742, email: {daqingd;jun;resnik;nossal;msubotin}@umiacs.umd.edu, {oard;wangjq;kareem}@glue.umd.edu; Khudanpur, The Johns Hopkins University, Baltimore, MD 21218, email: khudanpur@jhu.edu; Leuski, University of Southern California, Marina del Rey, CA 90292, email: leuski@isi.edu.

## 1. INTRODUCTION

Over the past four decades, search processes have evolved to exploit new capabilities. Searches based solely on Boolean logic have been overtaken by those that augment Boolean logic with relevance ranking, assessment based on abstracts has been extended through seamless access to the full text of selected documents, and searches performed by trained intermediaries are now less common than information seeking by end users. Support for cross-language information retrieval (CLIR), in which searchers seek to find relevant documents in a language that they may not be able to read, is generally following the same path, but fully interactive search by end users is not yet common. That is the focus of the work reported in this article.

We view searching for information as, ultimately, a human activity. People pose questions, interpret what they read, and determine when their needs have been met. The machine provides some essential capabilities during the process, notably speed, scalability, and consistency. But machines lack many important characteristics that humans can bring, including intentionality, understanding, and serendipity. Therefore, the initiative in the search process lies with the human searchers. However, human searchers have weaknesses, especially at the start of a search session, where their understanding of what they are looking for and how to find it are often inadequate. A well-designed interactive search system thus should help human searchers to overcome their limitations, while drawing on human strengths to cover system weaknesses. One key strategy for achieving this synergy is known as "iterative refinement."

Iterative refinement depends on two types of knowledge: an understanding of why the machine produced the results that were obtained, and an understanding of the ways in which the outcome could be altered. Searchers who lack the ability to read the languages in which the documents are written need some form of automated translation assistance. Current "machine translation" technology is far from perfect, and the effects of its deficiencies on an interactive search process are not yet well understood. We, therefore, built MIRACLE (Maryland Interactive Retrieval Advanced Cross-Language Engine), an interactive search system designed to support rapid prototype iteration, to explore interaction design for interactive CLIR. In this article, we describe the design of the MIRACLE system and explain how it was rapidly adapted to handle Cebuano and Hindi documents.

Rapid incorporation of new document languages was an original design goal for MIRACLE. The MIRACLE query language is always English, so our design objective is to provide effective information access to searchers who are unable to read the document language. The system incorporates the following four key innovations.

*User-assisted query translation*: This is designed to provide greater transparency and control, facilitating the searcher's development of mental models of system operation. Automated creation of explanations makes it possible to rapidly apply this technique to new languages.

*Progressive refinement*: Search results are presented immediately using all known translations, and then updated in response to control actions. Used with iterative refinement, this provides rapid feedback on the effect of control actions.

*Weighted structured query methods*: Evidence about translation probabilities are assembled from multiple sources and used to optimize the ranked display of retrieval results. Structured queries make it possible to exploit term-scale and document-scale evidence simultaneously.

*Configurable translation*: Three factors are important when displaying translated documents: accuracy, fluency, and focus. Early in the introduction of a new language, term-by-term gloss translation (see Section 5.2) can provide some degree of accuracy and fluency; eventually statistical machine translation typically outperforms gloss translation. We, therefore, provide extensive support for gloss translation and a set of focus mechanisms based on passage selection and term highlighting that can easily be applied to any translation results.

We began our work with MIRACLE on day 8 of the 10-day Cebuano surprise language "dry run." In a three-day period, we brought up the system using only gloss translation. We began working with MIRACLE on day 1 of the 29-day Hindi surprise language exercise, starting with gloss translation, later incorporating statistical machine translation, and ultimately evolving the interface design substantially based on a series of informal design critiques from experts in human–computer interaction. At the end of the Hindi exercise, we conducted one small user study in which search effectiveness was the focus. Throughout this process, we made substantial improvements to our automated components as better language resources became available. Our fundamental interaction design remained stable throughout the period, however.

In the next section, we describe the interactive search process, and the way in which we designed the interaction between a searcher and the system to support that process for CLIR. The three subsequent sections then present the three key technical capabilities that underpin that design: Section 3 describes four ways of explaining the meaning of translated query terms, Section 4 describes how documents are ranked in response to a translated query, and Section 5 explains how translations of documents (or passages extracted from documents) were prepared for presentation to the user. Section 6 then explains how those capabilities were integrated to construct the MIRACLE system. Evaluation results for the ranked retrieval component and the entire system are then presented in Section 7. Finally, we conclude with a few remarks about what we have learned and the implications for future research on interactive CLIR in Section 8.

## 2. INTERACTION DESIGN

Searchers often find that their understanding of what they are actually looking for and of how to search for it are incomplete at the start of a search session. Strategies based on iterative refinement, which leverages easy access to full text to support increasingly focused exploratory searches, are commonly used in such cases [Marchionini 1995]. The searchers can be viewed as seeking to refine

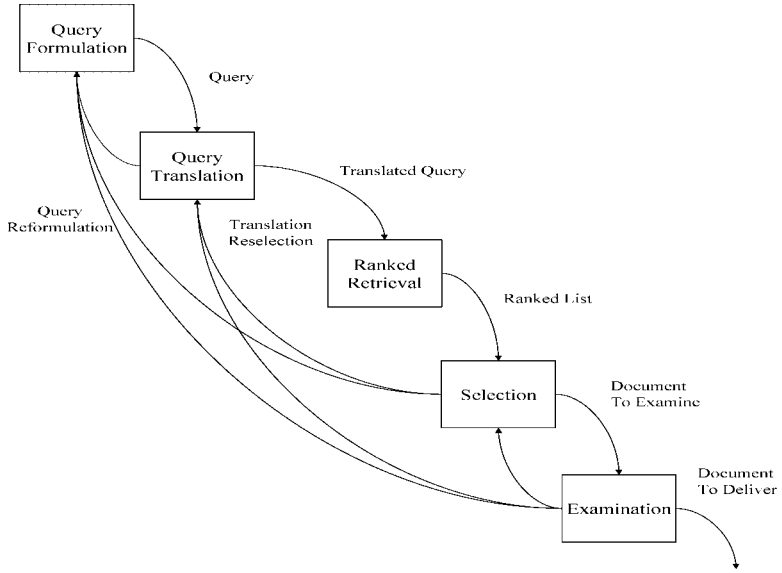


Fig. 1. MIRACLE interaction design. The four interaction points are query formulation, query translation, selection, and examination. Ranked retrieval is a fully automatic process.

three mental models: (1) their information need, (2) appropriate query terms that might be present in the documents that are sought, and (3) ways of combining these terms to best express the need (i.e., the "query language"). Searchers often engage in probing behavior to support refinement of these mental models.

Figure 1 illustrates the four interaction opportunities provided by MIRACLE. Three of these, query formulation, selection from a ranked list, and examination of selected documents, are familiar from monolingual applications such as Web search engines. The fourth, query translation, has typically been treated as an automated step in prior work. We are aware of one prior system to explore interactive query translation, the New Mexico State University Keizai system [Ogden et al. 1999]. In Keizai, searchers were able to select the appropriate translations based on manually prepared English definitions of each translation alternative. MIRACLE provides a similar capability, but our English "definitions" are automatically generated.

Perhaps surprisingly, our principal motivation for incorporating interactive query translation in MIRACLE was not to improve the effectiveness of the forward search path in single iteration. Selecting appropriate translation choices can indeed improve the quality of the result set, but inadvertent removal of a useful translation can equally well adversely affect the search results. Rather, it is the five feedback paths that are the focus of our interaction design. Searchers can reformulate their query or their translation selections based on viewing translations, examining the ranked list, or examining the full text of translated documents. If searchers make bad choices in one stage, they can see the effect and learn to make better choices in future iterations. Our goal is therefore to allow searchers to iterate towards improved searches that meet their needs.

The principal way in which we seek to operationalize this benefit is by focusing on supporting the selection of appropriate query terms. When the query and the documents are expressed in the same language, the searcher must ultimately discover query terms that were used by the authors of the documents that are sought. Searchers typically do this by initially guessing some terms, conducting a search, and then observing the way in which terms are being used in the result set to express ideas. If they were to use an a fully automated cross-language systems, searchers would need discover query terms that the system would translate into terms that might be found in the relevant documents, but they must do so indirectly by examining translations, rather than the original documents. Interactive translation selection serves to demystify the query translation process, thus supporting the development of useful mental models for both system behavior and vocabulary selection.

In a series of controlled user studies for the Cross-Language Evaluation Forum's interactive track, we have found that interactive translation selection can have a beneficial effect on search outcomes, and that the effect is more pronounced with longer search sessions [He et al. 2002; Dorr et al. 2003]. This comports well with our intuition regarding the iterative refinement of mental models.

## 3. EXPLAINING QUERY TERM TRANSLATIONS

User-assisted query translation can only be helpful if the searchers understand the meaning of the translations that they select or deselect. MIRACLE provides the following four types of cues to help searchers who know only English decide which of the available translations to use: (1) pronunciation, (2) synonyms, (3) examples of usage, and (4) translation probabilities.

Cebuano and English are written using the same character set, and colonial influences have resulted in the adoption of numerous “loan words” from English and Spanish. Although the written form of these words may differ somewhat from the typical spelling in English, they can often be recognized visually by English speakers, and in other cases sounding out the spelling can help to recognize an English word from which a Cebuano word might have been derived. Hindi and English are written using different character sets, but again colonial influences have resulted in the presence of loan words from English in Hindi documents. In this case, however, phonetic transliteration is needed to allow speakers of English to reconstruct the pronunciation of Hindi words. We used a locally developed transliteration scheme similar to ITRANS[1] for this purpose. We found that searchers were sometimes able to sound out this transliteration and recognize a corresponding English term, although visual recognition of the related English term was typically not possible.

We extracted English-to-Hindi translation probabilities from the translation lexicon (described below in Section 4) and displayed them graphically in a bar-chart format. Our search techniques give greater weight to more likely translations. Searchers who understand that fact might use this cue to focus their efforts on the translations that the system believes are most probable

[1] www.aczone.com/itrans.

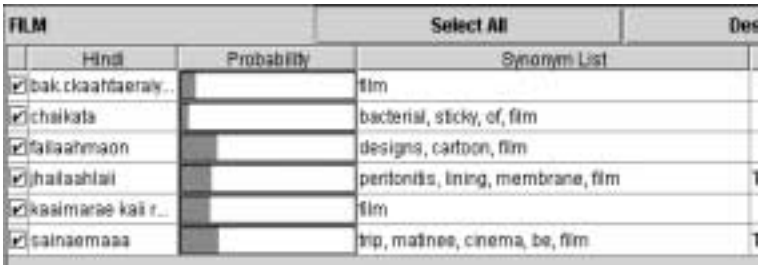


Fig. 2. Synonym list of "film."

since deselecting an improbable translation would have little effect on the retrieval results. This illustrates an interdependence between the introduction of new capabilities and the development of new strategies; if we are to leverage this capability, we will likely need to include some form of embedded training (e.g., "tool tips" on mouseover) in future versions of MIRACLE.

Ideally, we would prefer to provide the searchers with English definitions for each Hindi translation alternative. Dictionaries with these types of definitions do exist for some language pairs, and indeed we were able to obtain one for Hindi in electronic form. Bilingual term lists are, however, much more widely available, and translation lexicons that include probabilities (typically derived from parallel text) would not be expected to come with associated definitions. In the remainder of this section, we describe the two types of cues that can help to fill this gap.

### 3.1 Synonym List

The simpler of our two techniques attempts to construct a set of synonyms for each translation using round-trip translation (what we call "back translation" into English). For example, Figure 2 shows several Hindi translations for the English word `film`. The Hindi word `jhailaahaii`[2] has four known English translations: `film, peritonitis, lining, membrane`. From this it seems clear that `jhailaahaii` corresponds to the use of `film` in a "thin layer" sense. The Hindi word `sainaemaaa` also has four known English translations: `film, trip, matinee, cinema`. This seems to clearly correspond to the "movie" sense of `film`. The situation is not always this clear, of course, since some Hindi words will have only one known English translation, and in such cases that word is always the original query term (since we use the same term list in both directions). Moreover, many of the Hindi translations will themselves have multiple senses; detecting a reliable signal among the noisy cues provided by back translation therefore sometimes requires commonsense reasoning. Fortunately, that is a task for which humans are uniquely well suited.

Since the original query term is always present as a back translation, including it in what we call the "synonym list" might seem unnecessary. Indeed, we initially omitted the original query term in an effort to simplify our user interface. We were persuaded to add it back, however, by several searchers who

[2]All Hindi words in this article are presented in our ITRANS-like transliteration.

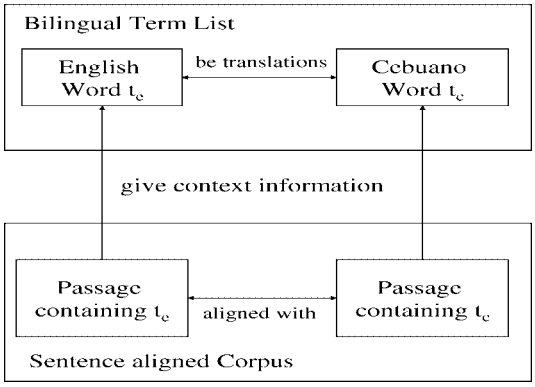


Fig. 3. Constructing cross-language KWIC using a sentence-aligned parallel corpus.

explained to us that the presence of that term made it easier for them to understand how that list is generated. This experience served to reinforce our view that supporting the development of mental models is an important factor in the design of interactive systems.

### 3.2 Examples of Usage

Examples of usage can provide complementary evidence about the meaning of an unfamiliar translation. This idea is often referred to as "KeyWord In Context" (KWIC) [Baeza-Yates and Ribeiro-Neto 1999]. For each Hindi translation of an English term, our goal is to find a brief passage in which the English term is used in a manner that is appropriate for the translation in question. For example, associating `jhailaahlaii` with the sentence "there is a film of oil on the water of this pond," and `sainaemaaa` with the sentence "the film now showing at the theater is very good" would make each sense clear. We have developed two ways of automatically finding such associations.

When a sentence-aligned collection of translation-equivalent ("parallel") text is available, identifying appropriate examples of usage is fairly straightforward. We obtained a Cebuano Bible in electronic form on the first day of the dry run, and the verse labels on Bible text make it fairly easy to align with an English Bible (in our case, the New International Version). Figure 3 illustrates the process that we used to identify an appropriate example of usage. Formally, let $t_e$ be an English term for which we seek an example of usage, and let $t_c$ be a known Cebuano translation of $t_e$ that is found in a translation lexicon. Let $S_e$ and $S_c$ be the shortest pair of sentences that contain $t_e$ and $t_c$ respectively. Then $S_e$ can be presented as the example of usage for translation $t_c$.

While examples of usage found using parallel text are generally correct and often informative, parallel text with the desired translation relationships might be hard to obtain. The New International Version of the Bible contains only about half the words (by type) that are found in typical modern news texts [Resnik et al. 1999]. During the Hindi surprise language experiment, even the Bible was initially unusable because of encoding differences between the available Hindi Bibles and our test collection. For such cases, we have developed a

second approach that relies solely on two types of resources that we already had in hand: a bilingual term list and a large collection of English news text that is comparable (on a topical basis) to the Hindi collection that we wish to search. The key idea of this "comparable corpus" technique is to use the translation relationships that we know from the bilingual term lists to help pinpoint appropriate examples of usage. Our technique leverages back translation, and it works only when a nontrivial set of back translations is available.

We used a sentence collection derived from English newswire stories used in a recent Topic Detection and Tracking evaluation (TDT-4). We broke the TDT documents into sentences, then filtered out sentences that were very short (less than 10 words) or very long (more than 50 words). This resulted a collection of 483,242 sentences. The computation proceeds as follows:

(1) Generate a pool of candidate examples of usage:
   —Given a query term $e$, one of its translations $h$, and a bilingual term list, find the set of $n$ back translations, denoted $\{bte_i | 1 \leq i \leq n\}$.
   —For each back translation $bte_i$, search the monolingual English text collection $E$ to obtain a set of sentences containing $bte_i$. The set is denoted as $H_{bte_i}$.
   —Merge $H_{bte_i}$, $1 \leq i \leq n$ to build a sentence pool $P$. $P$ is the pool of examples of usage for $bte_i$, and since we treat $e$ and $bte_i$ as synonyms, the pool can also be viewed as a pool of examples of usage pool for $e$. Of course, this pool is likely to also contain some inappropriate examples because some terms in $bte_i$ could have multiple meanings.
(2) Identify the English terms that best characterize that pool:
   —Remove English stopwords.
   —Calculate term weights as the product of the number of occurrences of the term in $P$ and the logarithm of the number of sentences in $E$ divided by the number of sentences containing the term in $E$.
   —Rank the terms in decreasing order of term weight and select the top 10 terms as the contextual term set of $e$ given the translation $h$, denoted as $CS_{e,h}$.
(3) Choose an example of usage:
   —Identify the sentences in $E$ that contain the query term $e$, denote this set $E_e$.
   —Using $CS_{e,h}$ as a weighted query and the sentences in $E_e$ as the documents, rank the sentences in $E_e$ in decreasing order of similarity to $CS_{e,h}$ and select the top-ranked sentence as the example of usage for $h$ given $e$.

Both approaches to generating examples of usage can be run offline, so these techniques need not be highly tuned for efficiency. It is not clear that sentences are the optimal context length for our English-only technique or that using the most frequent terms (after stopword removal) is the best approach, but brief inspection of the results seems to indicate that this first implementation of our idea often works. Figures 4 and 5, later in this article, show some of the examples that were found with this technique.

Since our primary focus in this work is on integration, we leave intrinsic evaluation of these techniques to future work. Our two techniques for generating examples of usage exploit different resources and exhibit complementary strengths, so there is no reason not to try both when the necessary language resources are available.

## 4. CROSS-LANGUAGE RANKED RETRIEVAL

The heart of our cross-language search process is automated ranking of documents based on the selected translations for each query term. In this section, we address three key questions: (1) what terms should be translated (root forms, words, or phrases)?; (2) what translations for those terms can be found and represented in a translation lexicon?; and (3) how should those translations be used to rank the documents [Oard and Diekema 1998]? Our goal is to build ranked lists with the greatest possible density of relevant documents near the top of the list, so we begin by explaining how we measured our progress as we tried different alternatives. We then describe some of the most interesting ideas that we explored for the three key questions identified above, organizing our presentation in roughly the order that we tried them for Hindi.

### 4.1 Formative Evaluation

Sparck Jones defines formative evaluation as assessment performed to support the development process; distinguishing it from summative evaluations intended to measure progress against target criteria [Sparck Jones and Galliers 1996]. The key resource for formative evaluation in information retrieval is a test collection consisting of queries, documents, and relevance judgments. One common way of generating a CLIR test collection is to start with a monolingual test collection that contains documents in the desired language and then translate the queries by hand into English. Unfortunately, we were not aware of any existing ranked retrieval test collections for Cebuano or Hindi. We, therefore, set out to build the test collections that we needed from scratch.

Queries and documents are relatively easy to obtain, the expensive and time-consuming part of creating a typical test collection is the relevance judgment process. Conventional techniques based on pooling the output of many systems could be (and were) used for summative evaluation of our Hindi systems, but adopting such a strategy for formative evaluation would pose a chicken-and-egg problem; until reasonable systems exist, pooled relevance assessment will not work. Exhaustive relevance assessment on a small collection would be one alternative, but we needed something that could be built in a few days, and exhaustive assessment on a collection of any size would take far longer than that.

Fortunately, we had some experience with a less expensive approach to evaluation of ranked retrieval systems based on known-item retrieval [Garofolo et al. 1997; Hackett and Oard 2000]. The key idea is to obviate the need for relevance judgments by constructing queries for which one (and hopefully only one) document is already known to be relevant. The natural measure of

effectiveness in such a case is mean reciprocal rank (MRR), which is the inverse of the harmonic mean across the query set of the position in a ranked list (counting down from the top) at which the known relevant document is found. This is equivalent to the more familiar uninterpolated mean average precision (MAP) measure when only a single relevant document exists for each query.

We were able to leverage existing resources to create a test collection for Cebuano by using verses from the Bible as the document collection and an online Bible quiz with a verse-based answer key (from `http://www.swefil.com`) as the source of queries and relevance judgments. We extracted 50 questions from this quiz, manually reformatting some of them in a manner more similar to typical TREC queries. This test collection proved to be useful, with our best Cebuano system at the end of the 10-day dry run achieving a MRR of 0.14, corresponding roughly to an average placement of the relevant passage at position 7 of the ranked list (i.e., often on the first page of results). Since verse numbers in the Bible are typically consistent across languages, we would expect that this collection could easily be reused for other languages.

Of course, this turned out not to be the case for the very next language that we tried, Hindi, because we initially lacked access to a Hindi Bible in an encoding that was compatible with any available bilingual term list. Fortunately, we rapidly found the one key resource that is needed to construct a known-item CLIR test collection from scratch: someone fluent in both Hindi and English. The Hindi collection that we used contained approximately 3,000 BBC Hindi news articles from 2001 that were encoded in UTF-8, the same encoding as some of the bilingual term lists that were available. Native Hindi speakers were available on the first day; they helped us to develop 19 search questions within the first 3 days, and then another 10 at the beginning of the second week. Additional queries were later contributed for the same test collection by other teams using the same methodology.

Native speakers read several documents, seeking to identify those for which a question could be crafted that would uniquely retrieve that document. The questions were recorded in English, and then filtered by people with expertise in information retrieval to remove questions that seemed not to reflect an information need that would occur naturally. For the benefit of teams that preferred MAP as a measure, some additional relevance judgment was also performed on the top-ranked documents from our early runs. Although we did find some additional relevant documents, MAP and MRR tended to rank the systems that we compared consistently. We, therefore, report MRR results below. This test collection was widely used for formative evaluation in the surprise language CLIR community, and was commonly referred to as "UMD BBC test collection."

It is important to note that MRR suffers from quantization noise that makes it difficult to reliably distinguish between systems unless their retrieval effectiveness is quite different. Apparent differences in MRR should therefore be treated as reliable only if the difference is fairly large. MAP suffers from the same deficiency when only a small number of relevant documents are known. Nevertheless, several teams were able to make effective use of this test collection as a basis for system tuning.

### 4.2 Translation Lexicons

Translation lexicons might be obtained in several ways. They can be found on the Internet, scanned from a bilingual dictionary and then constructed using OCR and extraction, or rekeyed manually from a printed bilingual dictionary. In the case of Cebuano and Hindi, we were able to obtain bilingual term lists from the Internet.

The first translation lexicon that we obtained for Hindi had been manually developed by the Indian Institute of Information Technology (IIIT) (`http://www.iiit.net/ltrc/Dictionaries/Dict_Frame.html`). It contained 25,000 translation pairs, each of which included an English sentence as an example of usage. In the rest of this article, we refer this as *the IIIT dictionary*. Some dictionaries rank the translations that they provide in the order of decreasing likelihood, but we were not able to determine whether this was the case for the IIIT dictionary. We, therefore, treated each alternative as equally likely.

Translation probabilities are well known to be useful for CLIR [Darwish and Oard 2003; Xu et al. 2001], so we drew on multiple sources of evidence to construct a translation lexicon that incorporated reasonable estimates of translation probability as soon as some parallel text became available. In addition to the IIIT dictionary, we obtained a list of Hindi translations for country names and the names of Indian cities and states from the Linguistic Data Consortium (LDC). We also used term-alignment counts from the EMILLE corpus (prepared at the University of California at Berkeley) and term alignments from multiple sources (prepared at the University of Southern California Information Sciences Institute (USC-ISI)). We combined these resources as follows:

—Merge the IIIT dictionary with a LDC list of location names and assign a uniform distribution across the known translations for each English term.

—Estimate translation probabilities from the raw term counts in each source of aligned parallel text by dividing the counts for a particular Hindi translation of an English word by the total count for that English word.

—Merge the resulting probabilities using a weighted sum, with a weight of 0.5 for the IIIT/LDC probabilities and 0.25 for each of the other two sources. This reflected our belief that appearance of a translation in the IIIT dictionary was a more reliable indicator of expected usage than presence in any single source of statistically aligned text would be.

The resulting translation lexicon is referred to as the *UMD combined probabilistic dictionary* in the rest of this article.

Near the end of the month, IBM made an additional source of statistical term alignments available. Used alone, the resulting translation lexicon was the best single translation resource for CLIR that was created during the surprise language exercise, yielding about a 10% relative improvement in MRR over that of the use of our combined dictionary. To obtain better coverage, we merged the IBM lexicon with the UMD combined probabilistic dictionary, weighting the IBM lexicon 0.6 and the UMD combined dictionary 0.4. The merged lexicon

yielded a slight improvement in MRR over the IBM lexicon alone. We call this merged translation lexicon the *UMD final probabilistic dictionary*.

### 4.3 Transliteration of Out-of-Vocabulary Terms

When using dictionary-based query translation for CLIR, it can be helpful to augment dictionary lookup with some means of processing out-of-vocabulary (OOV) terms (terms not covered by the dictionary) [Demner-Fushman and Oard 2003; Al-Onaizan and Knight 2002]. Backoff translation [Resnik et al. 2001] (using translations of terms that share a common stem) can be helpful, and MIRACLE incorporates that capability. When translation still fails, it can be helpful to simply keep the untranslated word. This often works well for named entities, which are typically not well covered by manually prepared dictionaries. We did this for Cebuano.

For Hindi, some form of transliteration was needed to accomplish the same task, since English words do not normally appear unchanged in Hindi text. For example, among 18 OOV terms from our first set of 19 English queries, 13 were named entities. Manually translating these OOV words and adding them to the translated queries raised the MRR dramatically, from 0.23 to 0.70. We, therefore, explored resolving OOV terms by transliteration, a common practice for person, location and organization names. First, a set of phones was produced for each English word by using the Festival text-to-speech system. Each phone was then replaced by its nearest Hindi "character." Often, more than one Hindi character is a reasonable match for an English phone; in such cases, the intuition of a native Hindi speaker was used to decide a preference order for possible replacements. To limit fanout, the number of Hindi character alternatives for any phoneme was limited to 3, and an upper limit of 16 overall transliteration alternatives for any English word was imposed. The transliteration alternatives were automatically sorted in an order that reflected the relative likelihood of each constituent mapping, but we did not make use of this ordering in our CLIR system. Use of transliteration with the full set of 93 queries improved MRR from 0.54 to 0.57, too small a difference for us to make any strong claims.

### 4.4 Weighted Structured Queries

It is common for an English word to have multiple Hindi translations. Pirkola's structured query translation method (in short, "Pirkola's method"), exploits the structure imposed by the translation process to help to limit the adverse effect of this translation ambiguity [Pirkola 1998]. Its key idea is to view multiple translations of a query term as "variants" of the query term, using the sum of the term frequency and the union of the document frequency across the set of translations to compute a weight for the query term in each document. Darwish and Oard extended this idea as weighted structured queries (in short, "Darwish's method") to accommodate the explosion of relatively unlikely translations that are often hypothesized by statistical techniques based on parallel text. Darwish's method replaces Pirkola's unweighted sum with a sum weighted by translation probability and approximates Pirkola's union operator

Table I. Comparing Word-Based CLIR Approaches Using the UMD BBC Collection, UMD Combined Probabilistic Dictionary, and 19 Queries

| All (Darwish) | One-best (by probability) | All (Pirkola) | One-best (by RATF) |
|---|---|---|---|
| 0.378 | 0.373 | 0.203 | 0.137 |

Darwish's method used PSE, others used InQuery.

with a second weighted sum. When reliable estimates of translation probabilities are available, Darwish's method yields retrieval effectiveness equal to that achieved by well-tuned implementations of Pirkola's method, but with far less sensitivity to the way in which translation probability thresholds are chosen [Darwish and Oard 2003].

We used the University of Massachusetts Inquery system (version 3.1p1) to implement Pirkola's method and Darwish's Perl Search Engine (PSE) to implement Darwish's method. PSE is a vector space system based on Okapi BM25 term weights. We used a hexadecimal ASCII representation of UTF-8 to avoid problems with the processing of character codes for which these systems were not originally designed.

Before translation probabilities became available, we had also tried using a relative average term frequency (RATF) technique that had been proposed by Kwok [Kwok 1996; Pirkola et al. 2002], finding that Pirkola's method yielded better retrieval effectiveness. We, therefore, chose Pirkola's method as our baseline. When the probabilities in the UMD combined probabilistic dictionary became available, we actually observed a marked decline in the retrieval effectiveness of Pirkola's method due to the addition of a large number of low probability translations. Without a separate collection on which to tune a probability threshold, we lacked a principled way of selecting which translations to use. Pirkola's method was actually beaten by one-best translation (chosen using translation probabilities) under these conditions, and RATF demonstrated an even more marked decline. Darwish's method yielded results similar to one-best translation in this case, suggesting that our translation probability estimates were not yet as well tuned as they might be at that point. Table I summarizes these results.

### 4.5 CLIR Using Character *n*-Grams

Indexing overlapping character *n*-grams offers an attractive alternative to word-based indexing in rapid development scenarios because *n*-grams can capture semantically meaningful constituents of words without the need for stemming or decompounding tools that may not be available early in the development process. Indeed, *n*-gram techniques have been previously tried for monolingual Hindi retrieval [Natrajan et al. 1997]. Unfortunately, neither Pirkola's method nor Darwish's method can be used when *n*-grams are indexed because both rely on a known mapping between English query terms and the Hindi terms that are indexed; no such mapping is known for Hindi *n*-grams.

We have previously explored the use of balanced translation with post-translation resegmentation for use in such cases [Meng et al. 2000]. For Hindi we tried a simple-variant of this approach that uses one-best translation

Table II. Comparing Character *n*-Gram-Based CLIR with Pirkola's Word-Based Method Using the UMD BBC Collection, IIIT Dictionary, 19 Search Topics, InQuery

| 2-gram | 3-gram | 4-gram | 5-gram | Word (Pirkola) |
|---|---|---|---|---|
| 0.15 | 0.32 | 0.31 | 0.29 | 0.23 |

Table III. The Effect of Phrase Translation Using the UMD BBC Collection, IBM Probabilistic Dictionary (UMD Cleaned Version), and 93 Queries

| InQuery | | | PSE | | |
|---|---|---|---|---|---|
| Plain query translation without phrase translation | | | | | |
| One-best | Pirkola | 3-gram | One-best | Darwish | 3-gram |
| 0.453 | 0.435 | 0.428 | 0.449 | 0.454 | 0.415 |
| Plain query translation with phrase and component translation | | | | | |
| 0.460 | 0.448 | 0.489 | 0.451 | 0.491 | 0.478 |

One-best selected based on translation probability; 3-grams also used one-best translation.

selected based on the translation probability. Our early experiments with a set of 19 English topics and the IIIT dictionary showed that retrieval with UTF-8 character 3-grams, 4-grams, or 5-grams was better than the best of word-based retrieval under the same conditions (Pirkola's method). Among different *n*-grams, character 3-grams seemed to perform the best (see Table II). Our later experiments with 93 topics and the IBM probabilistic dictionary showed that CLIR with character 3-grams, when coupled with phrase translation (described below), could achieve retrieval effectiveness comparable to the best of word-based retrieval under comparable conditions (Darwish's method). Table III summarizes these results.

### 4.6 Phrase-Based Query Translation

Multiword expressions (which we refer to as "phrases") typically exhibit far less translation ambiguity than their constituent words, and that reduction in translation ambiguity can have beneficial effects on retrieval effectiveness in CLIR applications [Ballesteros and Croft 1998]. We observed little benefit from phrase translation in our early experiments, perhaps because our 29 queries included only three English phrases that were present in our translation lexicon. We tried again with the 93 queries that became available towards the end of the month, and in that case we observed a consistent improving trend from the use of phrases across a number of CLIR techniques. Table III shows these results of that later experiment.

## 5. DOCUMENT AND PASSAGE TRANSLATION

Reading translations can support four tasks in an interactive cross-language search process.

*Selecting documents to examine*: Brief passages extracted from several highly ranked documents are normally shown to help searchers recognize documents that deserve closer examination. Searchers can also sometimes assess

the quality of the overall result set by examining those passages; if no highly ranked documents are noted, that could indicate a need for a shift in their search strategy.

*Recognizing relevant documents*: The goal of the retrieval process is to find relevant documents, and our experience indicates that searchers are often able to accomplish this task using imperfect translations [Wang and Oard 2001]. If the translation quality is not sufficient to use a translated document directly, it may then be necessary to arrange for fluent human translations of the selected documents. In such cases, the search process can be thought of as focusing the more expensive human translation process on the most promising documents.

*Learning vocabulary*: If searchers are to find documents based on the terms that they contain, they must learn how to express concepts in ways that will ultimately match those terms. Observing an unexpected term in a translated document can help searchers acquire vocabulary that they might use to improve subsequent search iterations.

*Learning concepts*: As Belkin and others have observed [Belkin 1980], searchers often do not really know what they are looking for at the outset of a search session. In such cases, they learn as they go by selectively reading parts of documents that they find, using what they learn to refine their understanding of their topic. Reading machine-prepared translations for understanding can be difficult, even with state-of-the-art systems, so at present this is probably best accomplished by first searching in a language that the searcher can easily read, switching to cross-language search only after the nature of the information need is clear.

In our interactive CLIR research we have started at the top of this list, working down it as we gain additional insight into the requirements. In the remainder of this section, we describe the support that MIRACLE provides for each of the first three tasks. Where possible, we leverage the rich resources available in English to permit rapid development of capable systems, even early in the process when resources available in the other language may be sharply limited.

### 5.1 Selecting Documents to Examine

The brief summaries displayed in a ranked list are typically intended to be "indicative" (saying what a document is about). One common strategy in monolingual systems is to present a few excerpts from regions of the documents that contain several query terms. Our Hindi MIRACLE system incorporates a similar technique, selecting as many as three excerpts from the best available translations that contain terms closely related to the query terms (not including query terms that are stop words). We use the Porter stemmer as a measure of term similarity for this purpose; terms present in a translated document that share a common stem with any query term are selected as candidates. Excerpts that cover multiple query terms are preferred. Among excerpts covering a single query term, those that appear earlier in the document are preferred, a heuristic that is known to be useful when summarizing news stories. At present we make no use of information content measures such as inverse document frequency

for this purpose, but that is an alternative to position-based cues that we plan to explore in future work. Each excerpt contains 17 words (eight on either side of the query term), so the maximum length of a summary built in this way is 51 words. Terms in the excerpt that match any query term are highlighted (in red, although this color could easily be made user-selectable).

We also tried two other types of brief summaries. As a simple baseline, we used the first 40 words from each document; this was the only technique available for Cebuano. MIRACLE is normally used to search news, and it is common for news articles to start with an overview of the story, so this is a fairly strong baseline. The query-based excerpts that we implemented for Hindi seem to us to be a clear improvement, although we have not yet performed a user study to substantiate that claim.

As a third alternative, we have also integrated a capability to display the automatically generated English headlines described by Dorr et al. [2003] in this volume. Headlines are more compact than extractive summaries, and they are designed to be informative rather than indicative, but our present headlines are intended to be general rather than query-specific. Because the headlines are not query-specific, we could generate them in advance and cached the results. Online generation of headlines would be possible with our present techniques (which require translating only a few sentences), but that capability is not yet implemented. Between the time of the Cebuano and the Hindi experiments, we conducted a user study (with Spanish) to compare automatically constructed headlines with use of the first 40 words from news stories; we found that searchers were much faster when using headlines, although somewhat less accurate [Dorr et al. 2003].

### 5.2 Recognizing Relevant Documents

MIRACLE incorporates two translation techniques: (1) modular use of an existing translation system and (2) term-by-term gloss translation. In prior work, we have observed that users are better able to recognize relevant documents using a state-of-the-art machine translation system than when using gloss translation [Wang and Oard 2001], so the modular use of machine translation is preferred when it is available. For Cebuano, gloss translation proved to be essential since full machine translation only became available after the 10-day dry run. For Hindi, we were able to incorporate gloss translation immediately; full machine translation became available at about the midpoint of the 29-day exercise after encoding differences in some of the available sources of parallel text had been resolved. From this experience, we conclude that gloss translation is most useful early in the development process, when it provides only the practical support for the task of recognizing relevant documents.

The key idea in gloss translation is simply to replace each term with one or more possible translations. When multiple translations are shown, we found in prior work (with Japanese) that people are often able to identify the right meaning from context [Oard and Resnik 1999]. MIRACLE, therefore, incorporates two layout options to support this human ability. With the horizontal option, alternate translations are grouped using parentheses and listed in

order of decreasing likelihood. With the vertical option, alternate translations are stacked vertically in decreasing order of likelihood. In each case, the most likely translation is rendered in a bold font. The horizontal option is more space efficient, so it is the default for document display. The vertical option is, in our opinion, somewhat easier to skim; therefore, it is the default for the display of summaries that are intended to support document selection. When translation probabilities are not available (as is typically the case before statistical machine translation systems have been built), we use the term frequency in a comparable English collection as a surrogate for translation probability (i.e., more common English terms are treated as more likely translations).

The key resource for gloss translation is a bilingual term list. When a document-language term is found in the term list, its English translations are shown. If no translation is found, the document-language term was displayed unchanged, but in a distinguishing color (by default, blue) to indicate a translation failure. Because Cebuano and English use the same character set, untranslated words often turned out to be informative in that case, typically when they were names or domain-specific terminology. For Hindi we displayed untranslatable terms using our locally developed ITRANS-like transliteration. It proved to be very hard to pick out such terms by eye from among the relatively large number of untranslatable Hindi terms; we therefore assessed gloss translation to be far less useful for Hindi than it had been for Cebuano.

When statistical machine translation became available for Hindi from USC-ISI, we cached translations for the entire document collection. This made it possible to continue our development process without close coupling to the translation server. This proved to be an adequate approach for the relatively small UMD BBC collection; scaling up to the larger (40,000-document) evaluation collection was possible only with several days of effort. For high-volume operational applications, we will clearly need to integrate on-demand translation. An on-demand translation service did ultimately become available from USC-ISI, and incorporating that capability into MIRACLE should be straightforward.

### 5.3 Learning Vocabulary

For Hindi, statistical machine translation sometimes proved to be sufficiently fluent to help the searcher identify potentially useful query terms. In the present MIRACLE design, the searcher must manually type these new terms into the query. This process leads to two potential problems, both of which actually occurred in the brief user study described below. First, it was sometimes the case that the bilingual term list being used for query translation lacked the new English term. We tried to mitigate this effect by incorporating translations from the same parallel text alignment process that was used to construct the machine translation system. Second, even when the English term was known, the bilingual term list would often bring in translations other than the one that had resulted in the English term that the searcher had observed in a translated document. From this experience, we have concluded that it would be useful to add a function to MIRACLE for incorporating specific document terms into a revised query in a way that preserves translation

relationships. Doing so will require somewhat closer integration with the translation system, since we will need access to not just translations, but also term-level alignments.

## 6. INTEGRATION AND INTERFACE DESIGN

We built MIRACLE using a Java client–server architecture in order to balance easy integration of component technologies (on the server side) with rich interaction in a portable framework (on the client side). Extensive logging functions are provided on the server side to support use of the system for user studies. Our primary goal for MIRACLE is to evaluate interaction strategies, so processing is done offline whenever possible in order to minimize the need for a focus on run-time efficiency at this early stage in our development process. This design decision made it possible to routinely integrate new capabilities in a single day during both the Cebuano dry run and the Hindi surprise language exercise.

The user's understanding of MIRACLE's capabilities is shaped by the user interface. Our interface design was guided by two key design guidelines: (1) expose our interaction design to the user in a straightforward and easily understood manner and (2) provide immediate feedback in response to control actions. Both guidelines are intended to contribute to our overarching design goal, to support the progressive refinement of mental models that can contribute to improved search effectiveness.

### 6.1 The MIRACLE Interface

As shown in Figure 4, the MIRACLE interface consists of three major components: a query input panel, a translation selection panel and a result browsing panel. Searchers type their queries in the query input panel, just as they would in a monolingual Web search engine. At present, only unstructured ("bag of words") queries are supported, although users can specify phrases in addition to individual words. English stopwords are removed prior to query translation. We received several requests to support Boolean queries during design reviews with expert searchers; clearly, this would be a desirable additional capability for some types of users.

When the searcher clicks the "search" button, the system obtains all translations for each (nonstopword) query term from the translation lexicon, and makes them available for display in the selection summary area at the left side of the translation selection panel. A query term that could benefit from disambiguation (i.e., has two or more translation alternatives) is automatically select for expanded display on the right side of that panel. When more than one such term exists, the one with the fewest translation alternatives is automatically chosen. This simple heuristic is based on the observations that common terms, which are less useful as a basis for search, typically have many more translations than more selective terms. Search results are also displayed immediately (based, initially, on the use of all known translations), with brief summaries of the top 10 documents displayed on the first page of the document browsing panel. The searchers may then elect to examine individual documents, view additional result pages, deselect some translations for any query term and search


Fig. 4. MIRACLE CLIR system for Hindi.

again using the same query, or revise their choice of query terms by typing in a new query and reinitiating the search.

### 6.2 Translation Selection Panel

The translation selection panel (shown in Figure 5) includes a tree display of the full query context on the left and a tabular display of selection cues for a single selected query term on the right. The top nodes of the overview tree represent the current and previous queries. The descendents of these nodes contain query terms, and the descendents of each query term are its known translations. Translations that have been deselected by the searcher are displayed in a faded font. One term in the present query is automatically selected for tabular display, but the searcher may change that by clicking on any term in the translation tree for the present query.

The selected query term is shown at the top of the tabular display in a bold font to help the searcher maintain an understanding of the context of their selection actions. Each row of the table represents one translation alternative for the selected query term, and a checkbox is placed at the front of each row to allow searchers select or deselect that translation (all checkboxes are initially selected). The translation (transliterated if necessary) appears in the first column, in this case under the title “Hindi.”

The second column graphically depicts the translation probability associated with each alternative as a bar with a length proportional to the probability. The remaining columns show the back translations and examples of usage described in Section 3. The table layout can be adjusted by dragging the column

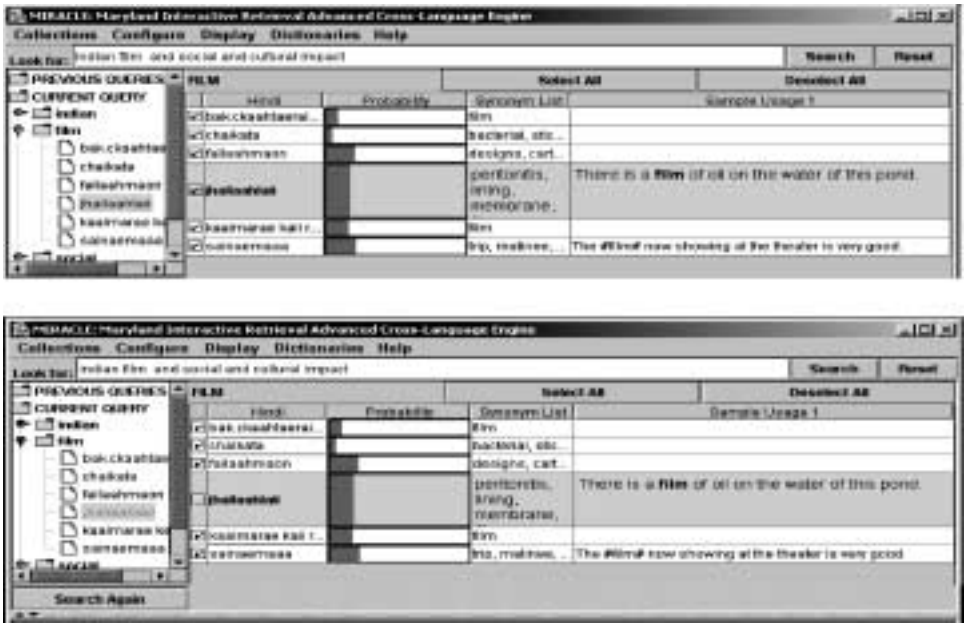

Fig. 5. Translation Selection in the Hindi MIRACLE system. Top: before deselection, bottom: after deselection.

separators, allowing searchers to devote more screen space to whichever types of explanations they find most useful. Clicking in a cell results in animated vertical expansion of that cell, providing sufficient space to display all available explanations simultaneously.

The translation selection panel requires a substantial amount of screen real estate, thus limiting the number of search results that can be examined at a glance. We therefore provide a small button on the divider bar to toggle between display of the translation selection panel and full-screen display of the document browsing panel. When the translation selection panel is hidden, the MIRACLE interface resembles that of a typical monolingual Web search engine.

### 6.3 Document Browsing Panel

The document browsing panel provides an overview of the results from a single search iteration (see Figure 6). Summaries of the translated documents are shown, ranked in decreasing order of similarity between the corresponding document and the searcher's query, as is typical in monolingual applications. The panel displays 10 summaries per page; although the user may need to scroll to see all ten if the translation selection panel is displayed. A total of ten result pages are available (using the "next" button), so the searcher could conceivably examine as many as 100 document surrogates. Dividing the result set in this way facilitates rapid delivery of search results when network bandwidth is limited. As described above, terms that share a common stem with any query term are highlighted (in red) to draw the searcher's eyes.

Fig. 6. Document selection and relevance judgments in MIRACLE CLIR system for Hindi. The pop-up window at the center is the full document view window. The pop-up window at the bottom left corner is the judgment history window.

Each summary is labeled with a numeric rank (1, 2, 3, ...), which is displayed as a numbered button to the left of the surrogate. The full text of the associated document can be viewed by clicking on the button. In order to maintain context, we repeat the numeric rank and the summary at the top of the document examination window. The pop-up window in the middle (in Figure 6) is a document examination window.

MIRACLE supports the optional collection of relevance judgments from the searcher, a useful capability for some controlled user study designs. Three degrees of relevance can be indicated (Not relevant, Somewhat relevant, and Highly relevant). A fourth value, "?" (indicating unjudged), is initially selected by the system. Similarly, three degrees of confidence in the relevance assessment can optionally be indicated (Low, Medium, or High), with a fourth value ("?") being initially selected by the system. Searchers can record relevance judgments and confidence values in either the document browsing panel or the document examination window (when that window is displayed). We also record the times at which documents are selected for examination and the times at which relevance judgments for those documents are recorded. This allow us to later compute an (approximate) examination time for each document.

To help maintain context during iterative refinement, MIRACLE can optionally present a pop-up window to show the judgment history. Previously judged documents are listed in the window, and they can be arranged by the

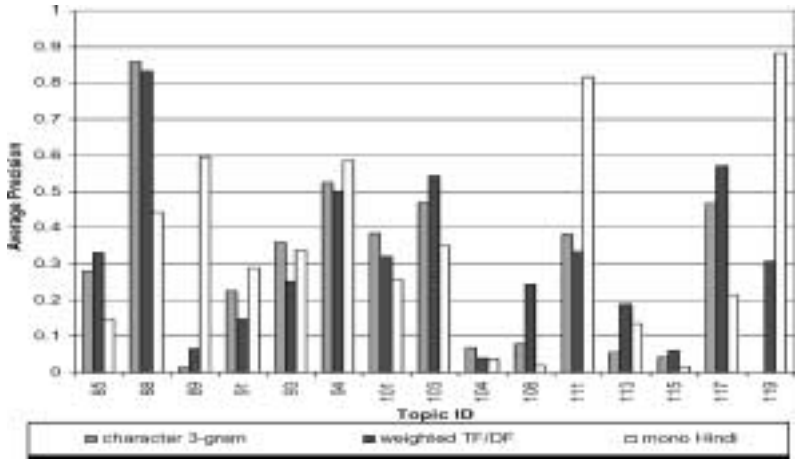


Fig. 7. Summative evaluation results, as reported by NIST, uninterpolated average precision by topic. For one-best 3-grams, word-based weighted structured queries, and monolingual Hindi.

user-assigned degree of relevance or by the user-assigned confidence in their relevance judgment. This capability was originally designed for use in controlled user studies (where explicit judgments of relevance are a part of the task), but it could also prove useful to search intermediaries who seek documents on behalf of other users.

## 7. EVALUATION

To test the effectiveness of the Hindi systems implemented during the surprise language exercise, the National Institute of Standards and Technology (NIST) and the LDC organized a small summative evaluation effort at the end of the month. For CLIR evaluation, 15 TREC-style topic descriptions and a Hindi document collection containing 41,697 documents drawn from several sources were used. Pooled relevance judgment was performed using the top 20 documents from selected runs.

We submitted two automatic CLIR runs using techniques that proved to be among the most effective in our formative evaluation process. We used the same queries for both runs, combining all terms in the title, description and narrative fields of the topic description. Both runs also used the UMD final probabilistic lexicon, a Hindi stemmer from the University of California at Berkeley, and a Hindi stopword list from the University of Massachusetts. One run used InQuery with one-best translation and 3-grams, and the other used PSE with word-based weighted structured query translation. The PSE contributed to the relevance judgment pools; the InQuery run did not. As Figure 7 shows, the two approaches achieved comparable retrieval effectiveness. A Wilcoxon signed rank test did not find a significant difference in the means of the uninterpolated average precision values between the two conditions (0.28 and 0.32 respectively).

We also submitted one automatic monolingual Hindi run to help establish a baseline for cross-language retrieval effectiveness measures on this collection; this run did not contribute to the judgment pools. Queries were formed in the same way as for the CLIR runs. The MAP in this case (0.34) was comparable to the better of the two CLIR runs, and Wilcoxon signed rank tests found no significant difference between the monolingual results and the results of either CLIR run. We did not optimize our monolingual system in any way (for example, we performed no blind relevance feedback), so this should probably be considered a relatively low baseline.

In order to assess our interactive system, we also submitted one "manual" CLIR run to NIST for evaluation. Ten-minute searches were performed by the second author of this article (who knows no Hindi) for each of the 15 topics. An automated timer was used, and written self-observation notes were recorded to note unexpected behavior, recommended improvements, and subjective reactions. At the time the experiment was run, translations were available for approximately 85% of the collection; the lack of an available translation was noted for the remaining documents where the translated summary would normally be displayed. Documents judged by the searcher to be relevant were marked as "highly relevant," no judgments were recorded for other documents, and judgment confidence was not recorded. The searcher was free to allocate time within the 10-minute period between query formulation, translation selection, and marking relevant documents. The goal of the search was to find as many truly relevant documents as possible in the available time, with a bias in favor of precision (to operationalize this guidance, the searcher imagined that someone would then have to pay for a professional human translation of each selected document). All documents marked by the searcher as relevant were added to the evaluation pools by NIST.

As shown in Table IV, it proved to be possible to obtain remarkably high precision in many cases (averaging 0.68 overall, and exceeding 0.85 for more than half of the topics). When interpreting these results, it is important to recognize that this searcher was unusually well prepared, with an intimate understanding of the system's design and capabilities, and far richer experience with the design and application of cross-language search strategies than most users would bring to similar task. Nevertheless, we believe that these results show that MIRACLE can support the task for which it was designed.

## 8. CONCLUSION AND FUTURE DIRECTIONS

In this article, we described the design of an interactive system that allows searchers who know only English to find relevant documents that are written in other languages. We have now demonstrated this capability for French, German, Cebuano, Spanish, and Hindi. The capabilities for Cebuano and Hindi were developed remarkably rapidly, reflecting both the easily adaptable design of our MIRACLE system and the unprecedented accomplishments of our colleagues around the world who rapidly developed the component technologies that MIRACLE incorporates.

Table IV. Interactive CLIR Results

| Topic ID | Retrieved | Relevant | Retrieved & Relevant | Precision | Recall |
|---|---|---|---|---|---|
| 085 | 1 | 45 | 0 | 0 | 0 |
| 088 | 23 | 37 | 19 | 0.83 | 0.51 |
| 089 | 13 | 53 | 7 | 0.54 | 0.13 |
| 091 | 3 | 11 | 3 | 1 | 0.27 |
| 093 | 7 | 49 | 3 | 0.43 | 0.06 |
| 094 | 18 | 55 | 14 | 0.78 | 0.25 |
| 101 | 9 | 46 | 8 | 0.89 | 0.17 |
| 103 | 11 | 60 | 5 | 0.45 | 0.08 |
| 104 | 6 | 65 | 0 | 0 | 0 |
| 108 | 9 | 43 | 8 | 0.89 | 0.19 |
| 111 | 2 | 23 | 2 | 1 | 0.08 |
| 113 | 8 | 27 | 7 | 0.88 | 0.26 |
| 115 | 16 | 40 | 10 | 0.63 | 0.25 |
| 117 | 16 | 62 | 15 | 0.94 | 0.24 |
| 119 | 9 | 11 | 9 | 1 | 0.82 |
| Avg | | | | 0.68 | 0.22 |

The second column is the number of documents that were marked as relevant by the searcher, the third column is the total number of relevant documents found in the judgment pools by LDC assessors, and the fourth column is the number of documents that were marked as relevant by the searcher that were also judged as relevant by LDC assessors.

We have learned a number important lessons about the design of interactive CLIR systems from this process, many of which have been mentioned above. But our experience has also influenced our thinking on three broader issues that we expect will help to shape our future work.

*Fully exploiting translation probabilities*: The utility of translation probability estimates in CLIR applications is indisputable, but it is not yet clear that we are making the best possible use of this information. We have only begun to seriously explore how translation models that are optimized for readability (where a single reading must be generated) can be adapted for search applications that can be designed to be far more tolerant of ambiguity.

*The primacy of parallel text*: Although MIRACLE can provide some degree of capability using only gloss translation, the overall capability of the system is more strongly influenced by translation quality than by any other factor. Before our experience with the Cebuano dry run and the Hindi surprise language, it was easy to be skeptical about whether enough parallel text to build usable statistical machine translation systems could rapidly be assembled for an unexpected language. Now it is easy to believe that this can often be done. This has serious implications for the design of our interactive CLIR systems; if we are likely to have parallel text in large quantities, then we should devote more of our energy to using that resource as effectively as possible.

*Usable systems can be built*: The CLIR community of the 1990s assembled a decade ago around a technical challenge: *given a query in one language, how best can we rank documents that are written in another*? This stands in marked contrast to the way in which CLIR research evolved in its first heyday, the 1970s: *given a group of documents in a language that the searcher cannot read, how best can we help searchers to find them*? The best answer at that time turned

out to be multilingual thesauri. Today, multilingual thesauri are used in many operational systems, while free-text ranked retrieval for CLIR has yet to achieve broad market penetration. Why? Perhaps because questions of usability have not yet received adequate attention. Our experience with MIRACLE is one part of our effort as a community to meet that challenge.

Our work with MIRACLE draws together many threads. From information retrieval, we know how to rank documents written in other languages. From machine translation, we know how to give the user an idea what those documents are talking about. From human–computer interaction, we know how to structure the interaction and the interface to support the development of useful mental models. But the real miracle is not in the pieces, it is the way the pieces come together. MIRACLE is as much a way of thinking as it is a system; the components exist to support a process, and that process should therefore shape the way we think about what our components must be able to do. Only by working together can we hope to ultimately accomplish what we have set out to achieve, to deliver useful technology that improves our ability to shape the information environment.

## ACKNOWLEDGMENTS

The authors are grateful to David Doermann for his help in rapidly assembling a Hindi document collection, to Franz-Joseph Och for providing the translations that we used, to the LDC for their critical role in assembling parallel text and translation lexicons for use by the community, and to our colleagues throughout that community for freely sharing their ideas and resources.

## REFERENCES

AL-ONAIZAN, Y. AND KNIGHT, K. 2002. Machine transliteration of names in Arabic text. In *Proceedings of ACL Workshop on Computational Approaches to Semitic Languages*.

BAEZA-YATES, R. AND RIBEIRO-NETO, B. 1999. *Modern Information Retrieval*. Addison Wesley.

BALLESTEROS, L. AND CROFT, W. B. 1998. Resolving ambiguity for cross-language retrieval. In *Proceedings of SIGIR98*. 64–71.

BELKIN, N. 1980. Anomalous states of knowledge as a basis for information retrieval. *Canadian Journal of Information Science 5*, 133–143.

DARWISH, K. AND OARD, D. W. 2003. Probabilistic structured query methods. In *Proceedings of 26th Annual International ACM SIGIR Conference*. 338–344.

DEMNER-FUSHMAN, D. AND OARD, D. W. 2003. The effect of bilingual term list size on dictionary-based cross-language information retrieval. In *36th Annual Hawaii International Conference on System Sciences (HICSS'03)—Track 4*. (Hawaii).

DORR, B., ZAJIC, D., AND SCHWARTZ, R. 2003. Cross language headline generation for Hindi. *This volume*.

DORR, B. J., HE, D., LUO, J., OARD, D. W., SCHWARTZ, R., WANG, J., AND ZAJIC, D. 2003. iCLEF 2003 at Maryland: Translation selection and document selection. In *Proceedings of CLEF'03*.

GAROFOLO, J. S., VOORHEES, E. M., STANFORD, V. M., AND SPARCK JONES, K. 1997. TREC-6 1997 spoken document retrieval track overview and results. In *Proceedings of TREC 97*. 83–91.

HACKETT, P. AND OARD, D. W. 2000. Comparison of word-based and syllable-based retrieval for Tibetan. In *Information Retrieval for Asian Languages Workshop*. Hong Kong.

HE, D., WANG, J., OARD, D. W., AND NOSSAL, M. 2002. Comparing user-assisted and automatic query translation. In *Proceedings of CLEF'02*.

KWOK, K. 1996. A new method of weighting query terms for ad-hoc retrieval. In *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Zurich, Switzerland, 187–195.

MARCHIONINI, G. 1995. *Information Seeking in Electronic Environments*. Cambridge Series on Human-Computer Interaction. Cambridge University Press, Cambridge, MA.

MENG, H., CHEN, B., GRAMS, E., KHUDANPUR, S., LO, W.-K., LEVOW, G. A., OARD, D. W., SCHONE, P., TANG, K., WANG, H.-M., AND WANG, J. 2000. Mandarin-English Information (MEI): Investigating translingual speech retrieval. Technical Report, Summer 2000 Workshop, Center for Language and Speech Processing, John's Hopkins University. http://www.clsp.jhu.edu/ws2000/final_reports/mei/ws00mei.pdf.

NATRAJAN, A., POWELL, A. L., AND FRENCH, J. C. 1997. Using N-grams to process Hindi queries with transliteration varions. Technical Report CS-97-17, Department of Computer Science, University of Virginia. ftp://ftp.cs.virginia.edu/pub/techreports/CS-97-17.ps.Z.

OARD, D. W. AND DIEKEMA, A. 1998. Cross-language information retrieval. *Annual Review of Information Science and Technology 33*, 223–256.

OARD, D. W. AND RESNIK, P. 1999. Support for interactive document selection in cross-language information retrieval. *Information Processing and Management 35*, 3, 365–382.

OGDEN, W., COWIE, J., DAVIS, M., LUDOVIK, Y., NIRENBURG, S., MOLINA-SALGADO, H., AND N., S. 1999. Keizai: An interactive cross-language text retrieval system. In *Machine Translation Summit VII, Workshop on Machine Translation for Cross Language Information Retrieval*.

PIRKOLA, A. 1998. The Effects of Query Structure and Dictionary Setups in Dictionary-Based Cross-Language Information Retrieval. In *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, Melbourne, Australia.

PIRKOLA, A., LEPPNEN, E., AND JRVELIN, K. 2002. The RATF formula (Kwok's formula): exploiting average term frequency in cross-language retrieval. *Information Research 7*, 2.

RESNIK, P., OARD, D., AND LEVOW, G. 2001. Improved cross-language retrieval using backoff translation. In *First International Conference on Human Language Technologies*. http://www.glue.umd.edu/∼oard/research.html.

RESNIK, P., OLSEN, M. B., AND DIAB, M. 1999. The Bible as a Parallel Corpus: Annotating the 'Book of 2000 Tongues'. *Computers and the Humanities 33*, 1-2, 129–153.

SPARCK JONES, K. AND GALLIERS, J. R. 1996. *Evaluating Natural Language Processing Systems, An Analysis and Review*. Springer, Berlin.

WANG, J. AND OARD, D. W. 2001. iCLEF 2001 at Maryland: Comparing word-for-word gloss and MT. In *Evaluation of Cross-Language Information Retrieval Systems: Second Workshop of the Cross-Language Evaluation Forum, CLEF 2001*, C. Peters, M. Braschler, J. Gonzalo, and M. Kluck, Eds. Darmstadt, Germany, 336–354.

XU, J., FRASER, A., AND WEISCHEDEL, R. 2001. TREC 2001 cross-lingual retrieval at BBN. In *Proceeding of TREC01*.

Received August 15, 2003; revised September 29, 2003; accepted October 24, 2003