美

Mandarin-English Information (MEI): Investigating Translingual Speech Retrieval

Helen Meng, Berlin Chen, Erika Grams, Sanjeev Khudanpur, Wai-Kit Lo, Gina-Anne Levow, Douglas Oard, Patrick Schone, Karen Tang, Hsin-Min Wang and Jian Qiang Wang

October, 2000

Abstract

This report describes Project MEI (Mandarin-English Information), one of the four projects selected for the Johns Hopkins University Summer Workshop 2000. Our research focus is on the integration of speech recognition and embedded machine translation technologies in the context of *crosslingual spoken document retrieval* (CL-SDR), also known as *translingual speech retrieval*. Project MEI advocates a *multi-scale paradigm,* i.e., the use of *both* words and subwords (Chinese characters and syllables) in the development of one of the first systems for English-Chinese CL-SDR. Experiments are based on the Topic Detection and Tracking Corpora (TDT-2 and TDT-3) provided by the Lingusitic Data Consortium.[1] English newswire stories from Associated Press and New York Times are used in their entireties (query-by-example) to retrieve related Mandarin news broadcast from Voice of America. Relevance judgements are based on the manual topic annotations, and the *mean average precision* (MAP) is used as evaluation metric. We describe our research findings and contributions in (i) multi-scale query formulation, (ii) multi-scale translation, and (iii) multi-scale retrieval.

[1] http://morph.ldc.upenn.edu/Projects/TDT/

1. Introduction

Massive quantities of multimedia content is becoming readily available in the growing global information infrastructures. While current Internet search engines preponderantly search on textual documents in a monolingual setting (i.e. queries and documents are in the same language), there is a strong demand for *cross-media* retrieval (i.e. queries and documents are in different media). In mid-February 2000, www.real.com listed 1432 radio stations, 381 Internet-only broadcasters, and 86 television stations with Internet-accessible content, with 529 broadcasting in languages other than English. There is ample information in the form of recorded speech – from conferences, meetings, speeches and news broadcasts. Monolingual spoken document retrieval applications are crossing the threshold of practicality, as evidenced by Compaq's SpeechBot [Moreno et al., 1999], where the queries are textual, and documents are in audio form. Research efforts exist in other languages as well, e.g. French, German, Italian[2] and Chinese [Wang et al., 2000] [Meng et al., 2000]. As the Internet user population grows globally, there is strong motivation for the development of *cross-lingual* and *cross-media* (CLCM) information retrieval technologies. CLCM retrieval enables the user to search for personally-relevant content in any language or medium. Potential applications of CL-SDR technologies include audio and video browsing and searching, automatic user alert upon detection of specified events, and automatic information routing. A study by Global Reach projected that by the year 2005 English and Chinese will be the predominant languages used by the Internet user population. These factors motivated our team to work on English-Chinese CL-SDR.

The inclusion of Chinese renders it important for us to consider the characteristics of the language and its implications to our task. Mandarin Chinese is phonologically compact – an inventory of about 400 base syllables provides full phonological coverage of Mandarin audio. In addition, an inventory of about 6,800 characters provide full textual coverage of written Chinese (in GB code). Each character is pronounced as a syllable, and there is a many-to-many mapping between characters and syllables. For example, the character 行 may be pronounced as /hang2/,[3] /hang4/, or /xing2/. On the other hand, a given tonal syllable may correspond to multiple characters. Consider the two-syllable pronunciation /fu4 shu4/, which corresponds to a two-character word. Possible homophones include 富庶, (meaning "rich"), 負數, ("negative number"), 復數, ("complex number" or "plural"), 覆述 ("repeat").[4] In addition, each word may contain one to several characters (with no word delimiter). Different

[2] In Text Retrieval Conference (TREC-7) http://trec.nist.gov

[3] These are Mandarin pinyin, the number encodes the tone of the syllable.

[4] Example drawn from [Leung, 1999].

character sequences create different meanings and there is ambiguity in Chinese word tokenizations. Consider the syllable string:

/zhe4 yi1 wan3 hui4 ru2 chang2 ju3 xing2/

The corresponding character string has three possible segmentations – all are correct, but each involves a distinct set of words:

這一晚 會 如常 舉行 (Meaning: It will be take place tonight as usual.)

這一 晚會 如常 舉行 (Meaning: The evening banquet will take place as usual.)

這一 晚會 如 常 舉行 (Meaning: If this evening banquet takes place frequently…)

As can be seen, the inventory of Chinese characters can create an unlimited number of Chinese words. The Chinese language presents unique complexities for our task of CL-SDR.

CL-SDR requires the integration of three key technologies: (1) speech recognition, for indexing the document collection; (2) machine translation, for crossing the query/collection language barrier; and (3) information retrieval, to search for relevant documents. English-Chinese CL-SDR is hampered by several prevailing problems, including:

(i) **Open vocabularies in translation and recognition** – Documents may contain OOV which are unknown to the speech recognizer. The OOV words are substituted by in-vocabulary words during recognition. These are indexing errors that downgrade retrieval performance. Additionally, there may be phrases or *out-of-vocabulary* words (OOV) which are unknown to the machine translation system, e.g. proper names or specialized terms. Failure to translate content-carrying query terms precludes them from contributing towards retrieval performance.

(ii) **Term selection** – both queries and documents contain content-carrying terms that are important for retrieval, and function words / terms that are less important. Term selection is an important procedure that can enhance retrieval performance. Query term selection is especially important if long queries, e.g. entire passages or news stories are used, as in the case of the MEI task.

(iii) **Multiplicity in translations** – Dictionary-based query translation often encounter words with multiple translations. Selection of topically-relevant translation alternatives is a research challenge.

(iv) **Speech recognition errors** – Speech recognition output is imperfect. Errors may be caused by OOV or acoustic confusions among in-vocabulary words. SDR needs to be robust towards recognition errors.

(v) **Ambiguity in Chinese word tokenization –** The Chinese word consists of one to several characters, without word delimiters. Much ambiguity exists in tokenizing a sequence of Chinese characters into words. In English-Chinese CL-SDR, word tokenization in the translated queries are dependent on the translation term list(s); while word tokenization in the indexed documents are dependent on the recognizer's vocabulary and language model. Tokenization mismatches between queries and documents affect word-based retrieval.

(vi) **Ambiguity due to Chinese homophones –** The Chinese character-to-syllable mapping is a many-to-many mapping. Homophones are words with the same (syllable) pronunciation. For a given a syllable sequence correctly recognized from a spoken document, the different homophones may be produced via lexical access. Some of these may be extraneous for retrieval.

As an illustration of the above we can consider the example in Table 1.1, which assumes that we have to handle a query about Iraq (which is a three-character word 伊拉克 in Chinese). If our speech recognizer contains the word 伊拉克 and indexes it correctly (see row 1 in Table 1.1), then we will achieve a word-level match. However, should 伊拉克 be an OOV for recognition, there is still a chance that our recognizer produces three mono-character words 伊. 拉 and 克 sequentially (see row 2 in Table 1.1). Under this situtation, we face a mismatch at the word level, but matches are fine in character or syllable space. Row 3 in Table 1.1 illustrates the case with a character substitution error, and the substitution is due to a homophone of the first mono-character word. This produces a mismatch in word space, a partial match in character space and a full match in syllable space. Row 4 indicates a case with tokenization error, and only the match in syllable space survives. Row 5 presents a case with a syllable recognition error, which only managed to maintain a partial match in syllable space. The last row is where there are two syllable substitutions, and hence no matches are preserved at all.

We surmise that the use of subwords may provide partial relief to the problems mentioned above. The use of overlapping Chinese character n-grams in retrieval may be robust to word-level mismatches due to ambiguous Chinese word tokenizations. The use of overlapping Chinese syllable n-grams in retrieval may be robust to word- or character-level mismatches due to ambiguity in Chinese homophones. Futhermore, syllable-based retrieval circumvents the OOV problem in Chinese spoken document indexing, because the compact syllable inventory can fully index Chinese spoken audio. However, we believe that the use of Chinese words complements the use of subwords in retrieval, because words contain lexical

knowledge which is conceivably important for precision. Consequently, in Project MEI, we advocate a *multi-scale paradigm* for English-Chinese CL-SDR which involves the use of *Chinese words*, *Chinese characters* as well as *Mandarin syllables*. Our research challenges are to address the problems for English-Chinese CL-SDR mentioned above.

Recognition	English "translation"	Syllable Pronunciations	Word	Char	Syl
伊拉克	"Iraq"	/yi la ke/			
伊 • 拉 • 克	"she" "pull" "gram"	/yi la ke/	X		
一 • 拉 • 克	"one" "pull" "gram"	/yi la ke/	X	Δ	
易 • 拉客	"easy" "solicit"	/yi la ke/	X	X	
易 • 拉 • 的	"easy" "pull" particle	/yi la de/	X	X	Δ
依賴 • 的	"rely" particle	/yi lai de/	X	X	X

Table 1.1 The problem of ambiguity in Chinese word tokenization and Chinese homophones in English-Chinese crosslingual spoken document retrieval. The example is based on the topic word "Iraq" (伊拉克). represents a match at the specified level, Δ represents a partial match, and X no match.

This report is organized as follows: we will begin by describing the MEI task, summarize previous work done in related areas, provide details about our experimental corpora, and then present our research contributions and experimental findings in (i) multi-scale query translation, (ii) multi-scale audio indexing and (iii) multi-scale retrieval. We have also included an analysis of our work as well as conclusions and future plans.

2. The MEI Task

Figure 2.1 depicts the overivew of the MEI task. The English textual query is translated into Chinese. The Mandarin audio is processed by automatic speech recognition for document indexing. The translated query and indexed audio are both fed into the retrieval engine which produces a ranked list of Mandarin spoken documents for each query. We have adopted a query translation strategy (as opposed to document translation), because it is computationally less intensive (document collections may be huge), and also our queries are textual and not susceptible to speech recognition errors.

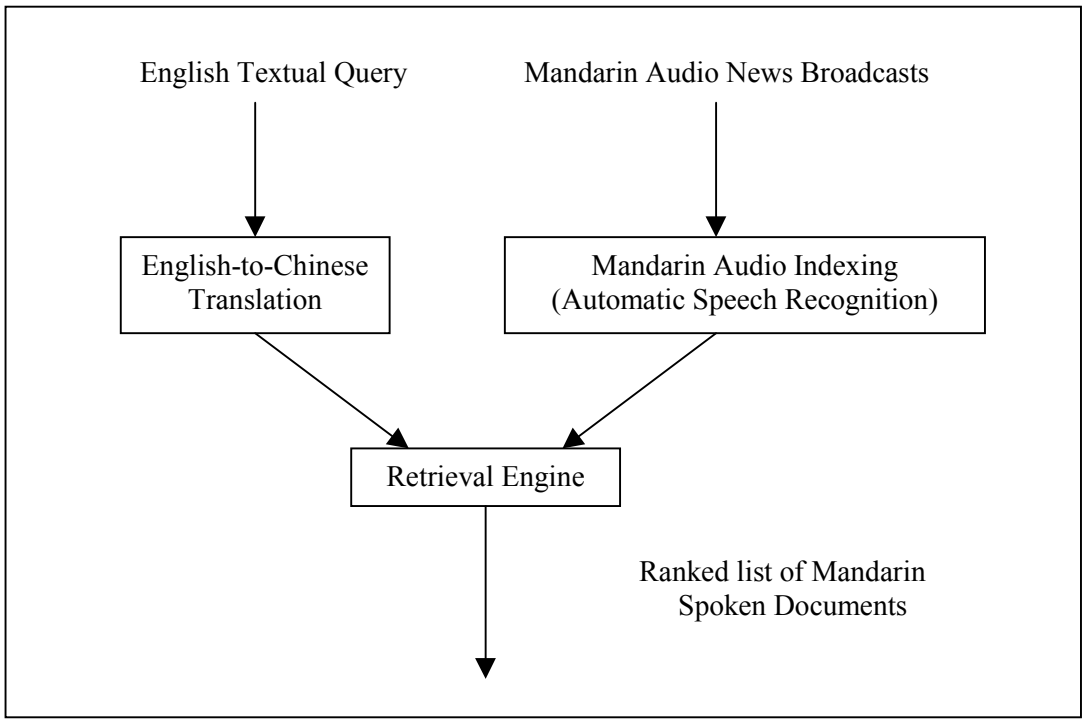


Figure 2.1 An Overview of the MEI Task.

We strive to advance the state of the art for the automated search component of interactive retrospective CL-SDR systems. We assumed a ranked retrieval paradigm, in which the system produces a ranked list of potentially useful documents and then the user interactively examines documents (or summaries), proceeding downward from the top of the list until either their information need is satisfied or their search is modified or abandoned. Ranked retrieval has proven to be popular in many applications, at least in part because it provides a simple way of guiding the user's attention, thereby leveraging human pattern recognition abilities that far exceed those of any automated process. We focused exclusively on building the ranked list, relegating the equally important user interface design problem to future work.

We chose to use a single exemplar as our statement of the information need, an approach known as query-by-example, because there is available corpora well suited to that research design. We used contemporaneous news stories in both cases, because that is what was contained in the available test collection that best fit our overall objectives. Manually determined story boundaries were available for the (otherwise unsegmented) audio news stories that we used, and we chose to use those known boundaries rather than to determine them automatically in order to better match the scope of our task with the available time and resources.

We chose as a measure of effectiveness a variant of the widely reported mean average precision metric. The mean average precision metric represents the expected density of relevant documents at the point where the user chooses to stop, where the expectation is taken over possible information needs and over possible stopping points. We extended the metric to additionally compute the expectation over alternative exemplars for the same information need. Our specific task was thus to produce a CL-SDR system that achieved the greatest possible mean average precision on a previously unseen test collection.

3. Previous Work

The MEI project draws on prior work in four communities: information retrieval, cross-language information retrieval, spoken document retrieval, and translingual topic tracking. In this section we describe each in turn, with particular attention to the systems and techniques that have directly benefited our work.

Research on support for retrieval of information using digital computers extends back to the 1950's. Our approach in the MEI project is in the ranked retrieval tradition, a major branch of the field in which a ranked list of potentially relevant documents is produced from which an interactive user could select the documents that they actually desire. This branch of the field has produced a widely used methodology for evaluating the effectiveness of ranked retrieval systems, and we adapted that methodology to our task in the MEI project. A detailed description of the methodology and the way in which we adapted it to the MEI task is presented below. Several information retrieval systems are now freely or inexpensively available for research use. By incorporating one of these systems (Inquery version 3.1p1 from the University of Massachusetts) into our architecture, we were able to exploit capabilities that would otherwise not have been practical to develop in the time we had available.

The past decade has seen a steady increase in the attention paid to a specialized subtopic known as Cross-Language Information Retrieval (CLIR) in which queries and documents may be expressed in different natural languages (e.g., queries in English, documents in Mandarin Chinese). The key feature of CLIR research has been integration of translation and retrieval technologies, and we made extensive use of CLIR techniques in our work. In some cases (e.g., dictionary-based machine translation), we adapted software that we had originally written for the CLIR track of the Text Retrieval Conference (TREC) series. In other cases (e.g., cross-language phonetic mapping) we tried new ideas to address issues that have been identified but not yet fully explored by the CLIR research community.

Another emerging subtopic that we drew on for the MEI project was research on Spoken Document Retrieval (SDR). The key feature of SDR research has been integration of Automatic Speech Recognition (ASR) and retrieval technologies. Again, we made use of both software that we had written for earlier research (e.g., Mandarin Chinese syllable lattice recognition) and our understanding of the open research issues in the SDR community (e.g., integrated indexing at both word and subword scale).

The final subtopic of information retrieval that informed and supported our work in the MEI project was topic tracking, one of five tasks in the Topic Detection and Tracking (TDT) evaluation series. The key feature of the topic tracking task in the 1999 TDT

evaluation was the requirement to integrate techniques from the TREC CLIR, SDR and Filtering tracks into a single system. We made use of two important resources from the 1999 TDT evaluation in the MEI project: the development test and evaluation collections (described in detail below), and linguistic resources (e.g., a bilingual term list) that had been assembled for use by participants in that evaluation.

Of course, we built on a broad array of other resources as well -- most notably from computational linguistics (e.g., the Identifinder named entity tagger from BBN) and from statistics (e.g., computing statistical significance tests using Matlab). And none of what we did would have been possible without the excellent computing facilities and support at the Center for Speech and Language Processing at Johns Hopkins University. But the research communities identified above, information retrieval, CLIR, SDR, and topic tracking, together provide both the principal intellectual heritage for the work reported here and the most important of the existing tools and techniques that we used to explore some important and interesting questions over the course of a single six-week workshop.

4. Our Experimental Corpora – the TDT Collection

Without question, the most important existing resource that we exploited were the TDT collections. Unlike the TREC collections, in which retrieval experiments are based on manually prepared topic descriptions, the design of the TDT collections are models a query-by-example task in which the user presents one or more exemplar documents to illustrate their information need. We used two TDT collections, the TDT-2 collection for development testing and the TDT-3 collection for evaluation. Both collections consist of documents from multiple sources in two modalities (speech and character-coded text), and documents in both modalities are available in two languages (English and Mandarin Chinese). We used a subset of each collection that consisted of a set of English exemplar documents for each topic from the Associated Press newswire and/or the New York Times that were represented in the Latin-1 (ISO-8859-1) character set and a set of Mandarin Chinese audio documents from the Voice of America that were to be searched. The English exemplars were drawn from the specified sources with only minimal restructuring to produce a consistent format; the Mandarin Chinese audio documents were extracted from a continuous audio stream through manual insertion of story boundaries and manual removal of non-story segments such as advertisements. This processing was performed by the Linguistic Data Consortium (LDC) for the TDT evaluations, and our effort in this regard was limited to the automatic processing required to use these resources with the Inquery information retrieval system. The English portion of the TDT-2 collection spans the period from January through June of 1998, with the Mandarin Chinese portion covering only March through June of 1998. Both the English and Mandarin Chinese portions of the TDT-3 collection span the period from October through December of 1998. There are 2,265 stories to be searched in the TDT-2 Mandarin Chinese audio and 3,371 stories to be searched in the TDT-3 Mandarin Chinese audio.

Figure 4.1 summarizes the statistics for the part of the TDT collections that we used. English examplar documents are available as character-coded text for 17 topics in the TDT-2 collection and for 56 topics in the TDT-3 collection. In TDT, documents are defined to address the same topic (and thus to be relevant to that topic) if their creation can be ascribed to the same event. For example, a document that described the TWA flight 800 crash and another document that described the investigation of the cause of that crash would be considered to be relevant to the same topic, but a third document that described the crash of a different airplane would not be relevant to that topic. The manually prepared relevance assessments that are distributed with the TDT collections specify the relationship between each topic and each document as YES (relevant), BRIEF (topically related, but lacking a substantial treatment of the topic), or NO (not relevant). English exemplar documents that were marked as BRIEF were not used, but because a document might be marked as BRIEF

with respect to one topic and YES with respect to another, we retained Mandarin Chinese audio documents that were marked as BRIEF in our collection. The effect of these documents was, however, subsequently removed from the retrieval effectiveness computation using a procedure that is described below.

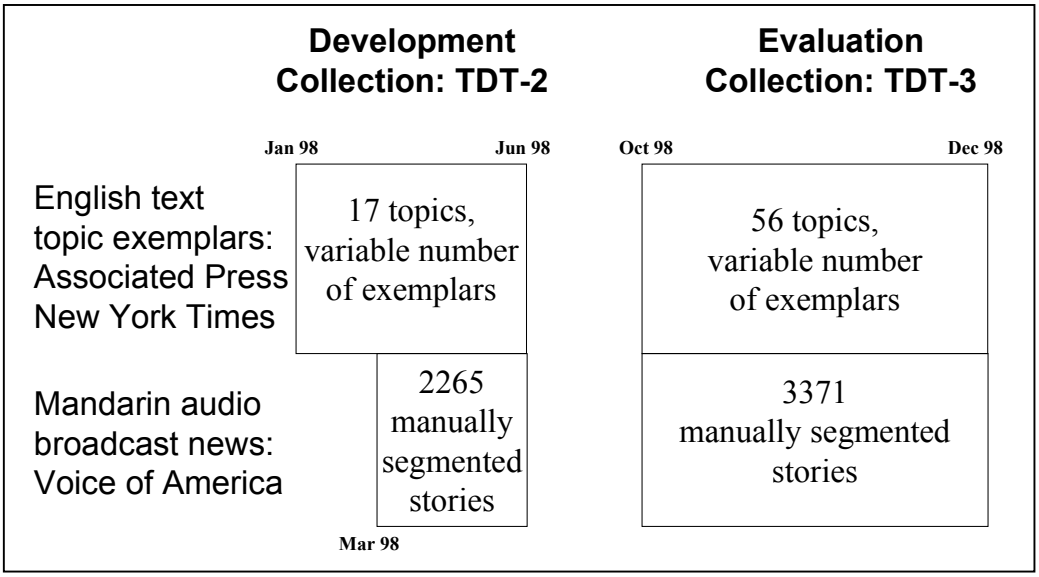


Figure 4.1 The part of the Topic Detection Tracking (TDT-2 and TDT-3) corpora that are used in the MEI experiments.

Two characteristics of the TDT collections were particularly important for our work: they include the only Mandarin Chinese audio collections for which relevance judgments are already available with respect to a standard set of topics, and at the time of our work they formed the only paired CLIR evaluation collections in which development testing and evaluation could be performed on separate document collections. The query-by-example structure of the TDT collections necessitated only relatively minor changes to the evaluation methodology that we chose, so that aspect of the TDT collections was suitable for our purpose. The relatively small size of the Mandarin Chinese audio subcollections does serve to somewhat limit our ability to extrapolate our results to applications in which far larger collections will need to be processed, but this is not an inappropriate limitation for the sort of exploratory research that we sought to conduct in the MEI project.

5. Evaluation Metric

Since our focus in the MEI project was on the automated search component of a cross-language speech retrieval system rather than on the associated interactive components (interactive query formulation and interactive document selection), we chose an evaluation methodology designed to evaluate component-level performance in ranked retrieval applications. For the purposes of component-level evaluation, the input is given, and the goal of the system is to produce a ranked list with the greatest utility for some model of the downstream task of interactive document selection. The most widely reported single-valued figure of merit for the utility of such a ranked list is mean uninterpolated average precision, which is computed as follows in the TREC evaluations:

$$\beta_{\text{TREC}} = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{n_i} \sum_{j=1}^{n_i} \frac{j}{ri,_j},$$

where m is the number of topics, n_i is the total number of number of documents that are relevant to document i, and $r_{i,j}$ is the position (rank) of the jth document that is relevant to topic i, counting down from the top of the ranked list. Since we have multiple exemplars for each topic in the TDT collection, we modified this measure as follows for the MEI project:

$$\beta_{\text{MEI}} = \frac{1}{m} \sum_{i=1}^{m} \frac{1}{e_i n_i} \sum_{k=1}^{e_i} \sum_{j=1}^{n_i} \frac{j}{r_{i,j,k}},$$

where e_i is the number of exemplars for document i and $r_{i,j,k}$ is the rank of the jth relevant document that is relevant to topic i in the ranked list that is computed for exemplar k. This measure has a useful intuitive explanation that helps to ground the component evaluation in a broader model of the interactive information retrieval task:

- We assume that the user will begin scanning the ranked list at the top and will stop after they have examined whatever number of relevant documents that they desire to see.
- We assume that the user's satisfaction will be related to the density of relevant documents that they have experienced between the time that they started scanning and the time that they terminate their scan. The inner ratio expresses this value, which is known as "precision."
- Since we assume that the user will stop scanning after some relevant document but that we don't know which one, we compute the expected value of precision over all possible stopping points, assuming a uniform distribution on possible stopping points. This measure is known as "uninterpolated average precision."

- Since we are do not know which exemplar document will be presented, we compute the expected value over all exemplars for a topic. This is the step that is unique to our adaptation of the principal TREC effectiveness measure for use in a query-by-example context, and the resulting value does not have a widely used name.
- Since we are unable to predict what topic the user will be interested in at some future time, we take the outer expectation over a set of randomly selected topics and use that expectation as an estimate of system effectiveness on previously unseen topics. This measure is known as "mean uninterpolated average precision," or simply as "mean average precision."

The number of exemplar documents varies considerably across topics. When more than twelve exemplars were available for a topic, we randomly selected twelve exemplars (without replacement) because we felt that this would give us a sufficiently close estimate of the true expected value over exemplars while limiting the computational resources required to evaluate each system configuration that we wished to try.

6. The Perfect Retrieval Myth

When evaluating information retrieval results, we are always faced with the problem of determining how well our systems work. As mentioned previously, mean average precision (mAP), which combines knowledge about precision and recall, has been adopted as our evaluation metric. Perfect retrieval produces a mean average precision of 1.0, often quoted as 100%, which means that the retrieval system has not only found ALL *n* relevant documents in a collection, but that it also ranked them in the top *n* positions in the retrieved list.

Perfect average precision cannot happen in most systems because of mismatches between query data and document data. By undertaking a set of experiments to remove these sources of corruption, and then gradually reintroduce them, we sought to establish various "upper bounds" for mAP in the MEI system. This allowed us to match our experimental results on actual data against estimates of what would occur in a best possible situations: documents retrieving themselves, optimal monolingual retrieval, and optimal translingual retrieval.

In the MEI system, data mismatches can be introduced in the query processing side, the document processing side, and in the process of retrieval. Figure 6.1 shows the different pieces of the MEI system and in italics, where data corruption can be introduced into the system.

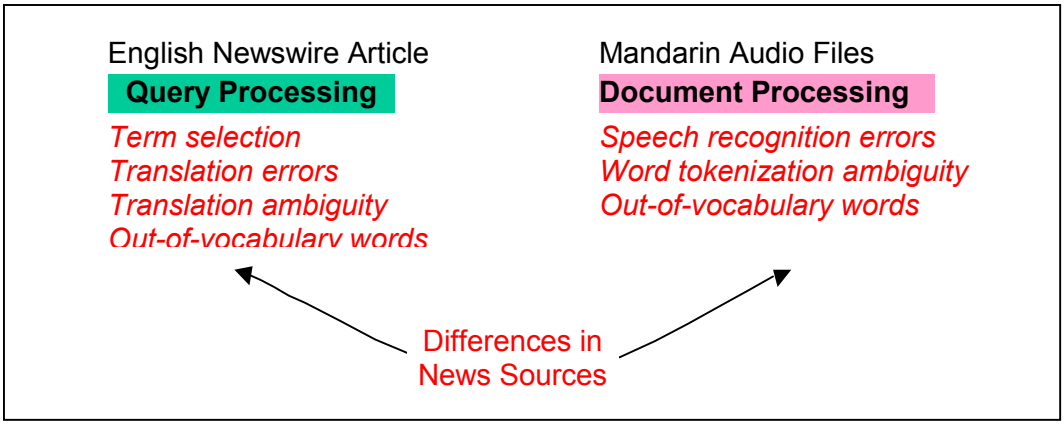


Figure 6.1 Processes in CL-SDR which may degrade retrieval performance.

The whole purpose of retrieval is to match a query in one form against a set of documents which are in a different form, in an attempt to retrieve those which are most similar to the query. Naturally, those differences can complicate the task of finding the similar documents. This is exemplified in the MEI system. On the query side, we are using stories written for the New York Times and the Associated Press, while the documents are written in Voice of America style. These media sources do not report the news in the same way, using the same

style and vocabulary. At this point, we have no method for converting from one writing style to another to make retrieval easier. However, we can overcome some of the problem due to language and modality differences.

In order to perform a usable retrieval experiment, the MEI System must transform the queries and documents into the same language, modality, and format. In practice, this means everything has to be textual and in the same language. The mismatch problem arises because these transformation processes introduce new corruptions into the system that can worsen retrieval results. In query processing, our system translates English text into Chinese text. Mismatches between the queries and documents can occur due to term selection, translation errors, translation ambiguity, and the existence of out-of-vocabulary (OOV) words. The document processing side requires that a Chinese audio stream be transformed into text. Problems can occur because of syllable or word recognition errors, tokenization ambiguity, and the existence of OOV words.

6.1 VOA Queries Rretrieving VOA Documents – A Cheating Upper Bound

Our first benchmark experiment sought to eliminate almost all of these possible errors. By taking a VOA Mandarin document and using it as a query to retrieve itself and its cohorts in the VOA Mandarin collection, we would avoid the corruptions introduced during query processing and the problems created by retrieving between different news sources. Corruptions due to document processing would still exist, but their impact was minimized because both the query and the documents being searched had been processed using the exact same speech recognition system.

For each of the 17 topics in the TDT-2 data, we randomly selected 3 relevant VOA Mandarin Dragon-recognized documents and divided them into 3 batches.[5] We then ran these query batches against our VOA Mandarin document set. The overall average precision was 0.733; this became our *highest upper bound* as it represented the best possible result we could possibly obtain – the query and documents are as similar as possible. In a corollary experiment, we used these same 3 query batches to retrieve from VOA Mandarin Anchor Script documents (i.e. approximating documents with perfect speech recognition), and achieved a non-statistically significant[6] different result at 0.739 overall average precision.[7]

[5] Since 2-3 of the topics only had 2 relevant documents in the VOA Mandarin collection, this experiment required some query duplication across three batches for those topics.

[6] According to a paired t-test at a p-value of 0.01.

[7] We did not have time to run an experiment with queries gathered from the VOA Anchor Scripts against the VOA Anchor Scripts document collection. That would have produced another interesting upper bound.

6.2 Xinhua Queries Retrieving VOA Documents – A Monolingual Upper Bound

The second series of experiments modeled the traditional monolingual problem. We benchmarked our system by avoiding corruptions due to translation from English to Chinese, but retained the problem of different news sources and the possibility of speech recognition errors in the document collection. The TDT-2 collection contains contemporaneous Mandarin news articles from the Xinhua news agency that could be considered comparable to the NYT/AP articles we planned to use when training and testing our system.

As in our previous experiment, we randomly selected 3 Xinhua articles for each of the 17 topics and divided them into 3 batches. Since written Chinese is not tokenized into words, we had to segment the Xinhua articles into words in order to use our word-based retrieval system. We took a dictionary of words known to be in the lexicon of the Dragon speech recognition system and then used the maximum-matching algorithm to segment the Xinhua stories. Since we did not have Dragon's language model, we could only approximate the type of segmentation their system would have produced in this way. When using these Xinhua batches against the Dragon-recognized documents in the TDT-2 collection, we recorded an overall average precision of 0.535. Additionally, using these Xinhua queries against the VOA Mandarin Anchor Scripts, we achieved an overall average precision of approximately 0.587. This result represents what we think we could achieve if we have almost perfect speech recognition and the best possible translations and thus became a *realistic monolingual upper bound* in performance.

6.3 Manual Query Term Selection and Translation – A Translingual Upper Bound

The third set of experiments allowed us to move into the translingual realm. We used a set of 17 randomly selected TDT-2 document exemplars as queries, with each of them corresponding to a TDT-2 topic. We only used one exemplar in this case, as opposed to three in the previous benchmarks, because subsequent manual processing is expensive. The default χ^2 term selection procedure[8] is applied to select up to 180 terms from each query. These are referred to as *long queries*. One of our team members then reads each query exemplar, and selected up to 4 terms which he deemed most informative to form a *short query*. The experimented was also allowed to select as many query terms as he wanted to form *a medium query*. These three query sets were fed into our dictionary-based query translation (DQT)

[8] This will be described in the next section.

routine, which provide up to 5 translations per term. The experimenter also manually corrected for translation insertions (i.e. the use of translations that are irrelevant to the context) and translation deletions (i.e. failure to cover appropriate translations). Multiple translations for each term were grouped with InQuery's synonym operator (#SYN) for retrieval. Retrieval performances are tabulated in Table 6.1.

Row	Description	Long Queries	Medium Queries	Short Queries
1	Average query length in English terms[9]	4	7	223
2	Average query length in translated Chinese terms (up to 5 translations per term)	10	16	571
3	Mean Average Precision **without** manual translation term selection	0.239	0.269	0.294
4	Average query length with manually selected Chinese translation terms	5	8	338
5	Mean Average Precision **with** manual translation term selection	0.417	0.499	0.512

Table 6.1 Benchmark performance with manual query term selection and translation term selection. This provides a translingual benchmark where we attempt to minimize performance degradations due to automatic query term selection and dictionary-based query translation. Rows 2 and 3 can be compared with rows 4 and 5 respectively.

Hence benchmark mean average precision in this experiment was 0.512 for VOA Mandarin Dragon-recognized documents, with "perfect" term selection and translation by manual processing. 0.469 for the VOA anchor scripts. Details of this set of experiments are provided in Appendix A.

6.4 Performance Benchmarks for Subword-based Retrieval

Most retrieval systems are optimized for term-based retrieval since, especially in English, that tends to produce the best results. However, one of the goals of the MEI project was to look at subword-based retrieval so we also benchmarked for performance upper bounds with character and syllable n-gram indexing in both the monolingual and translingual settings. Using the Xinhua and New York Times query batches created in the previous two experiments, we tested the system's performance with character and syllable unigrams,

[9] This is the average number of tokens per query. This number can be greater than 180 because in the term list, some terms which appear multiple times in the original query exemplar will be counted multiple times.

bigrams, and trigrams. Character and syllable bigrams significantly outperformed unigrams and trigrams in the experiments; since most words in Mandarin are composed of two characters, this makes perfect sense. Table 6.2 shows the overall average precision for the bigram runs against the VOA Mandarin Dragon-recognized documents.

Queries	Indexing Units	Mean Average Precision
Xinhua	Words	0.535
	Character bigrams	0.543
	Syllable bigrams	0.517
New York Times	Words	0.512
	Character bigrams	0.566
	Syllable bigrams	0.497

Table 6.2 Benchmark results for subword-based retrieval. Spoken documents are indexed with Dragon's recognition outputs. The monolingual upper bounds are provided by using the Xinhua News stories as queries. The translingual upper bounds are based on manually selected query terms and their translations from New York Times exemplars.

7. Multi-scale and Translingual Query Processing

This section describes our techniques in multi-scale and translingual query processing. Since we are working in a query-by-example paradigm, we begin with an English text exemplar story, with which we plan to select like documents from the speech collection. We use a dictionary-based query translation technique to bring the English text query into Mandarin "space" for matching. First we identify all named entities, such as people, places, and organizations using the BBN Identifinder system. This tagging process identifies items that are likely to be highly selective for information retrieval but also potentially difficult to translate. Next we identify the appropriate terms in the exemplar to use to form a query for retrieval. We extract multiword units, "phrases", as well single words and tagged named entities. We then rank these terms by their predicted utility or selection, based on a variety of statistics that we will describe in detail later. Now we perform dictionary-based term-by-term translation to produce a Mandarin query to present to the information retrieval system.

Query processing should maintain compatibility with our multi-scale document indexing procedure, which will be described in detail in the next section. Our audio document collection is transcribed by speech recognition technologies, including large-vocabulary continuous speech recognition by Dragon to produce a sequence of Chinese words. Indexing also takes place on a subword scale using the MEI syllable recognizer. Both the query and document representations were processed by the INQUERY 3.1pl information retrieval engine from the University of Massachusetts.

Query processing in MEI involves translation from English to Chinese. The following is a description of these procedures. We apply a dictionary-based translation approach, replacing each source language term with its target language counterparts in a bilingual term list.

7.1 Bilingual Term List

This key resource is formed by merging entries from the LDC's English-Chinese bilingual term list and entries created by inverting the Chinese-English Translation Assistance (CETA) file. The LDC's Chinese-English bilingual term list is a freely available resource produced by collecting English-Chinese translation resources from the World Wide Web. It is thus an inherently on-line resource intended for computational use. The CETA file, in contrast, was hand-constructed by a team of linguists from a collection of over 250 text bilingual and monolingual sources. In its original form, it contains Chinese words and their English translations. We selected entries from a twenty lexicon subset of the source, primarily from

contemporary general purpose or political-economic domains to produce the bilingual term list in the English-Chinese direction.

The term list is quite large, with almost 200,000 total English terms corresponding to almost 400,000 translation pairs. Detailed statistics appear in Table 7.1. In addition to single word terms, approximately 40% of the terms, 25% of the translation pairs, are multi-word, phrasal terms. Use of these larger units of meaning can lead to more precise, less ambiguous translation as in the example below. Although both "human" and "right(s)" have many translation alternatives, there is a single translation for the phrase "human rights." (see Table 7.2).

Total English Terms	199,444
Total Translation Pairs	395,216
Phrasal Terms	81,127
Phrasal Translation Pairs	105,750

Table 7.1 Statistics of our bilingual term list used for English-to-Chinese translation.

Term	No. of Translations
Human	7
Rights(s)	20
human rights	1

Table 7.2 Number of translation alternatives for the terms "human", "rights" and "human rights", indicating the importance of phrase-based translations.

7.2 Query Term Selection

As we reformulate the English text exemplar document into a query to the Chinese document collection indexed by the information retrieval system, we must identify the terms to translate as query components. We need to identify the relevant units for translation, as well as select from among those terms the ones that will be most useful as query terms.

The simplest unit size for translation would be white-space delimited words. However, the ambiguity reduction provided by multiword units above suggests a larger unit for translation. There are two sources of multiword units that we avail ourselves of. First all the English text exemplars have been tagged using BBN's Identifinder to delimit and label named entities in the text. The three main classes of tagged named entities are: name expressions, time expressions, and numeric expressions. The name expressions include names of people, locations, and organizations. For example, "partners of Goldman, Sachs, and Co." is an example of a person name expression, while "U. N. Security Council" is an example of an organization name expression tagged by the system. Time expressions include dates and time of day expressions. Numeric expressions include amounts of money and percentages. We

use the information about the phrasal coherence of these named entities both to identify them as candidate elements to translate and to determine how to translate them.

After named entities have been extracted as translatable units, we identify dictionary-based "phrases" in the text. These phrases correspond to multiword units in the source language in the bilingual term list and thus can be treated as single units for translation. We identify these terms through a simple automatic process, passing over the text left-to-right greedily gathering up the longest string, starting with the current position, that has an entry in the bilingual term list. This approach captures terms such as "Wall Street","best interests", "guiding principles", and "human rights." All of these terms are less ambiguous in translation than their component words.

Finally, we must select from among these terms those that will be used in the query. First we excluded all stopwords, based on the English default stopword list used by Inquery. Then we ranked all of the terms in the exemplar and all single word components of multiword units according to how well they distinguish the exemplar from other contemporaneous (and hopefully not relevant) stories. We used a χ^2 test in a manner similar to that used in [Schuetze et al., 1995] to select these terms. The pure χ^2 statistic is symmetric, assigning equal value to terms that help to recognize known relevant stories and those that help to reject the other contemporaneous stories. We limited our choice to terms that were positively associated with the known relevant training stories. For the χ^2 computation, we constructed a set of 999 contemporaneous documents in the English collection from which the exemplars were drawn.

7.3 Query Term Translation

Now we traverse the tagged English text exemplar and, for each identified term, if it is on the list of selected terms, we translate it. This approach preserves term frequency information and some ordering information in the query. For tagged named entities, we first attempt to translate the entity as a single unit by lookup in the bilingual term list. If the named entity is not found, we translate the individual words one by one. For example, "security council" is present in the bilingual term list and can be translated directly; "First Bank of Siam", however, is not present and is translated term by term. We also perform some special processing to handle numeric expressions and all other digit strings in the text, using information from the named entity tag, when available, and punctuation, when available. As a result, "12:30" would have the appropriate temporal translation, rather than that of the number 1,230.

All other terms are translated directly by search in the bilingual term list. Our experience with the TREC-8 CLIR track suggested that morphological analysis of terms

contained in documents and bilingual term lists could discover plausible translations when no exact match is found. We thus developed a four-stage backoff strategy that was designed to maximize coverage while limiting the introduction of spurious translations:

1. Match the *surface form* of a document term to *surface forms* of source language terms in the bilingual term list.
2. Match the *morphological root* of a document term to *surface forms* of source language terms in the bilingual term list.
3. Match the *surface form* of a document term to *morphological roots* of source language terms in the bilingual term list.
4. Match the *morphological root* of a document term to *morphological roots* of source language terms in the bilingual term list.

The process terminates as soon as a match is found at any stage, and the known translations for that match are generated.

7.4 Unbalanced, Balanced and Structured Queries

The translation process above replaces all translated terms with all of their target language translations found in the bilingual term list. We must therefore formulate the actual query to the information retrieval system in a way that appropriately incorporates and weights these alternative translations. Three candidate methods for incorporating these multiple translations are unbalanced queries, balanced queries, and structured queries.

Unbalanced queries simply replace all terms with all of their translation alternatives, This approach is perhaps the most common in dictionary-based cross-language information retrieval. However, this technique has some serious deficiencies. By replacing each term with all its translations in a standard vector space retrieval system, one effectively weights each term by the number of translations it has, distorting the statistics on which retrieval is based. Under this approach, a highly ambiguous term carries much more weight than an unambiguously translated term. Furthermore, common terms tend to be more ambiguous than uncommon ones. Thus, unbalanced query formulation can degrade performance by distorting the frequency statistics on which information retrieval depends.

We apply two techniques, structured and balanced query formulation, to address these problems with weighting. In balanced translation, the translations are treated as a single unit - a “pseudo-term” - with a weight that is the average over the number of translations of the

weights of the translations. This approach by default treats all translations as equally likely, though a weighted average could be applied based on translation probabilities.

Structured translation, as proposed in [Pirkola 1998], also treats the set of translation alternatives as a pseudo-term; however, the weighting strategy is more complex than that for balanced translation. Structured translation aim to treat all translation alternatives as synonyms and equivalents of the source language term. This approach recomputes, on the fly, the weight of the pseudo-term based on the term and document frequencies of the individual translations. Specifically the term frequency of the pseudo-term is the sum of the term frequencies of the translations. The document frequency is the document frequency of the union of the translations. This approach is more computationally expensive than balanced translation.

7.5 Results

We compared a variety of translations strategies. We contrasted word-based translation with phrase based term-by-term translation. We also evaluated our increase in effectiveness with an expanded set of named entities drawn from the Web and improved digit translation as outlined above. We performed a direct comparison of balanced and structured query formulation. Results are shown in Figure 7.1 below.

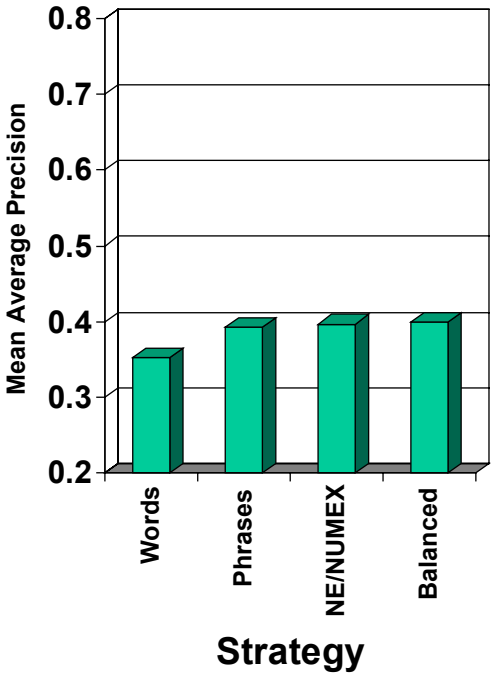


Figure 7.1 Comparative performance of various strategies in query translation. Experiments are based on the TDT-2 corpora. Strategies include word-based translation (first bar), word-based and phrase-based translation (second bar), word / phrase-based translations augmented by verbalized numeric expressions, balanced translations in queries.

We found significant improvement for phrase-based translation over word-based translation. A small improvement was obtained by improved digit and named entity translation. Finally we observe that structured and balanced translation achieved comparable effectiveness. Balanced query formulation has the advantages of faster processing and a direct extension to subword retrieval.

7.6 Translation Limitations

To better understand the limitations of this dictionary-based query translation approach, we consider the translation coverage of the corpus by our bilingual term list. We found that, of the approximately 87,000 term token to be translated, 3.5% were untranslatable. This amount corresponds to about 9% untranslatable terms by type. More detailed statistics appear in Table 7.3.

Terms	Count (by token)	Count (by type)
Total	87,004	12,402
OOV	3,028	1,122

Table 7.3 Translation coverage of our bilingual term list. OOV abbreviates out-of-vocabulary words.

We further inspected the untranslatable terms to try to characterize the out of vocabulary terms in the documents. We found that the vast majority of these terms are proper names, typically person names, such as “suharto”, “netanyahu”, “starr”, “arafat”, etc. These untranslatable terms are particularly problematic since these types of terms – named entities – are highly selective and likely to be very useful for information retrieval. This deficiency in the bilingual term list motivates our efforts in *subword transliterations*, through cross-lingual phonetic mapping, described in the next section.

7.7 Subword Transliteration – Crosslingual Phonetic Map (CLPM)

Subword transliteration is another research contribution in the MEI project. As mentioned previously, the main motivation is to salvage "untranslatable" terms for retrieval. These are often named entities e.g. names of people, places and organizations, etc. which are generally important for retrieval. As will be described in the next section, our multi-scale document indexing represents documents in terms of both words and subwords, hence the phonetic transliteration of subword units is useful for retrieval. We made use of *crosslingual phonetic mappings* derived from English and Mandarin pronunciation rules for this purpose. Chinese

translations of English proper nouns may involve semantic as well as phonetic mappings. For example, "Northern Ireland" is translated as 北愛爾蘭 — where the first character 北 means 'north', and the remaining characters 愛爾蘭 are pronounced as /ai4-er3-lan2/. Hence the translation is both *semantic* and *phonetic*. When Chinese translations strive to attain phonetic similarity, the mapping may be inconsistent. For example, consider the translation of "Kosovo" – sampling Chinese newspapers in China, Taiwan and Hong Kong produces the following translations:
科索沃 /ke1-suo3-wo4/, 科索佛 /ke1-suo3-fo2/, 科索夫 /ke1-suo3-fu1/,科索伏 /ke1-suo3-fu2/, or 柯索佛 /ke1-suo3-fo2/.

As can be seen, there is no systematic mapping to the Chinese character sequences, but the translated Chinese pronunciations bear some resemblance to the English pronunciation (/k ow s ax v ow/). In order to support retrieval under these circumstances, the approach should involve approximate matches between the English pronunciation and the Chinese pronunciation.

We have designed a subword transliteration process as illustrated in Figure 7.2. The spelling of a named entity may be derived from the spelling of a Chinese name, e.g. Wang Jiang Qiang and Wang Hsin-Min. In order to handle these cases, the spelling is segmented with a maximum-matching algorithm according to the inventory of Chinese syllables expressed either in the Pinyin convention or Wade Giles convention. For non-Chinese names, we attempt to look up its English pronunciation from LDC's pronunciation dictionary (PRONLEX), and failing that, we pass the spelling into a spelling-to-pronuncation generation system. This system is trained with 85K words from PRONLEX, using the transformation-based error driven learning approach [Brill 1994]. Benchmarking with a disjoint test set of 4.5K from PRONLEX shows a phoneme accuracy of 82% and a word accuracy of 45% in spelling-to-pronunciation generation. For example (see Figure 7.2), the name "christopher" produces a pronunciation of /kk rr ih ss tt aa ff er/. We have written a set of crosslingual phonological rules to transform the English phonetic structure into the Chinese syllable structure. This involves getting rid of consonant clusters and syllabifying postvocalic consonants. Hence /kk rr ih ss tt aa ff er/ becomes /kk ax rr ih ss ax tt aa ff er/. Then we applied our crosslingual phonetic mappings (CLPM) for transformation to a sequence of Chinese "phones", e.g. /k e l i s i t uo f u/. Our CLPM are trained from a corpus of 4800 English-to-Chinese transliterations,[10] where English phones are aligned with Chinese phones using a finite-state transducer. We expanded the sequence of Chinese phones into a phone lattice by referencing a confusion matrix, derived from running the CLPM on its training set.

[10] These include a partial list provided by Professor Hsin Hsi Chen from National Taiwan University, and other words that we have obtained from the Web.

A syllable bigram language model is then applied to search the lattice and produce a Chinese syllable sequence, e.g. /ji li si te fu/ or /ke li si tuo fu/ in Pinyin. This last step resembles lexical access in speech recognition. Should a character bigram be used, we can derive character sequences which are transliterations of the original English named entity, e.g. 克里斯托弗 and基里斯特弗.

As will be seen in our experimental results, CLPM provides a small but consistent performance improvement across different indexing units.

7.8 Multi-scale Query Construction

The input to our query construction process is a bag of selected English query terms. Multi-scale query constructure aims to integrate the translated phrases, named entities, verbalized numeric expressions, individual translated terms as well as transliterated syllables. Hence the output of our query construction process is a representation which include Chinese words, subwords or a mixture of both. Subwords refer to character n-grams (to capture sequential constraints) or syllable n-grams.

We have designed a multi-scale query formulation process as depicted in Figure 7.3. The initial form of our query consists of a bag of English terms, e.g. Israeli <ph> Prime Minister </ph> <ne> Benjamin Netanyahu </ne> (where the tags <ph> </ph> denote a phrase for translation, and <ne> </ne> denote a named identity for subword transliteration). Phrase and word-based translations together with subword transliteration for the named entity produces the representation in Box 2 of Figure 7.3. Notice that we have formulated a query representation containing both words and syllable n-grams at this stage.[11] To alleviate the problem of word tokenization ambiguities, we may formulate overlapping character n-grams[12] from the translated Chinese words, as shown in Box 3. If we use the query formulated at this stage, we have a hybrid of overlapping character and syllable n-grams. In Box 4 we have mapped each character into its syllable pronunciations, and produced a syllable representation of the query. While this representation is *homogeneous* (in terms of syllables), it preserves *heterogeneous* information derived from phrase/word-based translations, boundaries of terms, overlapping sequential constraints, and subword transliterations. The homogenous representation offers simplicity for retrieval.

[11] As will be seen, bigrams fare best, hence we have include the syllable bigram in our illustration (Figure 7.3).

[12] Again the bigram s is found to fare best among the character bigrams.

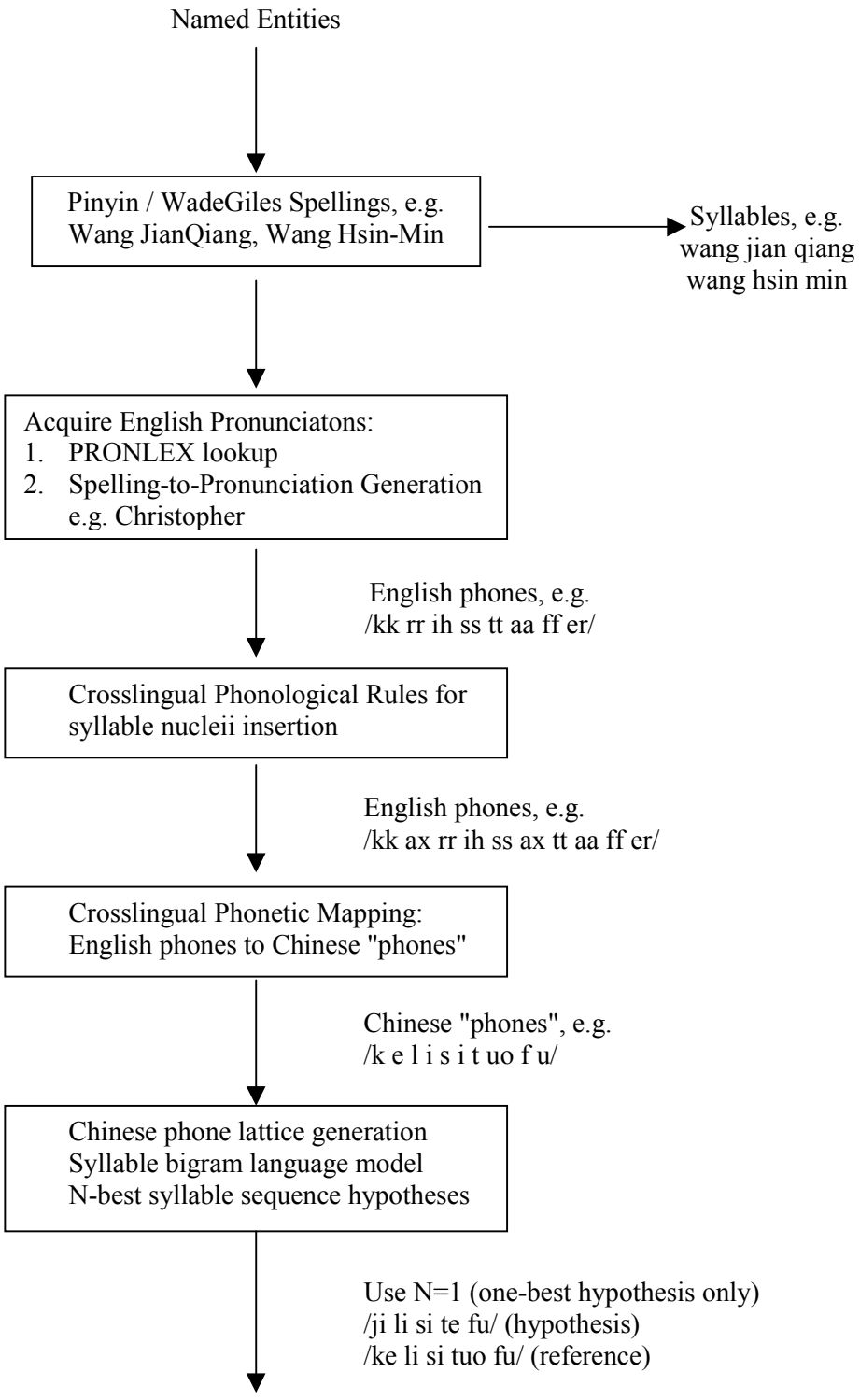


Figure 7.2 Our process for subword transliteration used to handle untranslatable named entities. Subword transliteration uses crosslingual phonetic mappings between English and Chinese pronunciations.

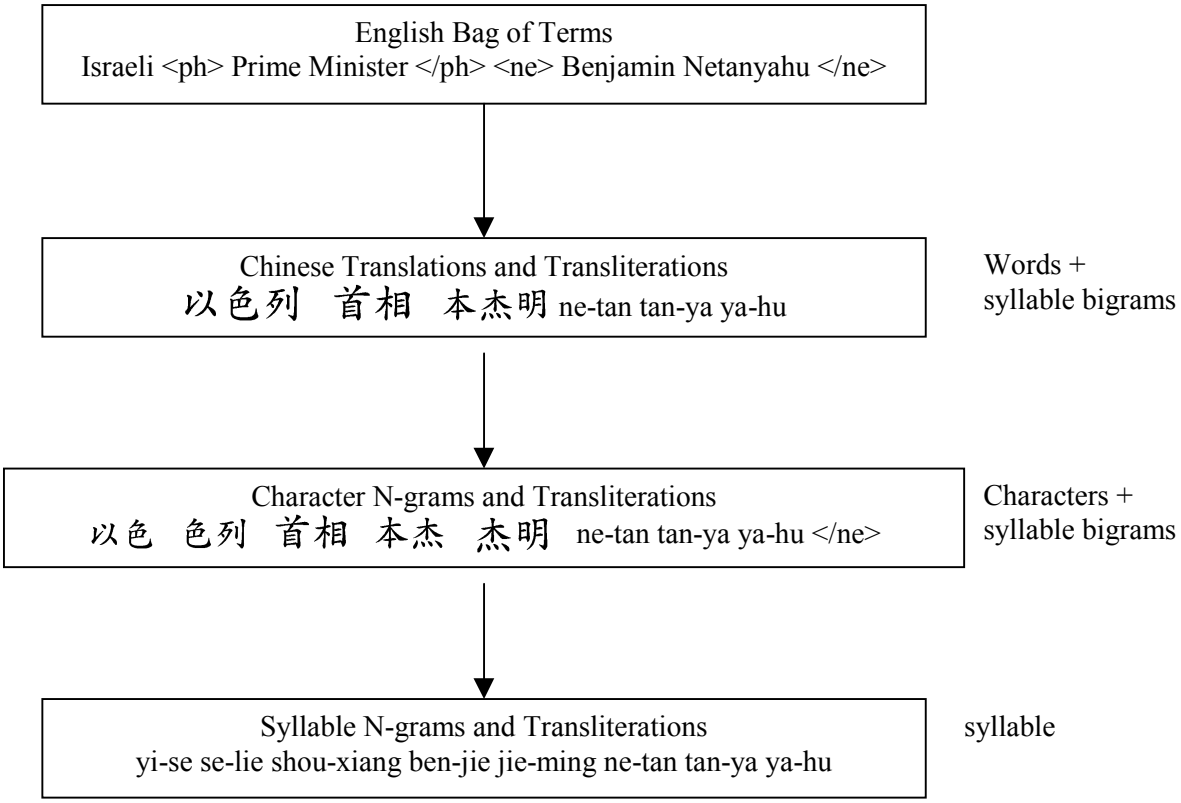


Figure 7.3 The process of multi-scale query construction in the MEI project. The query representations at various stages of processing may be used. The representations seek to integrate information such as phrase-based / word-based translations, subword transliteration, and overlapping character n-grams which alleviates the problem of word tokenization ambiguities

8. Multi-scale Audio Indexing

A popular approach to spoken document retrieval is to apply Large-Vocabulary Continuous Speech Recognition (LVCSR) for audio indexing, followed by text retrieval techniques. As mentioned previously, Mandarin Chinese presents a challenge for word-level indexing by LVCSR, because of the ambiguity in tokenizing a sentence into words. Furthermore, LVCSR with a static vocabulary is hampered by the out-of-vocabulary (OOV) problem, especially when searching sources with topical coverage as diverse as that found in broadcast news.

By virtue of the monosyllabic nature of the Chinese language and its dialects, the syllable inventory can provide a *complete phonological coverage* for spoken documents, and circumvent the OOV problem in news audio indexing, offering the potential for greater recall in subsequent retrieval. The approach thus supports searches for previously unknown query terms in the indexed audio.

This advantage was pointed out by Ng in [Ng 2000], which is a thorough study on subword indexing based on the TREC-8 spoken document retrieval evaluation. The subword approach is also more generalizable to other languages in translingual speech retrieval. However, Ng also cautioned that the subword inventory may lose discrimination power between relevant and irrelevant documents when compared to word indexing, due to the exclusion of lexical knowledge. It is important to mitigate the loss by modelling the sequential constraints of subword units. Monolingual English experiments were conducted, and he demonstrated that the overlapping phoneme trigrams were the best subword unit to index. The resultant retrieval performance is comparable to that of word-based indexing when error-free recognition is simulated.

We plan to investigate the efficacy of syllables as subword units for Mandarin audio indexing. First, the syllables need to be recognized accurately from the audio. Second, we need to model the syllable sequential constraints effectively for audio indexing. Third, we will investigate the use of a "syllable/word hybrid".

8.1 Audio Collections

The TDT-2 collection for development testing and the TDT-3 collection for evaluation. Both collections provide documents from two English newswire sources, six English broadcast news audio sources, two Mandarin Chinese newswire sources, and one Mandarin broadcast news source (Voice of America). Manually established story boundaries are available for all audio collections. The TDT-2 collection includes complete relevance assessments for 20 topics, and the TDT-3 collection provides the same for 60 additional topics, 56 of which have at least one relevant audio story. For each topic, at least four English stories and four Chinese stories are known. For both the TDT-2 and TDT-3 collections, some recordings that are not news stories were discarded, and the detailed statistical information is

	TDT-2				TDT-3			
Number of Testing Spoken Documents	2265 Docs (46.03 Hrs)				3371 Docs (98.43 Hrs)			
	Min.	Max.	Mean	Deviation	Min.	Max.	Mean	Deviation
Document Length (Syllables/Chars)	23	4,841	287.05	373.30	19	3,667	415.10	378.73
Document Length (Words)	15	3,042	182.82	244.35	9	2,523	268.07	317.62

Table 8.1 Some statistical information of the TDT-2 and TDT-3 Mandarin news audio collections.

summarized in Table 8.1. For both the TDT-2 and TDT-3 Mandarin news audio collections, Dragon's LVCSR word outputs are provided.

8.2 The DRAGON Benchmark

Before starting to run the retrieval experiments, we need to know the level of Dragon's recognition performance. For both the TDT-2 and TDT-3 Mandarin news audio collections, we found that the anchor scripts can cover a very high percentage of the audio data. Therefore, we can use them to evaluate the recognition accuracy of Dragon's recognized outputs. Dragon's recognized outputs are word sequences with word boundaries notated while the anchor scripts are plain Chinese text without word boundaries. Since we did not have Dragon's lexicon at hand, we segmented the anchor scripts using a modified LDC CallHome Mandarin lexicon, which consists of 48K Chinese words. To reduce the mismatch in word tokenization between the anchor scripts and Dragon's recognized word outputs, we re-segmented Dragon's word outputs using the same lexicon. The above procedures are definitely unable to provide a perfect evaluation, the word error rate obtained in this manner is still referable. The word error rates for both the TDT-2 and TDT-3 Mandarin news audio collections are given in Table 8.2, in which the detailed insertion/deletion/substitution rates are also provided. Note that only the documents with length less than 500 characters/syllables were used for this evaluation for simplicity. As a result, for the TDT-2 collection, 1954 out of the 2265 documents were used; and for the TDT-3 collection, 2430 out of the 3371 documents were used. The character error rate is a relatively simpler evaluation metric than the word error rate. We simply remove word boundaries from Dragon's recognized word outputs and its character outputs to the corresponding anchor scripts. It should be noted that the character error rate is the only evaluation matrix that gives a perfect evaluation, and hence

it is the most popular evaluation matrix for Chinese. To get the syllable error rate, both the Dragon's recognized outputs and the anchor scripts must be converted into syllables by pronunciation lookup using a Chinese pronunciation lexicon. Again, we used the modified LDC CallHome Mandarin lexicon. The character error rate and syllable error rate are also

	TDT-2 Test on 1954 Docs (22.99 Hr)	TDT-3 Test on 2430 Docs (27.52 Hr)
Word	17.96 (3.54/2.65/11.77)	19.12 (4.15/2.87/12.10)
Character	12.06 (1.93/0.77/9.36)	12.98 (2.61/0.83/9.54)
Syllable	7.91 (1.94/0.79/5.18)	8.60 (2.61/0.83/5.16)

Table 8.2 Error rates (Insertion/Deletion/Substitution) (%) of Dragon's recognized outputs with respect to the anchor scripts on the partial set of TDT-2 and TDT-3 Mandarin news audio collections.

listed in Table 8.2. These results are in parallel with those Dragon reported previously [Zhan et al 1999].

8.3 Syllable-based Indexing

For both the TDT-2 and TDT-3 Mandarin news audio collections, Dragon's LVCSR word outputs are provided. First of all, we can use Dragon's word outputs for word-based indexing. Second, we can use characters derived from Dragon's word outputs for character-based indexing. Third, we can use syllables derived by pronunciation lookup using Dragon's recognized words. That is, given Dragon's LVCSR words outputs, we can definitely investigate the use of words and subwords (including characters and syllables) in retrieval. However, we also want to address the problem of automatic speech recognition (ASR) errors by introducing alternative hypotheses. Although only the single best output from Dragon is available for every document, this still can be achieved by augmenting Dragon's recognized outputs with alternative word/character/syllable hypotheses. As mentioned earlier, once the syllable recognition failed, both characters and words would definitely be misrecognized. Thus, at the first stage, we plan to focus only on the syllables in addressing this problem. Another reason for working on syllables is that we have used filtered syllable lattices to achieve robust retrieval based on imperfect recognized transcripts in monolingual Chinese retrieval experiments [Wang 2000]. Consequently, we want to design a structure for document indexing which incorporates Dragon's word/character/syllable hypotheses and MEI's syllable hypotheses. Hopefully, MEI's syllable hypotheses can be complementary to Dragon's syllable hypotheses. In the following, we will first illustrate the syllable-based indexing scheme.

The indexing terms consisting of overlapping N-grams have been shown to be effective for subword-based SDR. Different kinds of recognition errors, such as insertions, deletions and substitutions, introduce different kinds of errorful N-gram terms. Among these recognition errors, substitutions cause the most serious problem. A simple example of using bi-grams is illustrated in Figure 8.1. With one substitution error, two desired bigrams are deleted while two errorful bigrams are inserted. In our previous syllable-based spoken document retrieval experiments for Mandarin Chinese, we found that providing alternative syllable hypotheses in a syllable lattice has helped improving the retrieval performance. Figure 8.2 depicts a simple example of syllable lattice of depth 3. However, not only correct syllable bigrams but also much more errorful syllable bigrams were extracted from the syllable lattice. To address this new problem, we applied a syllable verification technique to reduce the depth of the syllable lattice and replaced the term frequency with the normalized speech recognition scores. This approach was shown to be successful when the speech recognizer gave a syllable error rate of around 30-40%. In this task, as mentioned earlier, Dragon's recognizer gives a syllable error rate of less than 10%, which is much better than the recognizer we used before. From Table 8.2, we found that, if we try to augment Dragon's syllable outputs with alternative syllable hypotheses as shown in Figure 8.3, we can at most compensate the substitution errors. That is, if we provide one alternative syllable hypothesis for every Dragon's syllable, the accuracy for MEI's syllables can be at most 5%, or in other words more than 95% of the alternative syllables are errorful. We therefore designed a revised syllable lattice as shown in Figure 8.4. From now on, when we mention the syllable lattice, we mean the revised lattice rather than the original lattice. In addition to the lattice form, we can of course treat Dragon's syllables and MEI's syllables as two separate syllable outputs. Figure 8.2 explains the difference between two different combinations of Dragon's syllables and MEI's syllables when only one syllable or two contiguous syllables are different between these two recognition outputs. As to the combination of words and subwords, either loose coupling or tight coupling can be adopted, which will be described later on. In addition, to make MEI's syllables more useful, we need to improve MEI's syllable recognizer.

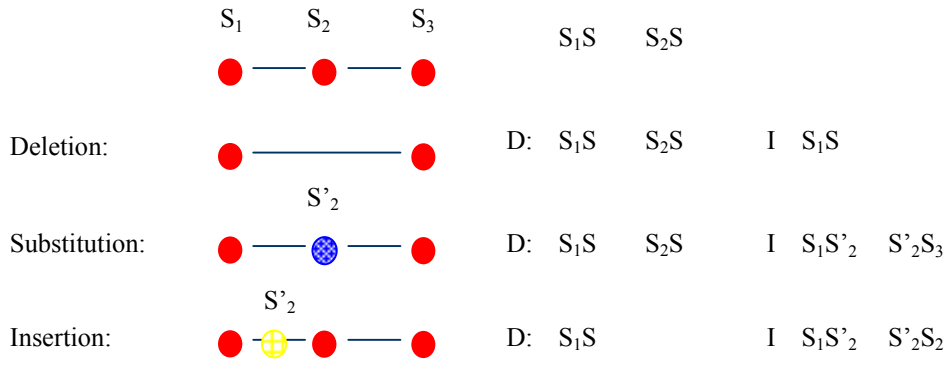


Figure 8.1 ASR errors vs. IR term errors, using bigrams

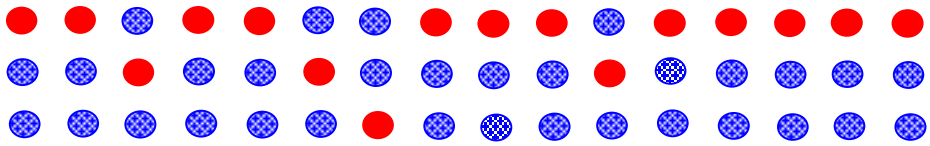


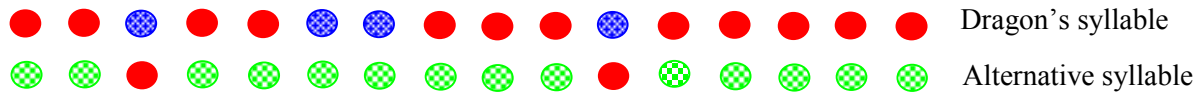
Figure 8.2 Syllable lattice of depth 3

8.5 MEI Syllable Recognition

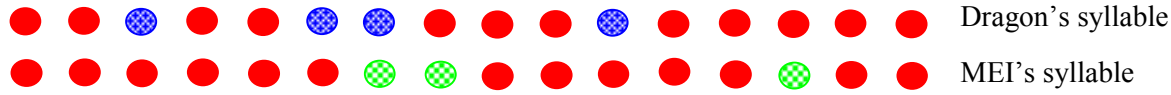
In MEI's syllable recognizer, spectral analysis is applied to a 20 msec frame of speech waveform every 10 msec. For each speech frame, 12 mel-frequency cepstral coefficients (MFCC) [Junqua et al., 1993] and log energy are extracted, and these 13 coefficients along with their first and second time derivatives [Furui 1986] are combined together to form a 39-dimensional feature vector. In addition, cepstral mean subtraction (CMS) [Furui 1981] is applied to all spoken documents. The acoustic units chosen for syllable recognition are 112 context-dependent initials and 38 context-independent finals based on the monosyllabic nature of Mandarin Chinese and the initial/final structure of Mandarin base syllables. Each initial is represented by a HMM with 3 states while each final is represented by one with 4 states. The Gaussian mixture number per state ranged from 2 to 16, depending on the amount of training data. Therefore, every syllable unit was represented by a 7-state HMM. The silence model was a 1-state HMM with 32 Gaussian mixtures trained using the non-speech segments. The above acoustic models were trained on 11 hours of the VOA part of the Hub4 Mandarin broadcast news collection. In addition, the syllable-based *N*-gram language models were trained on the 1998 XinHua newswire text corpus, which consists of 40 million Chinese characters. Note that both the newswire text corpus and the TDT2/TDT3 broadcast news collections were collected almost in the same time frame. Word segmentation and phonetic labelling were performed for the training materials using the same modified LDC lexicon.

Due to the very tight time limit of the workshop, a three-stage search procedure was used to reduce the recognition time needed. In the first stage, according to Dragon's recognized outputs, a segment of silence with duration longer than 0.2 second was taken as a sentence boundary. We simply performed a free syllable decoding (FSD) at the sentence level. In the second-stage, based on the state likelihood scores calculated in the first stage and the syllable boundaries of the best syllable sequence, the syllable recognizer further performed the Viterbi search on each utterance segment which might include a syllable and output several most likely syllable candidates. An initial syllable lattice was thus constructed. In the third stage, A* search with the syllable bigram language models applied in the forward search and the syllable trigram language models applied in the backward search was applied to the initial syllable lattice and generated a new best syllable sequence. Note here, the initial

syllable lattice was temporarily generated during the MEI's syllable recognition process. The MEI's syllable recognizer would finally output a single best syllable sequence. We first conducted syllable recognition experiments on the TDT-2 developing set. Table 8.3 summarizes the recognition results. Using free syllable decoding (FSD), the baseline syllable error rate is 37.09%. With the syllable language models applied (FSD+LM), the syllable error rate is reduced to 25.88%. Because Dragon's recognized outputs of high accuracy are provided, a MAP speaker adaptation procedure based on Dragon's recognized outputs is adopted to refine MEI's acoustic models. As can be found in Table 8.3, with such a speaker adaptation scheme applied, the syllable error rate is further reduced to 10.05% (MAP + FSD + LM), which is only about 2% higher than Dragon's syllable error rate. Note that MEI's syllables obtained in this manner are not necessary with the same sentence length as Dragon's syllables. Thus, they could only be used as an alternative source for syllable-based document indexing, but could not be incorporated with Dragon's syllable to construct a syllable lattice. In order to provide alternative hypotheses for every individual Dragon's syllable, a word-level force alignment (WLFA) was executed based on the word and word boundary information provided by Dragon's recognized outputs. The results for without syllable language models (MAP + WLFA) and with syllable language models (MAP + WLFA + LM) are also given in Table 8.3. As expected, they are worse than those obtained based on forced alignment at the sentence level. However, MEI's syllables obtained in this manner could be incorporated with Dragon's syllables to construct the syllable lattice for syllable-based document indexing. The recognition results for the TDT-3 Mandarin audio collections are also provided in Table 8.3 The recognition experiments were conducted based on the same conditions as the TDT-2 collection. It can be found that these results reveal the same trends as the results for the TDT-2 collection.

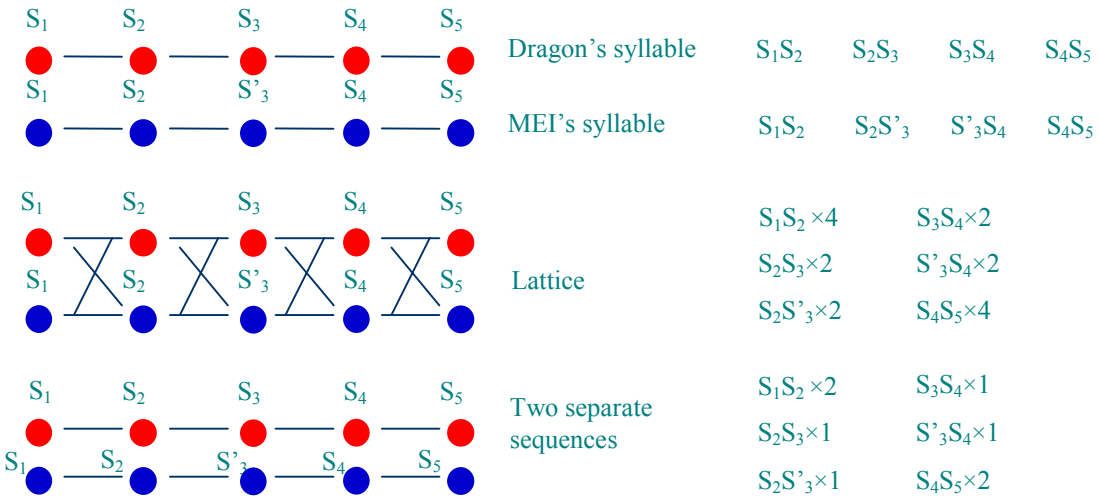


(a) Augmenting Dragon's syllables with alternative syllable (the original lattice)

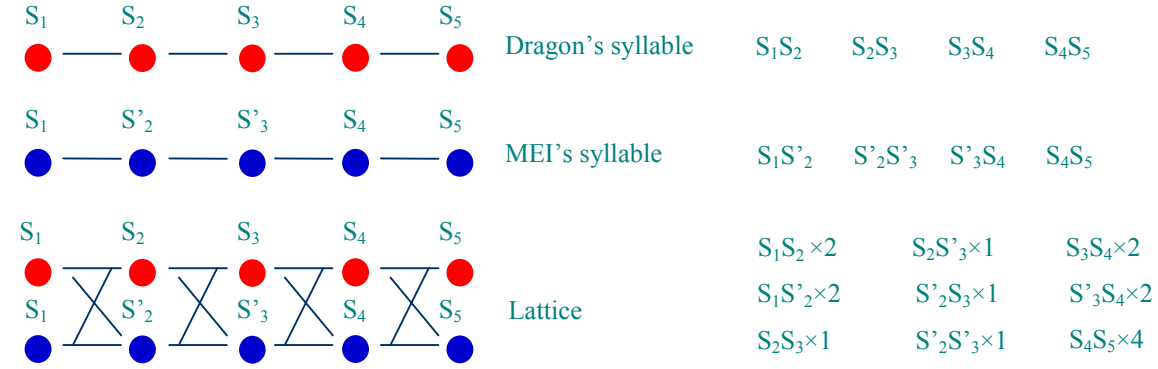


(b) Combination of Dragon's syllables and MEI's syllables (the revised lattice)

Figure 8.3 Augmenting Dragon's syllable outputs with alternative syllable hypotheses.



(a) One disagreement between Dragon's syllables and MEI's syllables



(b) Two contiguous disagreements between Dragon's syllables and MEI's syllables

Figure 8.4 Bigram terms extracted from different combinations of Dragon's syllables and MEI's syllables.

	TDT-2 Test on 22.98 hrs	TDT-3 Test on 27.52 hrs
FSD	37.09	36.13
FSD + LM	25.88	25.53
MAP + FSD	12.28	12.90
MAP + FSD + LM	10.05	10.82
MAP + WLFA	15.98	16.32
MAP + WLFA + LM	14.27	15.80

Table 8.3 Error rates (%) of different approaches of MEI's syllable recognizer.

9. Multi-scale Retrieval

Multi-scale query formulation and multi-scale audio indexing produces representations for query and document in both the word scale and subword scale. Multi-scale retrieval attempts to fuse the use of both words and subword for our CL-SDR task. There are two main approaches for integration:pre-ranking integration (tight coupling) and post-ranking integration (loose coupling).

9.1 Tight Coupling

Ttight-coupling seeks to combine different units together into a single hybrid representation with appropriate weighting. The retrieval process will then be carried out with the hybrid representation to produce a single ranked retrieval list. An example of tight coupling is shown below.

$$D_1U_1 = [\ v_{11}, v_{12}, v_{13}, v_{14}, v_{15}, v_{16}, v_{17}, v_{18}, \ldots v_{1N_1}\ ];$$
$$D_1U_2 = [\ v_{21}, v_{22}, v_{23}, v_{24}, v_{25}, v_{26}, v_{27}, v_{28}, \ldots v_{2N_2}\ ];$$
$$\vdots$$
$$D_1U_M = [\ v_{M1}, v_{M2}, v_{M3}, v_{M4}, v_{M5}, v_{M6}, v_{M7}, v_{M8}, \ldots v_{MN_M}\ ];$$

where D_1U_M is the representation of document 1 using indexing unit M and NM is the length of the vector for indexing unit M.

The tightly coupled representation will be:

$$D_1 = [\ c_1 v_{11}, c_1 v_{12}, c_1 v_{13}, c_1 v_{14}, c_1 v_{15}, c_1 v_{16}, c_1 v_{17}, c_1 v_{18}, \ldots c_1 v_{1N_1}$$
$$c_2 v_{21}, c_2 v_{22}, c_2 v_{23}, c_2 v_{24}, c_2 v_{25}, c_2 v_{26}, c_2 v_{27}, c_2 v_{28}, \ldots c_2 v_{2N_2}$$
$$\ldots$$
$$c_M v_{M1}, c_M v_{M2}, c_M v_{M3}, c_M v_{M4}, c_M v_{M5}, c_M v_{M6}, c_M v_{M7}, c_M v_{M8}, \ldots c_M v_{MN_M}];$$

where c_M is the weight for indexing unit M.

Since the original representation for each of the indexing units may have different number of terms, different N_M, normalization over the length may be applied. Furthermore, instead of simple concatenation, integration could use other approaches such as summation, logical-and, maximum-of, etc. In the MEI project we have not delved deeply into the issue of weight optimization.

9.2 Loose Coupling

Loose-coupling combines the *ranked lists* based on retrieval with units of different scales. The ranked lists are re-scored to form a single ranked list. One possibility of loose-coupling is a linear combination of ranked list from different runs

$$score_i = \sum_{j=1}^{M} c_j \, score_{ij}$$

The equation shown above is a linear combination of scores from many runs for each of the elements. C_j is the weighting factor for the indexing unit *j*. Again, the integration could adopt other approaches such product-of, maximum-of, minimum-of instead of summation. We have optimized our C_j weights using TDT-2 and applied them when testing on TDT-3.

9.3 Experiments

In the MEI project, we have investigated both tight and loose coupling. The documents are represented in multi-scale units – character n-grams, syllable n-grams and words.

For tight-coupling, we concatenated the three different representations together with equal weights for each of the units. The retrieval is carried out over the hybrid query / document representation.

For loose-coupling, we re-scored by linear combinations of the scores from the retrieval results of the different units. We then re-ranked the integrated scores to get the integrated results. The weighting for the different indexing units in the linear combination is obtained by sweeping from 0 to 1 for the weighting factors using the development test set. The optimized weighting factors are then applied to our testing set for performance evaluation.

The optimized results of these coupling are presented together with other experimental results. Before optimization of the weighting factors, the integrated results may show degradation over the individual runs. After optimization, the integrated results show a small amount of improvement over the retrieval results from individual runs.

Results on loose and tight coupling with word and character bigrams are shown in Figure 9.1. Loose coupling outperforms either ranked retrieval list alone, while tight coupling did not achieve such improvements. Results from loose coupling between words and various subword units are shown in Figure 9.2.

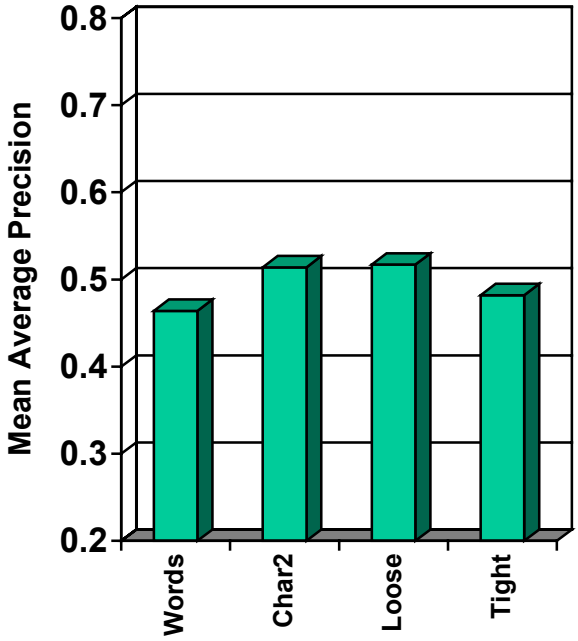


Figure 9.1 Retrieval performance based on loose and tight coupling of word and character bigram representations.

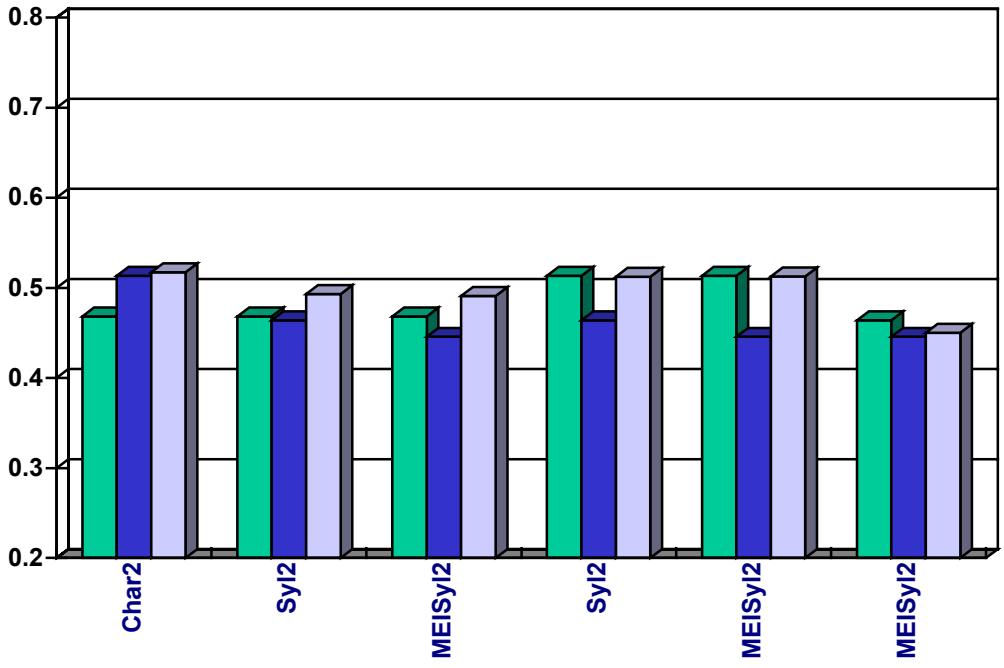


Figure 9.2 Retrieval performance based on loose coupling between word and various subword representations, e.g. character bigrams (char2), syllable bigrams from DRAGON's output (syl2), MEI syllable bigram representations (MEISyl2). In most cases, fusion with loose coupling outperforms the individual ranked retrieval lists alone.

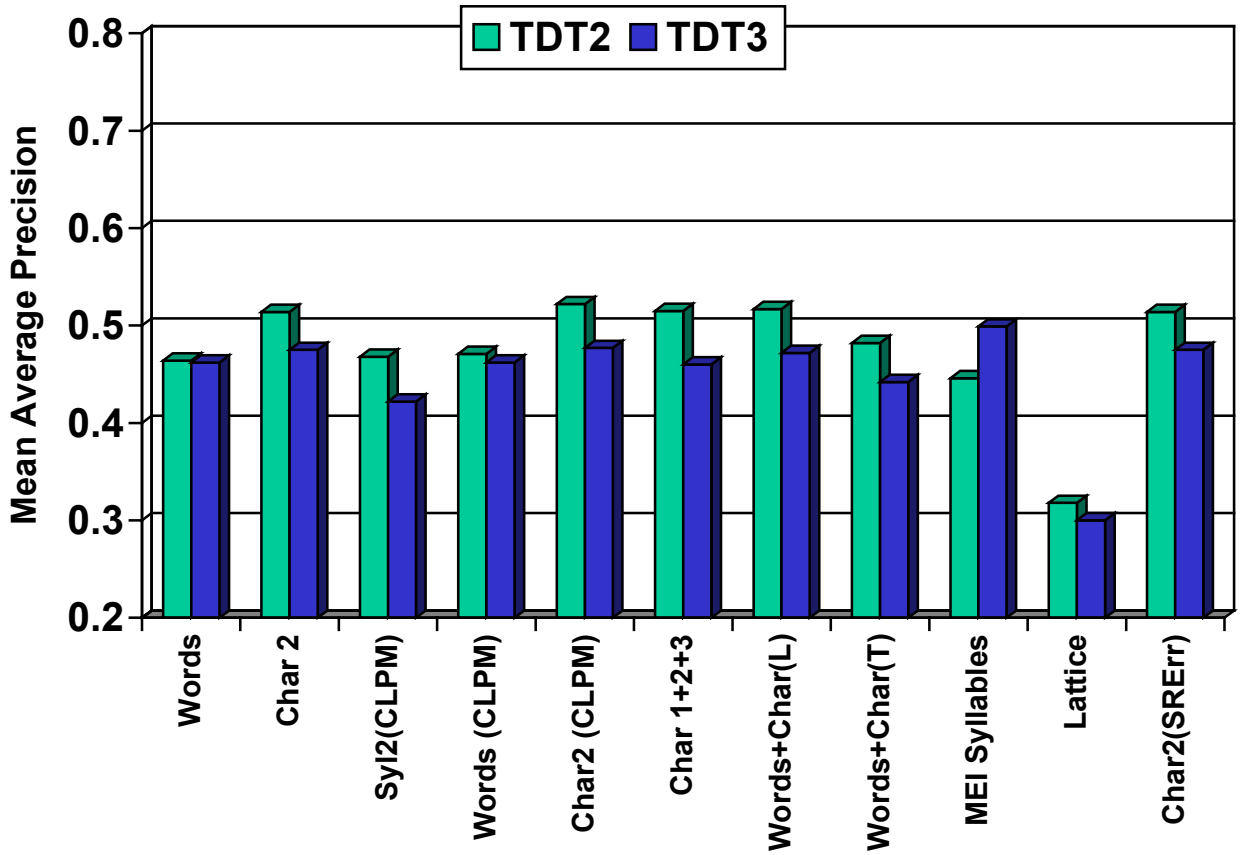


Figure 9.3 Retrieval performance of TDT-2 and TDT-3 Corpora, based on the fusion of variuos words and subword units.

Figure 9.3 shows our retrieval performance on our development test set (TDT-2) and test set (TDT-3), using different words and subword units for retrieval. Key observations include:

(i) the use of character bigrams outperform words, which suggests that these may be effective in ameliorating the problem of word tokenization ambiguities,

(ii) the use of CLPM brought about small but consistent gains to word- or character-bigram based retrieval,

(iii) fusion of character unigrams, bigrams and trigrams did not give improvements,

(iv) loose coupling of words and character bigrams outperformed either alone, but tight coupling needs further investigations,

(v) document indexing by lattice representation was surprisingly poor, the issue warrants further investigation,

(vi) query expansion with confusable words did not produce any retrieval improvements;

(vii) the MEI system generally fares better on TDT-2 than TDT-3, possibly due to optimization based on the development test set. However, it is surprising that the MEI syllable bigram representation shows reverse trends.

10. System Evolution

Earlier in this paper, we had discussed the architecture that we planned to build throughout the course of the Johns Hopkins Workshop 2000. By the end of the project, we were indeed successful in implementing everything that we had set out to construct. However, the implementation process, as is often the case, did not follow the exact course that was originally planned. This section chronicles the evolution of the MEI system throughout its development lifecycle over the course of the JHU Workshop.. We also provide insights into the performance gains and significant stumbling blocks that were encountered in each step.

10.1 System Zero

In the MEI original plan, we had intended that upon arrival at Johns Hopkins, we would begin on the first day to begin immediate failure analysis on the so-called System One which we had hypothesized would be in place. However, by the first day of the workshop, we had nothing more available than a demonstration system that we had built in order to instruct the attending undergraduate students. This meant that a backoff posture was necessary. We elected, therefore, to first build a baseline system: System Zero.

Before describing System Zero, we first remind the reader that the high-level motivation for the MEI team was to build a system to handle a scenario where a user could provide a set of English documents to try to query for Mandarin audio files on the same topic. This implies that one first must be able to extract meaningful components from the Mandarin audio files. As was described, the Dragon speech-to-text system had been applied to the Mandarin audio files in our collection, so we were able at the onset of this project to extract Chinese words from the audio. The next necessity for the system was some sort of translational feature that would either map English text to Chinese or would convert Chinese transcribed audio to English. Our group had elected for the former of these. We therefore made use of a word-for-word translation system provided by the University of Maryland to convert from English to Chinese. Lastly, we needed an information retrieval capability where we used the InQuery system e obtained from the University of Massachusetts.

System Zero was designed to have only this bare minimum capability. It expected that the user would be utilizing a fixed set of English queries and that the output of the Dragon speech-to-text would be the only document collection. Given these constraints, the system would permit the user to specify the maximum number of words that he or she would allow to be in any given query as well as the maximum number of Chinese translations that could be permitted for each English query word. It should also be mentioned that the user could additionally supply an automatically-generated list of terms from the queries that were ordered in such a way as to tell the system which words it should retain when the query sizes exceeded user-chosen maximums. System Zero is here illustrated in Figure 10.1.

As can be seen from the figure, this baseline system could only provide minimal performance for the overall system. When only one Chinese translation was allowed for each English query word, and given the optimal choice of number of terms in the query, mean average precision on a single query batch was only 0.169. We have mentioned in section 6 that given a cheating experiment, one could attain mean average precisions as high as 73.8%, and doing monolingual queries produced results in the mid-fifties. Furthermore, the example in which a human translated the English queries to Chinese *also* gave provided results in the low fifties. Given that these are upper bounds far exceeded the performance of System Zero, it was clear that there was substantial effort that would need to be put forth in order to obtain a state-of-the-art system.

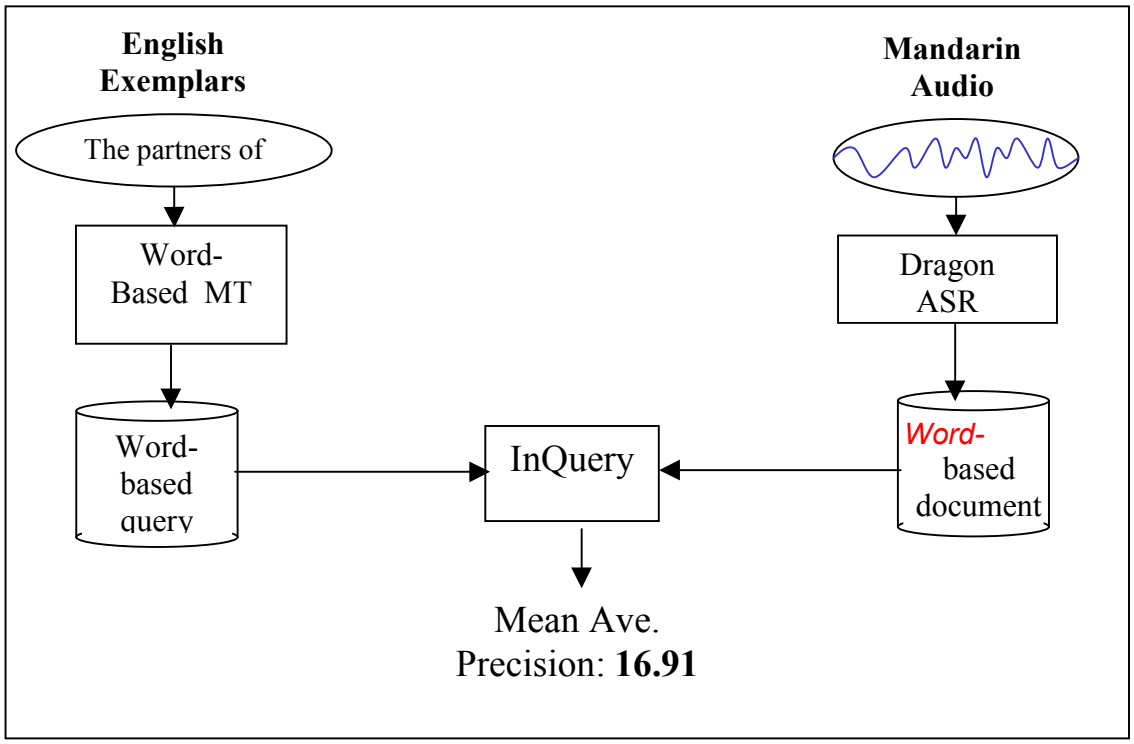


Figure 10.1 System Zero, our baseline system.

10.2 Incorporating N-grams

Our initial architecture designs called for a System One that would allow not only the processing of words, but additionally, the processing of subword units: in particular, overlapping character or syllable *n*-grams. As will be seen shortly, the incorporation of these overlapping *n*-grams would serve as a significant boost to overall system performance.

Incorporation of overlapping character n-grams was a relatively straightforward task. One basically needs only to take words and some desired size for *n* and then slide a running window along the words and report every window's *n*-gram as a single component of the output. Generation of syllable *n*-grams required an additional step in the processing. To effectively process syllables, one needs first to be able to represent the sounds that describe each word. To generate sounds for each word, the MEI team used a Mandarin pronunciation lexicon to first resegment the translated query or ASR document according to a greedy search (i.e., segment by always trying to find the longest Chinese string for which there is a lexicon entry), and then use the pronunciations provided by the lexicon as representations for the sounds of the words.

A difficulty with *n*-gramming, however, is determining whether overlapping should be limited to strictly the document side since it represents true Mandarin ordering or whether it should also include the query side. We elected to use full *n*-gramming on the document collection, but to use only *within* word *n*-gramming on the query. It should be noted that this means for $n>k$, any *k*-grams that were in the original query could not hope to match any words in the document collections. Hence, if *n* were chosen to be of size two, every token in the document collection would be a bigram, so residual unigrams in the query would have nothing that they could match.

Incorporation of *n*-grams performed surprisingly well. In fact, one could say that their performance represented the first big surprise of the MEI findings. Observations revealed that overlapping character bigrams ($n=2$) proved to be a better unit for querying the ASR documents than were words. Mean average precision jumped from 0.169 for the word-based approach to 0.200 for the character bigram approach. In fact, throughout the entire project, character n-grams routinely outperformed word-based approaches. This may be due largely to the fact that a majority of Chinese text can be tokenized as bigrams but, as was mentioned, by using bigrams one is effectively throwing out all information that the unigrams might have contributed**.** Syllable bigrams also outperformed words in this early stage of architecture implementation, but as will be seen later, this trend did not continue.

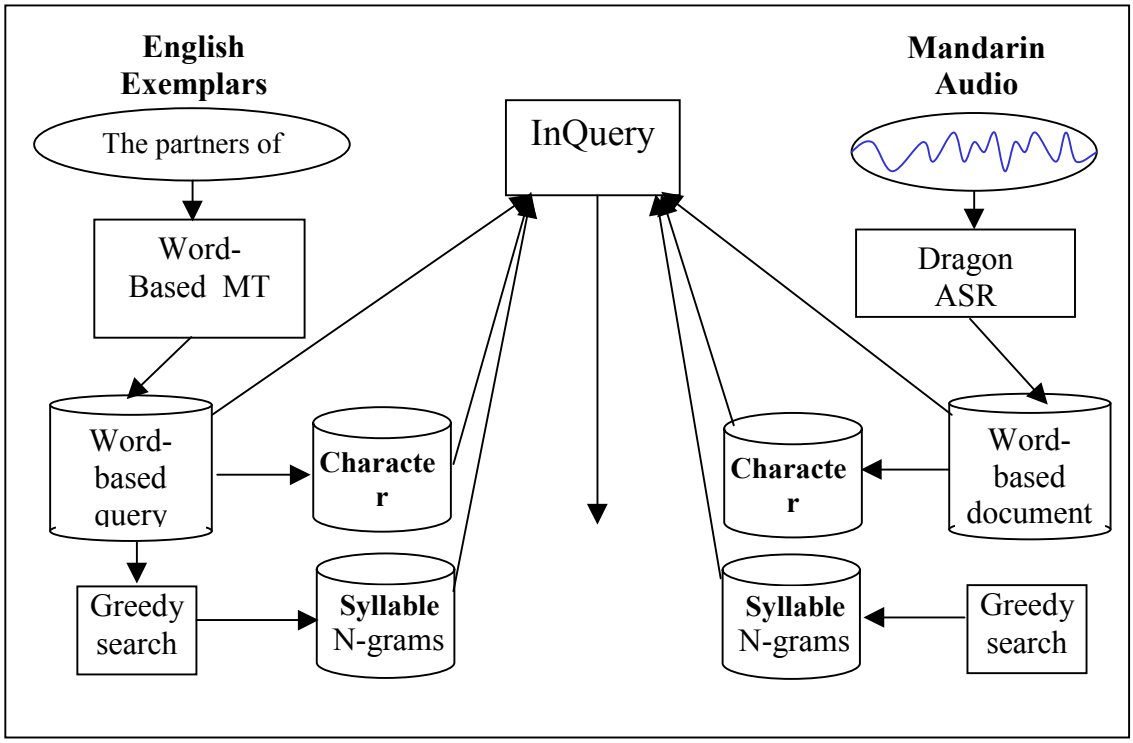


Figure 10.2 System One – incorporating subword (character / syllable) n-grams.

10.3 Structuring the Queries

Up until this point, translated queries consisted of little more that bags of words. This meant that when translation was being performed, more ambiguous words were getting more "credit" than words that were translated unambiguously. For example, suppose an English query consisted of only two words $\{E_1, E_2\}$ where E_1 had three Mandarin translations $M_{1,1}$ $M_{1,2}$, and $M_{1,3}$ and E_2 could only be translated as $M_{2,1}$. Then if the user selected a maximum of three or more translations per word, E_1 would have three words to represent it where E_2 would have only one. This is exactly opposite of the behavior one would desire, for two reasons. First, and very importantly, words that only have a single possible translation are likely to be very key words to a query, so they should actually be enhanced rather than de-emphasized. Second, words that are ambiguous should provide less confidence to the translation rather than more.

Fortunately, the InQuery system (the information retrieval engine) has the capability of implementing structured queries akin to a method proposed by [Pirkola 1998]. These kinds of queries have the capability of treating sets of words as if they were all synonymous. To be more explicit, one can specify a query such as

$$\#\text{syn}(M_{1,1}\ M_{1,2},\ M_{1,3})\ M_{2,1}$$

and InQuery will treat all of the first three M values as if they were each instances of some pseudoword. By this means, the query can be viewed as if it contains only two words, namely the pseudoword and $M_{2,1}$. This overcomes the problem of overweighting words of less confidence, so we incorporated this as the next incremental improvement of our system (see Figure 10.3).

This gave rise to significant improvements in performance (0.240 mean average precision on words). However, we began to get some very strange and interesting effects when we began to use the *n*-gram components. Since only the Chinese words in the query were decomposed into overlapping *n*-grams, this meant that the #syn operator was still wrapped around such *n*-grams. This implies that if one has several translations possible for a given word, then all of the *n*-gram *subcomponents* of those translations are synonyms of each other. At first glance, it seems that this should cause a catastrophic failure for the overall system, but such was not the case. On the contrary, much to our surprise, performance was actually improved. We were able to get character bigram performance of **27%?** average precision. Later in our experimentation, we were actually able to find a query formatting strategy (namely, balanced queries) that typically outperformed the structured format, but this strategy will be talked about later in the paper.

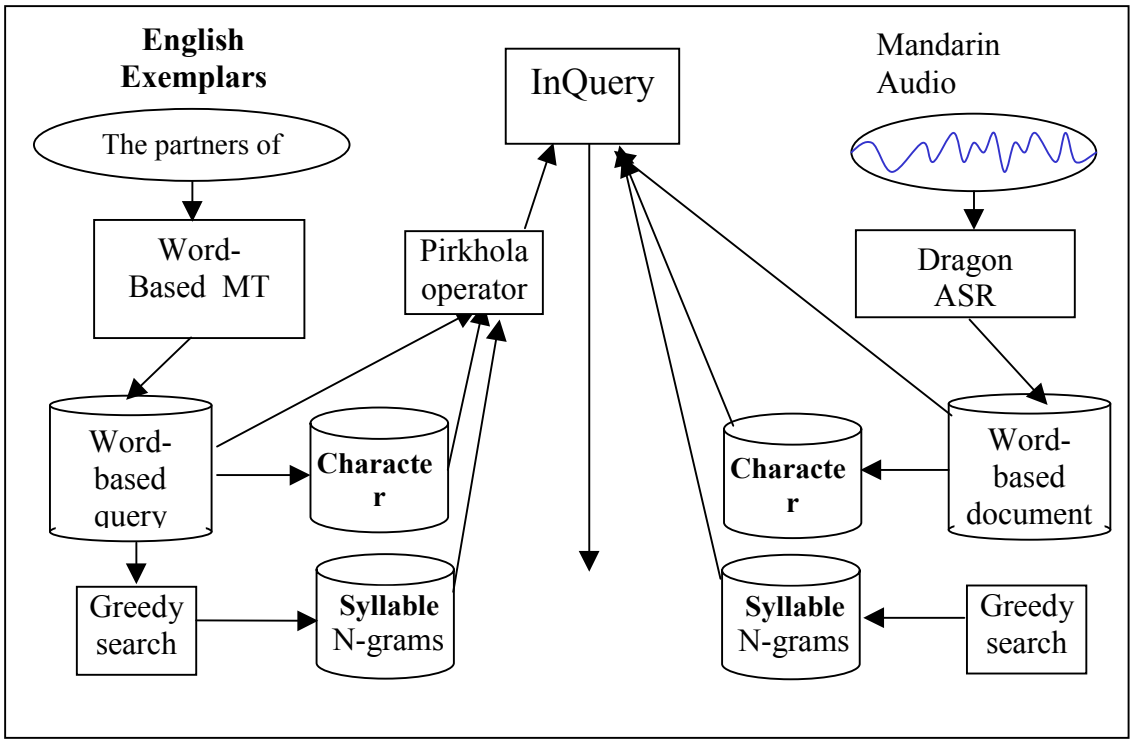


Figure 10.3 Structuring the queries.

10.4 Improving upon the Queries

Although the incorporation of structured queries assisted the overall performance, there was additional effort needing to be made in the query development process. Recall, for instance, that the translation process was a word-by-word operation. This worked reasonably well, but it is clear that in English (as well as Chinese), there are also phrases and multiword units that one might light to take advantage of. For an English example, one might expect that querying with a whole unit such as "United Nations" would yield better retrieval performance than querying with individual words "United" and "Nations." Furthermore, since the translation process was initially translating on a word-for-word basis, it would behoove a translation system to translate whole phrases as units rather than as singular terms.

The next improvement to the system, therefore, was to take advantage of such phrasing. One can detect phrasing in one of two ways. The first of these is to use a lexicon to find static phrases that are common to the language. (Refer to section 7 for details).

However, for queries, a particularly important kind of phrase is that describing named entities. Names of people and organizations can be highly beneficial in a search engine. This is even more true when one is trying to query by example, since names found in the document that is being used for the query may be quite selective of the kids of documents expected to be

retrieved. BBN Technologies was able to named-entity tag all of the English documents we had intended to use for queries, so by incorporating this kind of tagging, as well as the static phrasing component, the system how now evolved to that which is depicted in Figure 10.4.

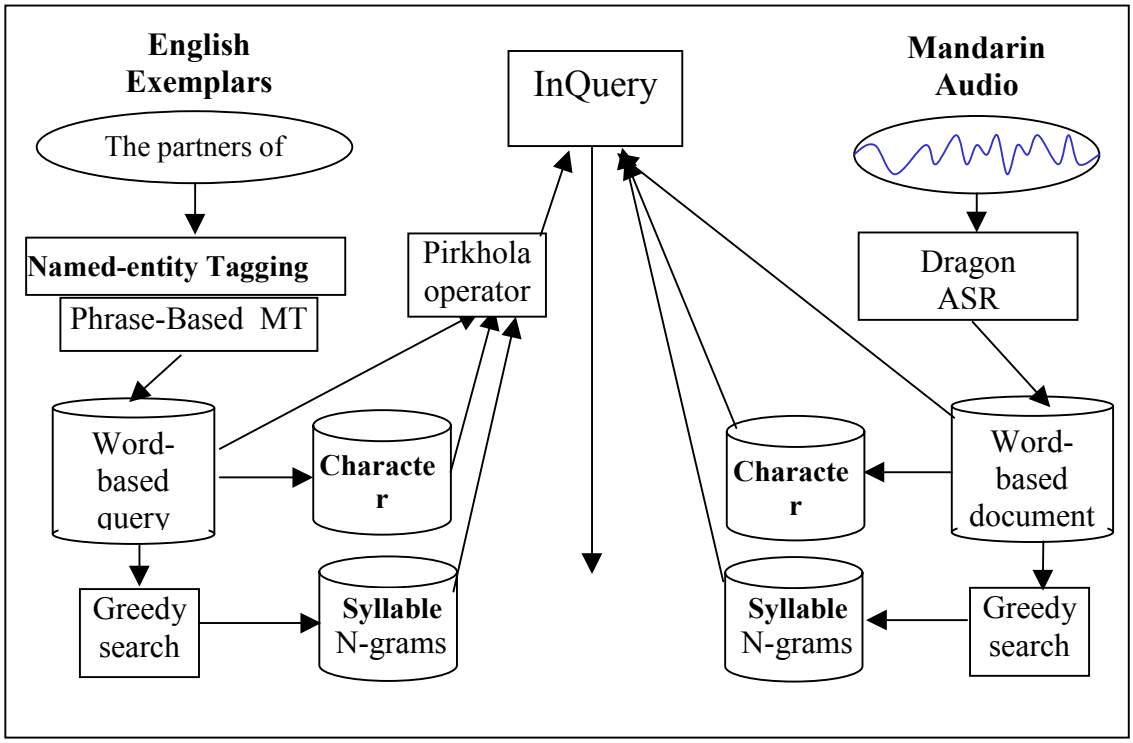


Figure 10.4 Migrating from word-based to phrase-based machine translation.

Phrasing and named-entity tagging created a significant boost in overall performance. No longer were we seeing mean average precisions in the twenties, since now we had attained precisions as high as **35%**?. Since this precision was better than 50% of the mean average precision for the monolingual system, we recognized that we had now brought the system to a state in which it was actually competitive. The question was, could we do more?

10.5 Numerical Processing and CLPM

Translating terms from English to Mandarin works adequately well when one uses an extremely large dictionary and when one is not interested in the cross-lingual equivalents of names and numerical values. Names and numerics are often excluded from lexicons, so regardless of the lexicon size, one typically has to either omit these kinds of terms or do special processing. We chose to do the latter with thoughts that this might, perhaps, improve the overall performance. When foreign names are represented in Chinese texts, they can either be translated (if subcomponents of the names are frequent enough to be found in

lexicons) or they can be transliterated. This calls for the CLPM algorithm described in Section 7.7. Verbalization of numeric expressions are also performed.

10.6 Coupling

The output of numerical processing is words, but the output from CLPM is syllables. Furthermore, up to this point in our research, the best overall system was character bigram based. In order for one to be able to take full advantage of each of these components, it became clear that there was going to be a definite need for some kind of system coupling. The strategy for combining, however, was a little unclear. Many people have experimented with *loose coupling*. In this paradigm, one might run two or more separate systems and then use training or development data to optimally reshuffle the outputs of the individual systems in a way that improves the overall mean average precision. Another kind of coupling is tight coupling, where multiple kinds of formats are represented both at the query level as well as the document level and are queried simultaneously. Details for loose coupling has been described in section 9. Our multi-scale query formulation (as described in section 7.8) provides one kind of tight coupling, which we refer to as "backing off". To follow up on our previous example with "prime minister Benjamin Netanyaho", the first three terms can be translated as 以色列 首相 本杰明.

The system requires the aid of the CLPM in order to transliterate “Netanyahu.” In this case, the system will have to “back off” to a syllable representation of this word, namely “ne tan ya hu.” This means that syllables are only used when there is no other alternative.

However, another alternative is to use both the word and its syllable *n*-gram components when both exist, or just use syllables when they alone exist. Due to the existence of the synonym operator, one could represent the query as

#syn(以色列 #1(yi-se se-lie)) #syn(首相, shou-xiang)

#syn(本杰明 #1(ben-jie jie-ming)) #2(net-tan tan-ya ya-hu)

where the #n operator tells InQuery that the words must be found in order and that last word can occur no more than m words away from the first. There is an innate problem with this kind of query. Specifically, by having to place the #n operator around the syllable components, the information retrieval becomes less robust to speech recognition errors. If the IR engine finds several but not all of the syllable bigrams within a particular document, then it will give no credit to that document, which is not true when the operator is not in place.

There are two ways to bypass this problem. One is to remove the operator altogether. We saw before that the synonym operator functioned well even when it was placed around syllable subcomponents. However, now we have not only the syllable subcomponents but the words themselves. This has the extremely ill effect that a word, no matter how rare, gets equated with the most common of its syllable n-gram subcomponents. Clearly, this should not be the method of choice. A second alternative is to use some other kind of operator. Such an operator is InQuery's #sum operator which computes the average of the terms within it. If the #sum operator were available, one could then construct the query as

#sum(以色列 #sum(yi-se se-lie)) #sum(首相, shou-xiang)

#sum(本杰明 #sum(ben-jie jie-ming)) #sum(net-tan tan-ya ya-hu)

This formulation suggests that the syllable n-grams for a word count collectively as much as the word they were derived from, and that the word and its syllables counts no more than any other.

However, though InQuery does have the #sum operator, this operator was not integrated into our system. It was our goal that our whole system be easily modifiable using only command-line arguments or a batch file. We therefore wanted the system to be easily adjusted to swap between the #syn, #n, and #sum operators. This, therefore, was the next component that needed to be added to the implementation stage.

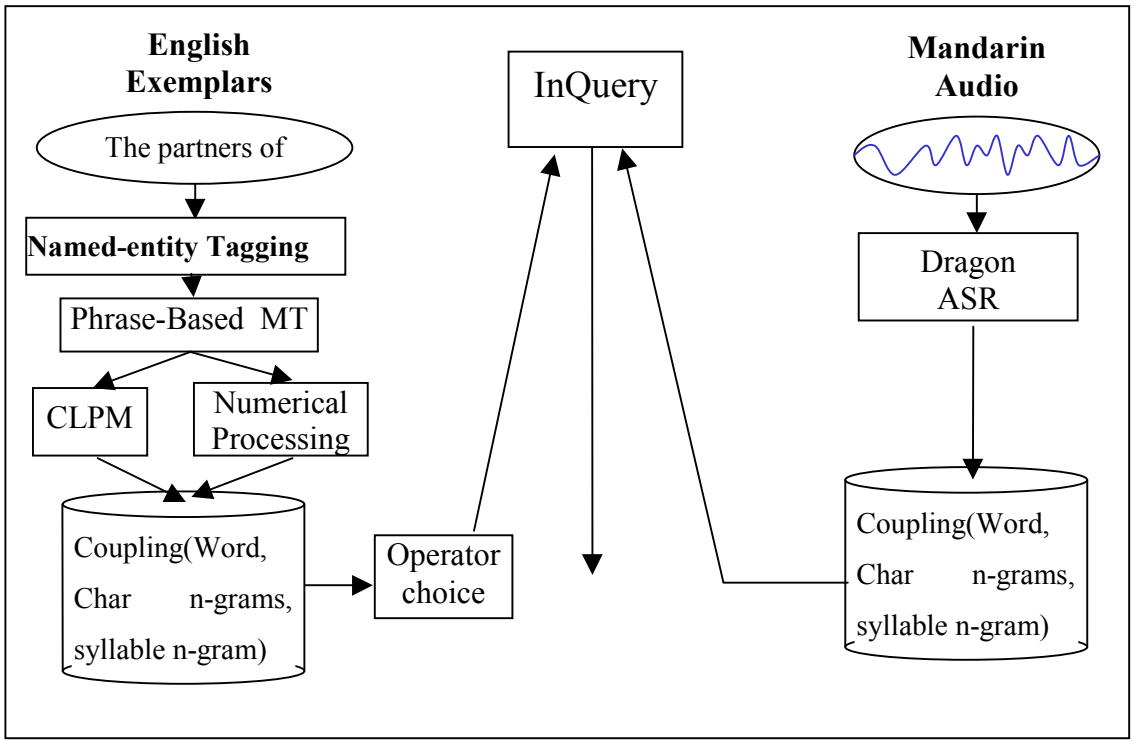


Figure 10.5 Incorporating CLPM, numerics, coupling, and operators

10.7 Operation Swaps

One might believe that something as "trivial" as adding switches in order to alter operator choice might hardly merit a section unto itself. The concept itself is quite simple. Yet as is often the case, the simplest things conceptually often turn out to be the hardest things to implement. This is definitely true about adding operators.

We had decided to allow operation changes at multiple levels in our queries. Two of those levels have already been mentioned: namely, the level which wraps around syllable words and syllables in tight coupling, and the level which wraps around syllable n-grams. We also wanted to be able to change the operator that encompasses all of the possible translations for a single English term, as well as perhaps altering the operator that envelops all of the words of a query.

This idea certain seems useful and easy to understand. Unfortunately, our system was written largely in Perl, and the main body of our code did not have access to all of the decision places where operators would be placed. It therefore had to infer the appropriate places. Furthermore, besides adding the operators that were discussed in section 10.6, we also wanted to be able to take advantage of one more InQuery operator, namely the weighted sum (#wsum) operator, which computes the weighted averages of the terms within it. Weighted averages therefore require the existence of weights in addition to words, which, of course, added an additional complexity. Nonetheless, we were able to incorporate all of this functionality. Figure 10.5 describes the system with all of the components mentioned thus far.

10.8 Lattice and MEI Syllables

In addition to efforts in query processing, we also had efforts in improving the overall speech recognition. Our speech recognition efforts gave us the ability to represent documents either as a lattice or as MEI syllables instead of Dragon output. Details have been described in section 8. To incorporate the MEI syllables into the system, one has little more to do than change parameter settings. To incorporate the lattice into the system, we merely had to adjust the system so that when a user specified the #n operator, the depth of the lattice (squared) could be built factored in. These changes are depicted in Figure 10.6.

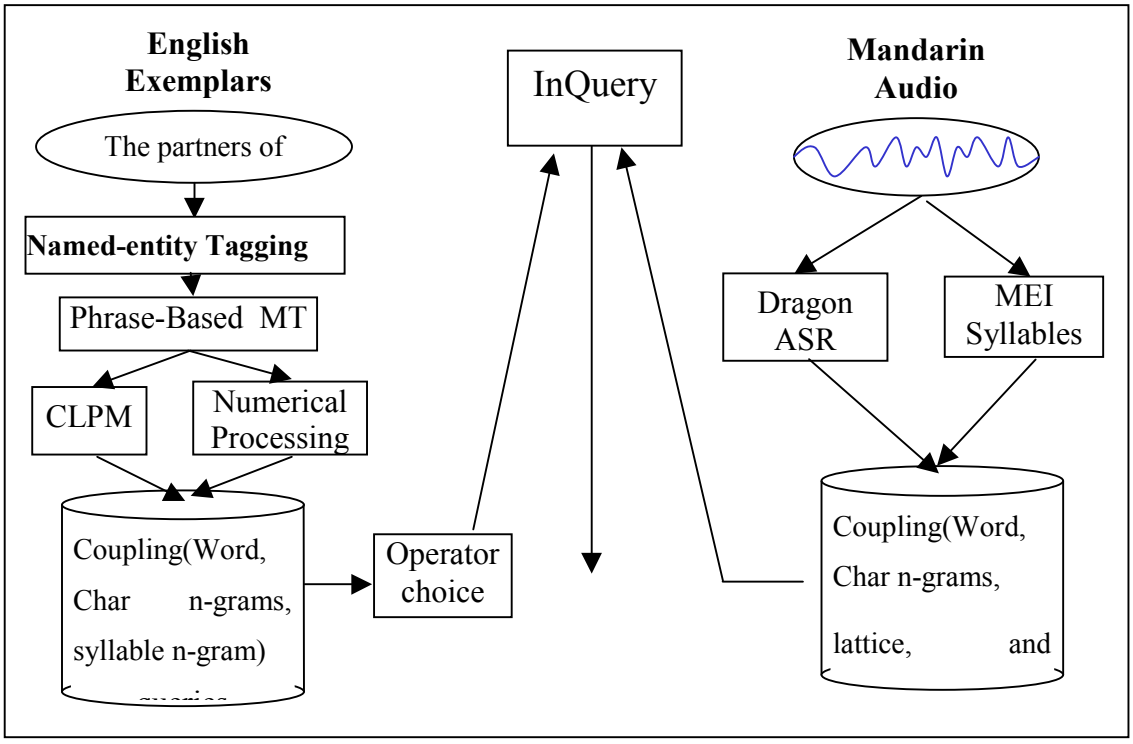


Figure 10.6 Changes to audio processing.

10.7 ASR Confusions

The vast majority of the MEI group efforts were spent in trying to overcome problems with translation. In particular, as has been seen, we had attempted to deal with multiple possible translations, names that were out of vocabulary, phrasing issues, and numerical conversions. Nonetheless, as the previous sections on coupling and speech recognition have suggested, MEI did make substantial effort as well in trying to overcome the ill effects of speech recognition. We were quite surprised to see that Dragon's recognition error rate was extremely low, and because it was so low, it is quite possible that all of the techniques we had applied up to this point may have compensated for these errors. Yet we wanted to look at one additional issue: speech recognition confusions.

The MEI team was of course equipped with the output of the Dragon recognizer on the Chinese audio. Yet as it has also been mentioned, we also had access to the scripts that the newscaster used to produce the original signals, which scripts one could think of as almost being perfect transcription. Therefore, we could use the scripts to identify those points in the Dragon transcripts where errors were being observed.

Suppose a word X should have been observed 50 times, but it was transcribed as X in 15 cases, as Y in 25 cases, and as Z in 10 other cases. Then one could potentially replace every occurrence of the word X in a query with a weighted sum

#wsum(50 15 X 25 Y 10 Z) .

This indicates that the denominator of the weighted average should be 50 and it also reports the respective weights for each of the possible transcriptions of X. For our purposes, we only incorporated confusions that occurred at least twice into the weighted sum, so essentially, we allowed no weights of one.

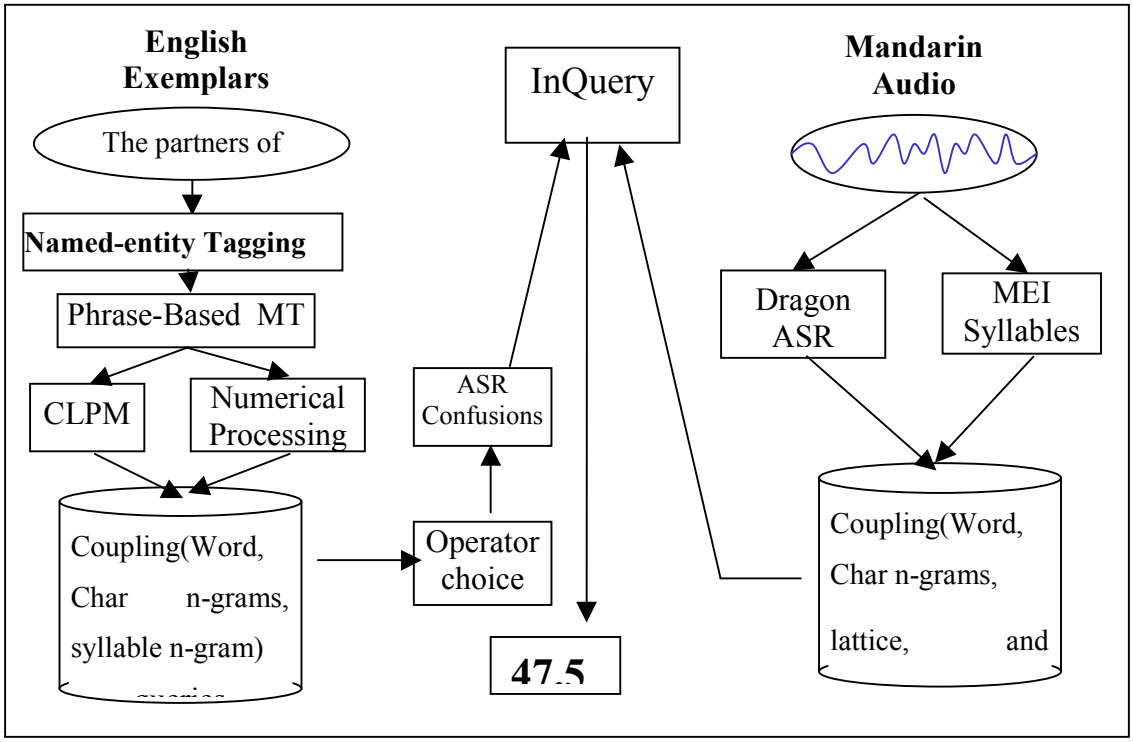


Figure 10.7 The complete system.

Figure 10.7 depicts the general architecture of our final system with the word confusion capability added. We were able to train the confusions using TDT2 data and then test without retraining using the TDT3 data. Although this notion of incorporating the ASR confusions seems logical, and in other applications it has seemed to yield improvements (Schone, et al., 1997; Franz, 1999), it gave our group absolutely no gain. We had observed a mean average precision of 0.475 on TDT3 when we had used character bigrams and this was exactly the same precision we observed after adding the confusions.

There are two possible reasons why this did not provide us with the same performance gain as others had observed. The first of these is that the speech recognition was so good, that most words were typically not confused. Another issue, though, is that when our system had completed the process of query development, each query was effectively a sea of words. We had eventually allowed the translation engine to report every possible translation for every query word. Hence, changing an occasional word into a weighted sum of confusions would be scarcely noticed in terms of the whole query.

Although we were disappointed that not all of our components gave the kinds of boosts that we had hoped for, we were still quite happy with the system we had built. We had nonetheless constructed a cross-lingual system whose mean average precision begins to

approach those of the word-only monolingual systems. Furthermore, in just the six short weeks allotted to the summer workshop, we were able to incorporate a significant number of components into our system and we were able to build everything into our system that we had originally hoped for.

11. Summary and Conclusions

In this project, we have designed and developed one of the first English-Chinese CL-SDR systems, and advocated a novel multi-scale paradigm for the task. Our approach aims to ameilorate inheret problems in the ask, including: open vocabularies in translation and recognition, ambiguities in Chinese word tokenization and Chinese homophones, speech recognition errors in document indexing and translation multiplicity or failures.

Multi-scale query and document processing utilizes both word and subword units to represent the query / documet. Subwords refer to both Chinese characters and syllables. The use of overlapping subword n-grams is effective in circumventing the problems listed above. We have also devised a multi-scale query formulation strategy, and the cross-lingual phonetic mapping procedue which can generate subword transliterations automatically, given the spellings of named entities.

We have experimented with multi-scale retrieval, which involved both tight and loose coupling strategies for word / subword fusion in retrieval. Extensive experiments were run on the TDT-2 and TDT-3 corpora. Our key findings include:

(i) character bigrams typically outperform words or syllable bigrams in retrieval;

(ii) fusion of word and subword units shows improvement in multi-scale retrieval tha either unit alone;

(iii) the use of a syllable lattice in spoken document indexing aims to provide alternate syllable hypothesis in case of recognition errors. However, retrieval performance shows degradation and the issue warrants further investigation;

(iv) balancing translated queries among the various translation alternatives is important and beneficial towards retrieval.

We have only but scratched the tip of the iceberg in this line of research and look forward to many follow up projects in the near future.

12. References

[Ballesteros and Croft, 1997] Ballesteros, L. and W. B. Croft, "Phrasal Translation and Query Expansion Techniques for Cross-Language Information Retrieval," Proceedings of ACM SIGIR, 1997.

[Callan et al., 1992] Callan, J. P., W. B. Croft, and S. M. Harding, "The INQUERY Retrieval System," Proceedings of the 3rd International Conference on Database and Expert Systems Applications, 1992.

[Caronnell et al., 1997] Carbonnell, J., Y. Yang, R. Frederking and R.D. Brown, "Translingual Information Retrieval: A Comparative Evaluation," Proceedings of IJCAI, 1997.

[Chen et al., 2000] Chen, B., H.M. Wang, and L.S. Lee, "Retrieval of Broadcast News Speech in Mandarin Chinese Collected in Taiwan using Syllable-Level Statistical Characteristics," Proceedings of ICASSP, 2000.

[Chien e t al., 2000] Chien, L. F., H. M. Wang, B. R. Bai, and S. C. Lin, "A Spoken-Access Approach for Chinese Text and Speech Information Retrieval," Journal of the American Society for Information Science, 51(4), pp. 313-323, 2000.

[Furui 1981] Furui, S., 1981. Cepstral analysis technique for automatic speaker verification. IEEE Trans. on Acoustics Speech and Signal Processing 29, 254-272.

[Furui 1986] Furui, S., 1986. Speaker-independent isolated word recognition using dynamic features of speech spectrum. IEEE Trans. on Acoustics Speech and Signal Processing 34, 52-59.

[Garafolo et al., 2000] Garofolo, J.S., Auzanne, G.P., Voorhees, E.M., "The TREC Spoken Document Retrieval Track: A Success Story," Proceedings of the Recherche d'Informations Assistée par Ordinateur: Content-Based Multimedia Information Access Conference , April 12-14, 2000,to be published.

[Junqua et al., 1993] Junqua J. C., Wakita H., and Hermansky H., 1993. Evaluation and optimization of perceptually-based ASR front-end. IEEE Trans. on Speech and Audio Processing 1, 39-48.

[Knight & Grael 1997] Knight, K. and J. Graehl, "Machine Transliteration," Proceedings of ACL, 1997.

[Levow & Oard, 2000] Levow, G. and D. Oard, "Translingual Topic Tracking with PRISE," Proceedings of the TDT Workshop, 2000.

[McCarely 1999] McCarley, S., "Should we Translate the Documents or the Queries in Cross-Language Information Retrieval," Proceedings of ACL, 1999.

[Meng et al., 2000] Meng, H., Khudanpur, S., Oard, D. W. and Wang, H. M., "Mandarin-English Information (MEI)," Working notes of the DARPA TDT-3 Workshop, 2000.

Ng, K., "Subword-based Approaches for Spoken Document Retrieval," Ph.D. Thesis, MIT, February 2000.

[Meng et al., 2000d] Meng, H., W. K. Lo, Y. C. Li, and P. C. Ching, "Multi-Scale Audio Indexing for Chinese Spoken Document Retrieval," Proceedings of the International Conference on Spoken Language Processing, 2000.

[Moreno et al., 1999] Moreno, P, J. M. Van Thong, B. Logan, B. Fidler, K. Maffey, and M. Moores, "SpeechBot: A Content-based Search Index for Multimedia on the Web," http://speechbot.research.compaq.com (white paper).

[Oard et al., 1999] Oard, D., J. Wang, D. Lin and I. Soboroff, "TREC-8 Experiments at Maryland: CLIR, QA and Routing," Proceedings of the 8th Text Retrieval Conference (TREC-8), pp. 623-636.

[Oard & Wang, 1999] Oard, D. and J. Wang, "NTCIR CLIR Experiments at the University of Maryland." Presented at the NACSIS Test Collection Information Retrieval Workshop in Tokyo, Japan in August, 1999.

[Pirkola 1998] Pirkola, A., “The effects of query structure and dictionary setups in dictionary-based cross-language information retrieval,” Proceedings of ACM SIGIR, 1998.

[Schuetze et al., 1995]

[Sheridan and Ballerini 1996] Sheridan P. and J. P. Ballerini, “Experiments in Multilingual Information Retrieval using the SPIDER System,” Proceedings of ACM SIGIR, 1996.

[Wang et al., 1999] Wang, H. M., "Retrieval of Mandarin Spoken Documents based on Syllable Lattice Matching," Proceedings of the International Workshop on Information Retrieval of Asian Languages, 1999.

[Wang 2000] Wang, Hsin-min, 2000. Experiments in Syllable-based Retrieval of Broadcast News Speech in Mandarin Chinese. Speech Communication 32, pp. 49-60.

[Voorhees 1995] Voorhees, E., “Learning Collection Fusion Strategies,” Proceedings of SIGIR, 1995.

[Wang 1999] Wang, H. M., “Retrieval of Mandarin Spoken Documents Based on Syllable Lattice Matching,” Proceedings of the Fourth International Workshop on Information Retrieval in Asian Languages, 1999.

[Wechsler & Schauble 1995] Wechsler, M. and P. Schaüble, “Speech Retrieval Based on Automatic Indexing,” Proceedings of MIRO-1995.

[Zhan et al., 1999] Zhan, P., Wegmann, S., and Gillick, L., 1999. Dragon Systems’ 1998 broadcast news transcription system for Mandarin. Proceedings of the DARPA Broadcast News Workshop.

Appendix A. Effects of Term Selection and Translation Correction in Crosslingual Spoken Document Retrieval

The MEI project provided us a great opportunity to test diverse ideas. In this section, we describe our research on the effects of term selection in forming queries in the original language (English), and the effects of translation correction in refining queries automatically translated into the target language (Chinese). Term selection intends to solve the problem of "what to translate", while translation correction intends to solve the problem of "what to translate to". Term selection techniques have been widely studied in the context of monolingual information retrieval, since whenever one needs to form a query, some sort of term selection strategy has to be explored. However, the upper bound of term selection has not been studied in the context of cross-language information retrieval (CLIR). On the other hand, translation correction is an issue unique to CLIR. Although various techniques have been proposed and tested for automatic translation processing (mainly dealing with the issues of out-of-vocabulary (OOV) terms and translation ambiguity), we are still not clear about: (1) at best how much can such translation correction techniques contribute? And (2) between failure to cover important translation and inclusion of irrelevant translation in the translation dictionary, which hurts us more? The experiments reported here seek to answer these questions.

A.1 Term selection: Manual versus Automatic

Previous experiment [Levow and Oard, 2000] showed that CHI-squared query term selection can significantly improve the effectiveness of CLIR. In our preliminary experiments, we tested the same technique of term selection and gained a boost to our results. This encourages us to further investigate the issue of query term selection. At best how well can automatic term selection techniques perform? In other words, we want to find an upper bound of query term selection, so that we can evaluate the performance of our term selection technique, and suggest how much can we do in order to further improve its effectiveness.

In doing this, we used a set of 17 randomly selected TDT-2 document exemplars as queries, with each of them corresponding to a TDT-2 topic. These documents vary in length, with an average length of about 500 words per document. The default setting of χ^2 term selection is that at most 180 unique words were selected for each query. This forms our baseline *long query* set. For comparative study, we manually formed another two sets of queries from the same document set. The process worked as follows. First, the experimenter read each document exemplar. Then he was asked to pick up the most informative terms (words or phrases) for that document exemplar. Selected terms were used to form a query representing that document. For the first set of queries, the experimenter was restricted to pick up no more than four terms per query. The set of queries was considered to be *short queries*,

as contrast to the baseline long query set. For the second set of queries, the experimenter was allowed to pick up as many terms as he wanted. This set of queries was called *medium queries*, since they were usually longer than short queries, but shorter than long queries.

It is generally believed that long queries often outperforms short queries if they are both automatically formed, since long queries contains more information than short queries. Our previous experiments in TREC-8 [Oard et al., 1999] and NTCIR [Oard & Wang, 2000] also confirmed this point.[13] Here we want to see if the same relationship holds between automatically formed long queries and manually formed shorter queries.

These three sets of queries were finally fed to our automatic dictionary-based query translation (DQT) routine. The results were three sets of queries in Chinese. Multiple translations for the same term were grouped with INQUERY's synonym operator (#SYN). At most 5 translations were adopted according to their position in the bilingual dictionary. Finally, the translated queries were used to retrieve document collections in Chinese[14].

Mean average precision (MAP) and average query length for each set of queries are shown in Table A.1. In this table, the third row refers to the average length of the query set before it was translated into Chinese, i.e., it is the average number of English terms per query. The fourth row "QL_C" is the average length of the query set after it was translated (the average number of Chinese words per query). Ratio of these two rows gives us a basic idea of on average how many translations of an English term were used in our experiment.

Query	**Short Queries**	**Medium Queries**	**Long Queries**
Mean Average Precision	0.2388	0.2686	0.2939
Average English Query Length[15]	4	7	223
Average Translated Chinese Query Length	10	16	571

Table A.1 Term selection: mean average precision and query lengths.

[13] There are at least two differences here between TREC/NTCIR and TDT. Short queries in TREC/NTCIR are usually created from the "title" field and/or the "description" field of the original topics. These two fields, however, are intentionally created by humans when the topics were produced. Also, relevance judgement in TREC and NTCIR are subject-oriented, while in TDT it is event-oriented.

[14] For more information about query structure and INQUERY IR engine, please refer to the related sections in this report.

	Medium Queries	Long Queries
Short Queries	0.048	0.429
Medium Queries		0.716

Table A.2 Statistical significance in performance differences among the use of short / medium / long queries for retrieval. A paired two-tailed t-test is used at 95% significance level.

We notice that mean average precision increases as the average length of queries increases. This is consistent with previous research on automatically formed long queries and short queries. Statistical tests indicate that putting restriction on number of terms selected will lead to significant degrade on IR effectiveness than without such restriction. However, we couldn't see such phenomena in comparing manual queries (short and medium queries) with automatic formed long queries (see Table A.2).

In conclusion, our χ^2 term selection technique performs quite well. It performs at least as effective as manual term selection. It seems that we have good reasons to believe that such term selection strategy can include all the information that otherwise a human searcher wants to have in forming queries from document exemplars.

A.2 Translation Correction

Automatic dictionary-based query translation (DQT) tends to produce two types of errors: translation insertion and translation deletion. By translation insertion we mean use of translations that are irrelevant to the context. This usually happens when a term has multiple translations in the target language. By translation deletion we mean failure to cover appropriate translations. This may happen as one of the two cases: there is no translation for a term since the term is not covered by the dictionary (an OOV term), or, there are some translations for a term but the important translations are not covered (incomplete translation coverage). In this section, we want to see by manually correcting these two types of errors how much we can gain. The result can be regarded as an upper bound for automatic DQT techniques.

We performed manual translation correction on translated versions of the above three sets of queries. Correspondingly, our query sets include: (i) short queries with manual translation corrections, (ii) medium queries with manual translation corrections, and (iii) long queries with manual translation corrections. Our objectives here are (1) to prove that translation correction improves the effectiveness of CLIR, and (2) to find an upper bound for our CLIR experiments.

---

[15] Notice that this number is bigger than 180. This is because some words in CHI-squared term list may appear more than once in the document exemplar, and in this case these terms were counted more than once. Strictly, the number of 223 is the average tokens per query.

Query	Short queries with manual translation correction	Medium queries with manual translation correction	Long queries with manual translation correction
MAP	0.4168	0.4988	0.5112
Query Lengths	5	8	338

Table A.3 Queries with manual translation corrections: mean average precision and query lengths.

	SQ_AT	MQ_AT	LQ_AT	MQ_MTC	LQ_MTC
SQ_MTC	0.025*			0.045*	0.076
MQ_MTC		0.015*			0.790
LQ_MTC			0.003*		

Table A.4 Statistical significance tests for performance differences among short/medium/long queries, with and without manual translation corrections. SQ abbreviates short queries, MQ for medium queries, LQ for long queries, AT for automatic translations and MTC for manual translated corrections. A paired two-tailed t-test at 95% significance level was used.

Experiment results are listed in Table A.3. We observed more than 100% improvement when translation correction is performed on each set of queries. The mean average precision for each query set in Table A.3 can be regarded as the upper bound for automatic translation of that query set. Not surprisingly, all these improvements are significant (see Table A.4). Also, for manually corrected queries, we continue to see the trend of improvement as query length increases. Again, we noticed that just like in the experiment of term selection, in translation correction medium queries (without restriction on number of selected terms) significantly outperform short queries (with such restriction).

We conclude that manual translation correction significantly improves the effectiveness of CLIR. With such technique, similar trend of improvement was observed as query length increases from manually formed short queries to automatically formed long queries. This suggests that while there is little room for improvement of automatic query term selection techniques, there are still a lot of things for us to do in order to improve the quality of automatic dictionary-based query translation.

A.3 Translation errors: Insertion versus Deletion

The significant improvement we gained when manual translation correction was applied encouraged us to further investigate this process. As described above, manual translation correction attempts to overcome two translation errors: translation insertion and translation deletion. In this experiment, we want to closely study the effects of these two errors on the

performance of CLIR. We only used the long query set (as called here "Both") simply because it has better performance than the other two sets. With this query set, we created another two sets of queries: one with only translation insertion errors corrected (as called here "Ins_free"), and the other with only translation deletion errors corrected (as called "Del_free"). That is, we manually added translations for some terms in the automatically translated long queries when the translation dictionary failed to cover these translation but the experimenter thought they were important. On the other hand, we manually removed some translations because the experimenter thought they were irrelevant.

The whole process of query processing in this experiment is illustrated in Figure A.1. We first performed CHI-squared term selection on the 17 randomly selected document exemplars. This produced long queries in English. We then passed this set of queries to automatic DQT routine. The output was our baseline query set ("Base"). Baseline queries contained both translation insertion errors and translation deletion errors. Next, we either manually removed insertion errors from the baseline to create a set of insertion error-free queries ("Ins_free", represented by thin arrow lines in the chart), or manually added translations not being covered to create a set of deletion error-free queries ("Del_free", represented by parallel line arrows in the chart). Finally, we performed deletion error correcting on "Ins_free" queries to create "Both" queries (represented by thick line arrows). It was In all query sets, INQUERY's synonym operator (#SYN) were used to group multiple translations for the same term.

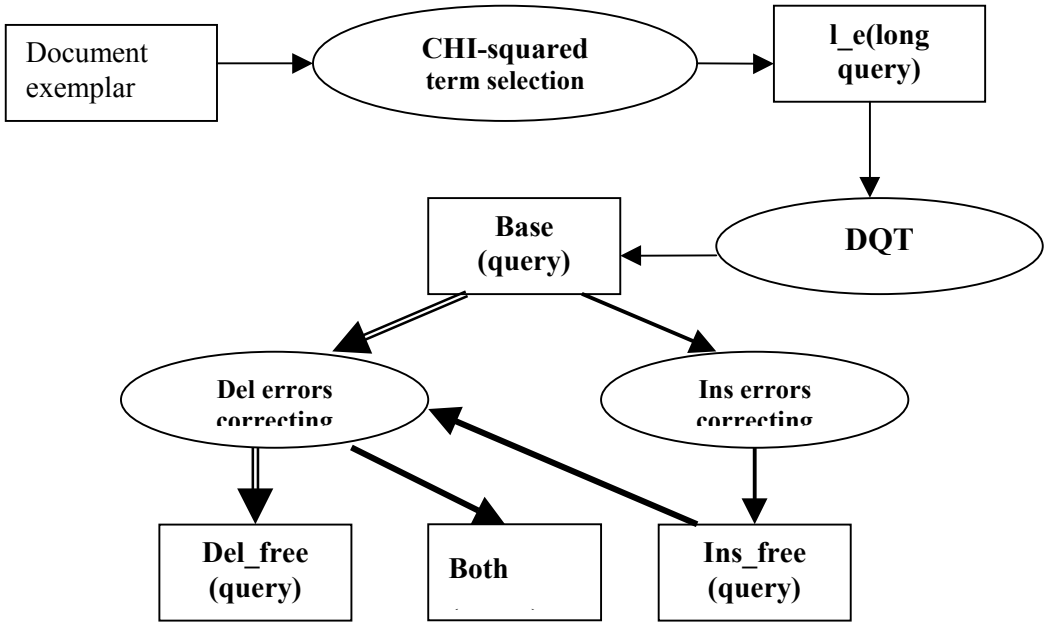


Figure A.1: Flow chart for query processing.

Table A.5 shows the mean average precision for each set of queries. Statistical significant tests are shown in Table A.6. As we have expected, we observed significant improvement when either types of translation errors were corrected. Correcting both types of errors was significantly better than correcting either of them. No significance was detected between correcting translation insertion errors and translation deletion errors. This indicates that in the dictionary we used in the experiment, both the errors of failing to cover important translation and the errors of including irrelevant translation can hurt us significantly.

Query	Base	Del_free	Ins_free	Both
MAP	0.2939	0.3667	0.3633	0.5112

Table A5. Translation insertion and deletion: mean average precision (MAP)

	Del_free	Ins_free	Both
Ins_free	0.966		
Both	0.010*	0.013*	
Base	0.009*	0.044*	0.001*

Table A.6 Statistical significance tests for performance differences among queries which are free of translation deletion errors, translation insertion errors, or both. A paired two-tailed t-test is used with a significance level of 95%.

A.4 Summary

This appendix describes our work in manual query term selection and its comparison with automatic CHI-squared term selection technique. Our experiment results indicate there is no significant difference between these two process. Based on this, it seems safe to say that our automatic χ^2 term selection technique performs quite well, given the restriction that terms selected must appear in the original document exemplars.

We also studied the problem of failure to cover important translations and inclusion of irrelevant translations in the translation dictionary, and more importantly, how much we can gain in solving these two problems separately and jointly. Our experiment results show that both problems hurts us a lot. Solving either of them significantly improve the effectiveness of CLIR, and solving both improves even greater.

One major limitation of our experiments is we applied only one experimenter in manual term selection and translation correction. This experimenter, a native Chinese speaker, is currently a doctoral student in information science in the United States. Inevitably, his English comprehension has influence on selection of query terms. However, since the document exemplars we used in the experiments are newswire stories. Comprehension of

such documents should not be a problem. But still, we tend to assume that had a native English speaker had been recruited for such task, the results would have been slightly different. On the other hand, even for native English speakers, different people may have different understanding of the same document. Therefore, experiment results could be different among them. Same problem exists for translation correction. There are at least two ways to overcome this limitation if we can have a group of subjects involved in the same experiments. The first way is to have them work together so that only the terms/translations that all subjects agree on are used. The second way is to have them work separately and some sort of quantitative method is used to generalize their results.

Acknowledgments

We wish to aknowledge our sponsors, Johns Hopkins University and the National Science Foundation for the support of this Workshop. To Fred Jelinek and his staff at the Center for Language and Speech Processing, for running the Workshop so well and providing all the rich resources to our research. To LDC, for making available the TDT-2 and TDT-3 corpora especially for our research. To Hsin Hsi Chen, for providing his partial name list. To John Garafolo, and Jim Allan, for their comments and advice. We have all worked hard and learnt a lot from Workshop 2000, and feel that this is an extremely worthwhile experience.