The Web as a Parallel Corpus

Philip Resnik[*]
University of Maryland

Noah A. Smith[†]
Johns Hopkins University

Parallel corpora have become an essential resource for work in multilingual natural language processing. In this article, we report on our work using the STRAND system for mining parallel text on the World Wide Web, first reviewing the original algorithm and results and then presenting a set of significant enhancements. These enhancements include the use of supervised learning based on structural features of documents to improve classification performance, a new content-based measure of translational equivalence, and adaptation of the system to take advantage of the Internet Archive for mining parallel text from the Web on a large scale. Finally, the value of these techniques is demonstrated in the construction of a significant parallel corpus for a low-density language pair.

1. Introduction

Parallel corpora—bodies of text in parallel translation, also known as **bitexts**—have taken on an important role in machine translation and multilingual natural language processing. They represent resources for automatic lexical acquisition (e.g., Gale and Church 1991; Melamed 1997), they provide indispensable training data for statistical translation models (e.g., Brown et al. 1990; Melamed 2000; Och and Ney 2002), and they can provide the connection between vocabularies in cross-language information retrieval (e.g., Davis and Dunning 1995; Landauer and Littman 1990; see also Oard 1997). More recently, researchers at Johns Hopkins University and the University of Maryland have been exploring new ways to exploit parallel corpora in order to develop *monolingual* resources and tools, using a process of annotation, projection, and training: Given a parallel corpus in English and a less resource-rich language, we project English annotations across the parallel corpus to the second language, using word-level alignments as the bridge, and then use robust statistical techniques in learning from the resulting noisy annotations (Cabezas, Dorr, and Resnik 2001; Diab and Resnik 2002; Hwa et al. 2002; Lopez et al. 2002; Yarowsky, Ngai, and Wicentowski 2001; Yarowsky and Ngai 2001; Riloff, Schafer, and Yarowsky 2002).

For these reasons, parallel corpora can be thought of as a critical resource. Unfortunately, they are not readily available in the necessary quantities. Until very recently, for example, statistical work in machine translation focused heavily on French-English translation because the Canadian parliamentary proceedings (Hansards) in English and French were the only large bitext available. Things have improved somewhat, but it is still fair to say that for all but a relatively few language pairs, parallel corpora tend to be accessible only in specialized forms such as United Nations proceedings (e.g., via the Linguistic Data Consortium, ⟨http://www.ldc.upenn.edu⟩), religious texts (Resnik, Olsen, and Diab 1999), localized versions of software manuals (Resnik and

[*] Department of Linguistics and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742. E-mail: resnik@umd.edu

[†] Department of Computer Science and Center for Language and Speech Processing, Johns Hopkins University, Baltimore, MD 21218. E-mail: nasmith@cs.jhu.edu

Melamed 1997; Menezes and Richardson 2001), and the like. Even for the top handful of majority languages, the available parallel corpora tend to be unbalanced, representing primarily governmental or newswire-style texts. In addition, like other language resources, parallel corpora are often encumbered by fees or licensing restrictions. For all these reasons, it is difficult to follow the "more data are better data" advice of Church and Mercer (1993), abandoning balance in favor of volume, with respect to parallel text.

Then there is the World Wide Web. People tend to see the Web as a reflection of their own way of viewing the world—as a huge semantic network, or an enormous historical archive, or a grand social experiment. We are no different: As computational linguists working on multilingual issues, we view the Web as a great big body of text waiting to be mined, a huge fabric of linguistic data often interwoven with parallel threads.

This article describes our techniques for mining the Web in order to extract the parallel text it contains. It presents, in revised and considerably extended form, our early work on mining the Web for bilingual text (STRAND) (Resnik 1998, 1999), incorporating new work on content-based detection of translations (Smith 2001, 2002), and efficient exploitation of the Internet Archive. In Section 2 we lay out the STRAND architecture, which is based on the insight that translated Web pages tend quite strongly to exhibit parallel *structure*, permitting them to be identified even without looking at content; we also show how we have improved STRAND's performance by training a supervised classifier using structural parameters rather than relying on manually tuned thresholds. In Section 3 we present an approach to detecting translations that relies entirely on *content* rather than structure, demonstrating performance comparable to STRAND's using this orthogonal source of information. In Section 4 we describe how we have adapted the STRAND approach to the Internet Archive, dramatically improving our ability to identify parallel Web pages on a large scale. Section 5 puts all the pieces together, using structural and combined content-structure matching of pages on the Internet Archive in order to obtain a sizable corpus of English-Arabic Web document pairs. Finally we present our thoughts on future work and conclusions.

2. The STRAND Web-Mining Architecture

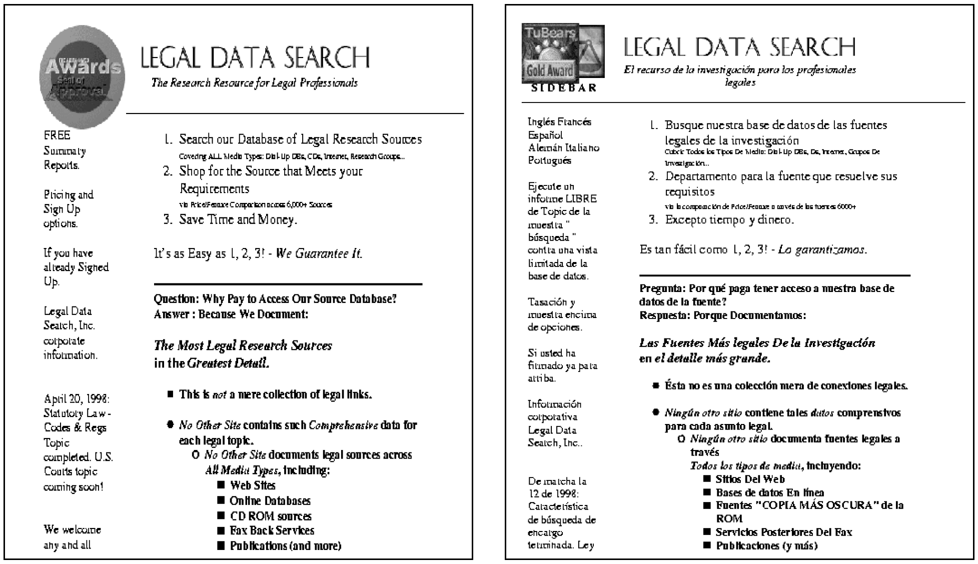
STRAND (Resnik 1998, 1999) is an architecture for structural translation recognition, acquiring natural data. Its goal is to identify pairs of Web pages that are mutual translations. In order to do this, it exploits an observation about the way that Web page authors disseminate information in multiple languages: When presenting the same *content* in two different languages, authors exhibit a very strong tendency to use the same document *structure* (e.g., Figure 1). STRAND therefore locates pages that *might* be translations of each other, via a number of different strategies, and filters out page pairs whose page structures diverge by too much.

In this section we describe how STRAND works, and we also discuss several related Web-mining methods, focusing on the overall architecture these systems have in common and the important system-specific variations. We then show how tuning STRAND's structural parameters using supervised training can significantly increase its performance.

2.1 STRAND

Finding parallel text on the Web consists of three main steps:

- Location of pages that might have parallel translations



LEGAL DATA SEARCH

The Research Resource for Legal Professionals

FREE Summary Reports.

Pricing and Sign Up options.

If you have already Signed Up.

Legal Data Search, Inc. corporate information.

April 20, 1998: Statutory Law - Codes & Regs Topic completed. U.S. Courts topic coming soon!

We welcome any and all

1. Search our Database of Legal Research Sources
   Covering ALL Media Types: Dial-Up DBs, CDs, Internet, Research Groups...
2. Shop for the Source that Meets your Requirements
   via Price/Feature Comparison across 6,000+ Sources
3. Save Time and Money.

It's as Easy as 1, 2, 3! - *We Guarantee It.*

Question: Why Pay to Access Our Source Database?
Answer : Because We Document:

The Most Legal Research Sources
in the *Greatest Detail.*

- **This is *not* a mere collection of legal links.**
- ***No Other Site* contains such *Comprehensive* data for each legal topic.**
  - ***No Other Site* documents legal sources across *All Media Types*, including:**
    - **Web Sites**
    - **Online Databases**
    - **CD ROM sources**
    - **Fax Back Services**
    - **Publications (and more)**

LEGAL DATA SEARCH

El recurso de la investigación para los profesionales legales

SIDEBAR

Inglés Francés Español Alemán Italiano Portugués

Ejecute un informe LIBRE de Topic de la muestra " búsqueda " contra una vista limitada de la base de datos.

Tasación y muestra encima de opciones.

Si usted ha firmado ya para arriba.

Información corporativa Legal Data Search, Inc..

De marcha la 12 de 1998: Característica de búsqueda de encargo terminada. Ley

1. Busque nuestra base de datos de las fuentes legales de la investigación
   Cubrir Todos los Tipos De Medio: Dial-Up DBs, Cds, Internet, Grupos De Investigación...
2. Departamento para la fuente que resuelve sus requisitos
   vía la comparación de Price/Feature a través de las fuentes 6000+
3. Excepto tiempo y dinero.

Es tan fácil como 1, 2, 3! - *Lo garantizamos.*

Pregunta: Por qué paga tener acceso a nuestra base de datos de la fuente?
Respuesta: Porque Documentamos:

Las Fuentes Más legales De la Investigación
en *el detalle más grande.*

- **Ésta no es una colección mera de conexiones legales.**
- ***Ningún otro sitio* contiene tales *datos* comprensivos para cada asunto legal.**
  - ***Ningún otro sitio* documenta fuentes legales a través**
    ***Todos los tipos de medio*, incluyendo:**
    - **Sitios Del Web**
    - **Bases de datos En línea**
    - **Fuentes "COPIA MÁS OSCURA" de la ROM**
    - **Servicios Posteriores Del Fax**
    - **Publicaciones (y más)**

Figure 1
Example of a candidate pair.

- Generation of candidate pairs that might be translations
- Structural filtering out of nontranslation candidate pairs

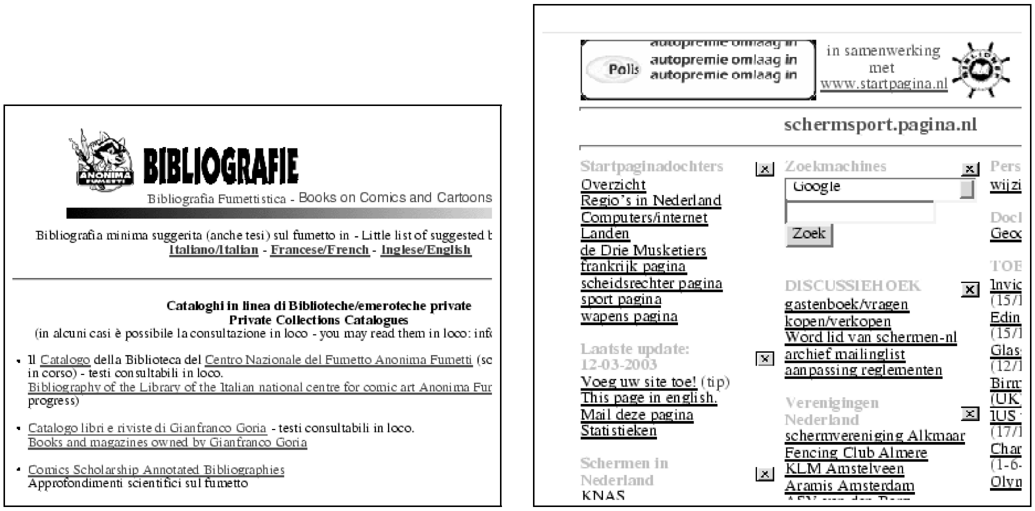
We consider each of these steps in turn.

2.1.1 Locating Pages. The original STRAND architecture accomplished the first step by using the AltaVista search engine's ⟨http://www.av.com⟩ advanced search to search for two types of Web pages: parents and siblings.

A **parent** page is one that contains hypertext links to different-language versions of a document; for example, if we were looking for English and French bitexts, the page at the left in Figure 2 would lead us to one such candidate pair. To perform this search for the English-French language pair, we ask AltaVista for pages in any language that satisfy this Boolean expression: `(anchor:"english" OR anchor:"anglais") AND (anchor:"french" OR anchor:"français")`. A 10-line distance filter is used to restrict attention to pages on which the English and French pointers occur reasonably close to one another—specifically, those for which the regular expression (in Perl) `/(english|anglais)/` is satisfied within 10 lines of the Perl regular expression `/(french|fran\w+ais)/` in the HTML source. This helps filter out a page that contained, for example, a link to "English literature courses" and also contained an unrelated link to "French version" at the top.

A **sibling** page is a page in one language that itself contains a link to a version of the same page in another language; for example, the page at the right of Figure 2 contains a link on the left that says "This page in english." To perform this search for English pages matching a given French page, we request pages in French that match the Boolean expression `anchor:"english" OR anchor:"anglais"`.

More recent versions of STRAND (unpublished) have added a "spider" component for locating pages that might have translations. Given a list of Web sites thought to contain bilingual text for a given language pair (e.g., sites identified using the AltaVista-based search), it is possible to download all the pages on each site, any



BIBLIOGRAFIE

Bibliografia Fumettistica - Books on Comics and Cartoons

Bibliografia minima suggerita (anche tesi) sul fumetto in - Little list of suggested b

Italiano/Italian - Francese/French - Inglese/English

Cataloghi in linea di Biblioteche/emeroteche private
Private Collections Catalogues
(in alcuni casi è possibile la consultazione in loco - you may read them in loco: inf

- Il Catalogo della Biblioteca del Centro Nazionale del Fumetto Anonima Fumetti (sc in corso) - testi consultabili in loco.
  Bibliography of the Library of the Italian national centre for comic art Anonima Fur progress)
- Catalogo libri e riviste di Gianfranco Goria - testi consultabili in loco.
  Books and magazines owned by Gianfranco Goria
- Comics Scholarship Annotated Bibliographies
  Approfondimenti scientifici sul fumetto

Polis autopremie omlaag in
autopremie omlaag in

in samenwerking met www.startpagina.nl

schermsport.pagina.nl

Startpaginadochters ☒
Overzicht
Regio's in Nederland
Computers/internet
Landen
de Drie Musketiers
frankrijk pagina
scheidsrechter pagina
sport pagina
wapens pagina

Laatste update:
12-03-2003 ☒
Voeg uw site toe! (tip)
This page in english.
Mail deze pagina
Statistieken

Schermen in
Nederland ☒
KNAS

Zoekmachines ☒
Google
Zoek

DISCUSSIEHOEK ☒
gastenboek/vragen
kopen/verkopen
Word lid van schermen-nl
archief mailinglist
aanpassing reglementen

Verenigingen
Nederland ☒
schermvereniging Alkmaar
Fencing Club Almere
KLM Amstelveen
Aramis Amsterdam

Pers
wijzi

Docl
Geo

TOE
Invic
(15/1
Edin
(15/1
Glas
(12/1
Birn
(UK
1US
(17/1
Char
(1-6-
Olyn

Figure 2
Excerpts from a parent page (left) and a sibling page (right). The parent page is in Italian and contains links marked "Italiano/Italian," "Francese/French," and "Inglese/English." The sibling page is in Dutch and contains a link marked "This page in english" in the leftmost column.

of which might have a translation on that site. Although simple to implement, this method of locating pages shifts the burden of narrowing down the possibilities to the process of generating candidate document pairs. The results reported here do not make use of the spider.

2.1.2 Generating Candidate Pairs. Pairing up potentially translated pages is simple when a search engine has been used to generate parent or sibling pages: One simply pairs the two child pages to which the parent links, or the sibling page together with the page to which it links.

When all the pages on a site are under consideration, the process is rather different. The simplest possibility is to separate the pages on a site into the two languages of interest using automatic language identification (Ingle 1976; Beesley 1988; Cavnar and Trenkle 1994; Dunning 1994), throwing away any pages that are not in either language, and then generate the cross product. This potentially leads to a very large number of candidate page pairs, and there is no particular reason to believe that most of them are parallel translations, other than the fact that they appear on the same Web site. The spider component of STRAND adds a URL-matching stage, exploiting the fact that the directory structure on many Web sites reflects parallel organization when pages are translations of each other. Matching is performed by manually creating a list of substitution rules (e.g., `english` → `big5`),[1] and for each English URL, applying all possible rules to generate URLs that might appear on the list of pages for the other language. If such a URL is found, the pair with similar URLs is added to the list of candidate document pairs. For example, suppose an English-Chinese site contains a page with URL ⟨http://mysite.com/english/home_en.html⟩, on which one combination of substitutions might produce the URL ⟨http://mysite.com/big5/home_ch.html⟩. The original page and the produced URL are probably worth considering as a likely candidate pair.

[1] Big5 is the name of a commonly used character encoding for Chinese.

Owing to the combinatorics (an exponential number of possible substitutions), only a fixed number of substitution combinations can be tried per English URL; however, in Section 4.3 we describe a more scalable URL-matching algorithm.

Another possible criterion for matching is the use of document lengths. Texts that are translations of one another tend to be similar in length, and it is reasonable to assume that for text E in language 1 and text F in language 2, length$(E) \approx C \cdot$length(F), where C is a constant tuned for the language pair. The use of a document length filter is described in Smith (2001), in which such a filter is shown, at the sentence level, to reduce the size of the search space exponentially in the confidence p in a $(1-p)$ confidence interval for a linear regression model with only linear loss of good pairs.

2.1.3 Structural Filtering. The heart of STRAND is a structural filtering process that relies on analysis of the pages' underlying HTML to determine a set of pair-specific structural values, and then uses those values to decide whether the pages are translations of one another. The first step in this process is to linearize the HTML structure and ignore the actual linguistic content of the documents. We do not attempt to exploit nonlinear structure (e.g., embedded chunks), for two reasons. First, we suspect that many HTML authors use tags for formatting text rather than for indicating document structure; therefore any "tree" structure is likely to be inconsistent or poorly matched. Second, we required the matching algorithm to be fast, and algorithms for aligning tree structures are more demanding than those for linear structures.

Both documents in the candidate pair are run through a markup analyzer that acts as a transducer, producing a linear sequence containing three kinds of token:

`[START:element_label]`	e.g., `[START:A]`, `[START:LI]`
`[END:element_label]`	e.g., `[END:A]`
`[Chunk:length]`	e.g., `[Chunk:174]`

The chunk length is measured in nonwhitespace bytes, and the HTML tags are normalized for case. Attribute-value pairs within the tags are treated as nonmarkup text (e.g., `<FONT COLOR="BLUE">` produces `[START:FONT]` followed by `[Chunk:12]`).

The second step is to align the linearized sequences using a standard dynamic programming technique (Hunt and McIlroy 1975). For example, consider two documents that begin as follows:

<HTML>	<HTML>
<TITLE>Emergency Exit</TITLE>	<TITLE>Sortie de Secours</TITLE>
<BODY>	<BODY>
<H1>Emergency Exit</H1>	Si vous êtes assis à
If seated at an exit and	côté d'une . . .
⋮	⋮

The aligned linearized sequence would be as follows:

```
[START:HTML]         [START:HTML]
[START:TITLE]        [START:TITLE]
[Chunk:13]           [Chunk:15]
[END:TITLE]          [END:TITLE]
[START:BODY]         [START:BODY]
[START:H1]
[Chunk:13]
[END:H1]
[Chunk:112]          [Chunk:122]
```

Using this alignment, we compute four scalar values that characterize the quality of the alignment:

dp The difference percentage, indicating nonshared material (i.e., alignment tokens that are in one linearized file but not the other).

n The number of aligned nonmarkup text chunks of unequal length.

r The correlation of lengths of the aligned nonmarkup chunks.

p The significance level of the correlation *r*.

The difference percentage (*dp*) quantifies the extent to which there are mismatches in the alignment: sequence tokens on one side that have no corresponding token on the other side. In the example above, one document contains an `H1` header that is missing from the second document. Large numbers of such mismatches can indicate that the two documents do not present the same material to a great enough extent to be considered translations. This can happen, for example, when two documents are translations up to a point (e.g., an introduction), but one document goes on to include a great deal more content than another. Even more frequently, the difference percentage is high when two documents are prima facie bad candidates for a translation pair.

The number of aligned nonmarkup text chunks (*n*) helps characterize the quality of the alignment. The dynamic programming algorithm tries to optimize the correspondence of identical tokens, which represent markup.[2] As a side effect, the nonmarkup text chunks are placed in correspondence with one another (e.g., the "Emergency Exit" and "Sortie de Secours" chunks in the above example). The more such pairings are found, the more likely the candidate documents are to represent a valid translation pair.

The remaining two parameters (*r* and *p*) quantify the extent to which the corresponding nonmarkup chunks are correlated in length. When two documents are aligned with one another and are valid translations, there is a reliably linear relationship in the length of translated chunks of text: short pieces correspond with short pieces, medium with medium, and long with long. The Pearson correlation coefficient *r* for the lengths will be closer to one when the alignment has succeeded in lining up translated pieces of text, and the *p* value quantifies the reliability of the correlation; for example, the standard threshold of $p < .05$ indicates 95% confidence that the correlation was not obtained by chance.

In our original work, we used fixed thresholds, determined manually by inspection of development (nontest) data for English-Spanish, to decide whether a candidate pair should be kept or filtered out. Thresholds of $dp < 20\%$ and $p < 0.05$ were used.

2.2 STRAND Results

As with most search tasks, performance at finding parallel Web pages can be evaluated using standard measures of precision and recall and by combining those figures using the *F*-measure. It is not possible for us to measure recall relative to the entire set of document pairs that should have been found; this would require exhaustive evaluation using the entire Web, or pooling results from a large number of different systems, as is done in the TREC information retrieval evaluations. Therefore, recall in this setting is measured relative to the set of candidate pairs that was generated.

[2] "Nonmarkup" tokens with exactly the same length almost always turn out to be pieces of markup-related text (e.g., `key=value` pairs within HTML tags).

Since the “truth” in this task is a matter for human judgment, we rely on bilingual speakers to judge independently whether page pairs are actually translations of each other for any given test set. In our experience *no* bilingual speaker is completely comfortable saying that another person’s translation is a good translation, so in creating the gold standard, we instead ask, “Was this pair of pages intended to provide the same content in the two different languages?” Asking the question in this way leads to high rates of interjudge agreement, as measured using Cohen’s κ measure.

2.2.1 Using Manually Set Parameters. Using the manually set thresholds for *dp* and *n*, we have obtained 100% precision and 68.6% recall in an experiment using STRAND to find English-French Web pages (Resnik 1999). In that experiment, 326 candidate pairs, randomly selected from a larger set of 16,763 candidates, were judged by two human annotators. The humans agreed (i.e., both marked a page “good” or both marked a page “bad”) on 261 page pairs (86 “good” and 175 “bad”), and it is relative to those 261 that we compute recall. A modified version of STRAND was used to obtain English-Chinese pairs (see related work, below), and in a similar formal evaluation, we found that the resulting set had 98% precision and 61% recall for Chinese ⟨http://umiacs.umd.edu/~resnik/strand/⟩. Both these results are consistent with our preliminary findings for English-Spanish using a less rigorous evaluation (using the judgments of the first author rather than independent bilingual evaluators) and a very small test set; precision in this preliminary experiment was near ceiling and recall was in the vicinity of 60% (Resnik 1998).

2.2.2 Assessing the STRAND Data. Although our focus here is finding parallel text, not using it, a natural question is whether parallel text from the Web is in fact of value. Two sources of evidence suggest that it is.

First, Web-based parallel corpora have already demonstrated their utility in cross-language information retrieval experiments. Resnik, Oard, and Levow (2001) showed that a translation lexicon automatically extracted from the French-English STRAND data could be combined productively with a bilingual French-English dictionary in order to improve retrieval results using a standard cross-language IR test collection (English queries against the CLEF-2000 French collection, which contains approximately 21 million words from articles in *Le Monde*). During document translation, backing off from the dictionary to the STRAND translation lexicon accounted for over 8% of the lexicon matches (by token), reducing the number of untranslatable terms by a third and producing a statistically significant 12% relative improvement in mean average precision as compared to using the dictionary alone. Similarly, Nie and Cai (2001) have demonstrated improved cross-language IR results for English and Chinese using data gathered by the PTMiner system (Chen and Nie 2000), a related approach that we discuss in Section 2.3.

Second, since bag-of-words IR experiments are not very illuminating with respect to fluency and translation quality, we conducted a ratings-based assessment of English-Chinese data, asking two native Chinese speakers (who are fluent in English) to assign ratings to a set of English-Chinese items. The set contained:

- 30 human-translated sentence pairs from the FBIS (Release 1) English-Chinese parallel corpus, sampled at random.
- 30 Chinese sentences from the FBIS corpus, sampled at random, paired with their English machine translation output from AltaVista’s Babelfish ⟨http://babelfish.altavista.com⟩.

- 30 paired items from Chinese-English Web data, sampled at random from "sentence-like" aligned chunks as identified using the HTML-based chunk alignment process of Section 2.1.3.

The human-translated and machine-translated pairs were included in order to provide upper-bound and lower-bound comparisons. The items were presented to one judge in a random order, and to the other judge in the reverse order.

Chinese-English Web data were those collected by Jinxi Xu using a modified version of STRAND (see Section 2.3), excluding those that did not pass STRAND's structural filter with the manually set thresholds. Sentence-like chunk pairs were defined as those in which the English side was 5–50 whitespace-delimited tokens long and that began with an uppercase alphabetic character and contained at least one token from an English stop list.[3] This fairly strict filter provided a set of approximately 7,000 pairs from which the 30 test items were sampled.

Participants were asked to provide each pair of items with three ratings, assessing English fluency, Chinese fluency, and adequacy of the translation. The choice and wording of the ratings criteria were derived from the human evaluation measures proposed by Dabbadie et al. (2002), with the wording of the translation assessment criterion modified to eliminate references to the direction of translation. (See Appendix B.)

For all three measures, the two judges' ratings were significantly correlated ($p < 0.0001$). Figure 3 shows additional quantitative results of the assessment, comparing judgments among human-translated, Web-generated, and machine-translated data.

The ratings indicate that pairs from the Web contain on average somewhere between "mostly the same meaning" and "entirely the same meaning" (median 3.25).[4] In comparison, current commercial-quality machine translation output achieves performance only between "much of the same meaning" and "mostly the same meaning" (median 2.5). Moreover, it is very likely that the Web translation quality is an underestimate: Some of the low-scoring outliers within the Web data could be eliminated by using state-of-the-art sentence alignment techniques and automatic detection and

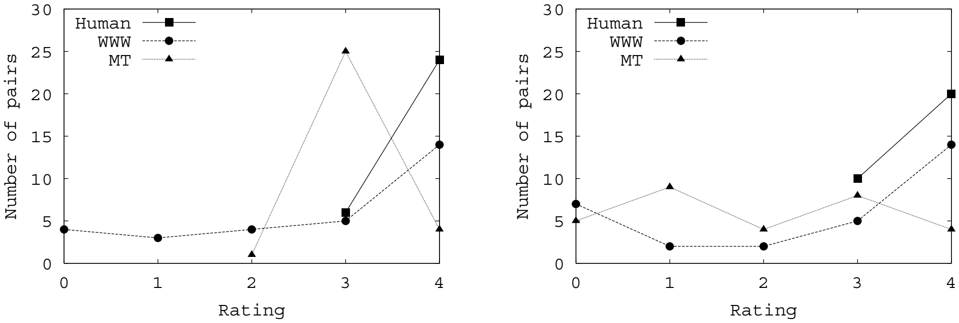


Figure 3
Translation adequacy ratings: Distribution over scores for human-translated (Human), Web-generated (WWW), and machine-translated (MT) data. The left plot provides results for judge 1, the right plot for judge 2.

[3] We also excluded pairs either side of which contained a curly bracket, since these were almost invariably fragments of Javascript code.

[4] The medians noted refer to the median of $\frac{R_1+R_2}{2}$ within the set, where R_n is the rating given by judge n. We use the median rather than the mean because it is less sensitive to outliers.

elimination of noisy pairs at the sentence level (cf. Nie and Cai [2001]). We observe that the distribution of scores for Web data peaks at the highest rating and that the data are in both cases modestly bimodally distributed. Machine-translated pairs, on the other hand, have generally lower quality. This suggests that high-quality parallel translations are present in this corpus and that poor-quality parallel translations are *very* poor (whether because of misalignment or simply because of poor translation quality at the document level) and might therefore be easily distinguishable from the better material. We plan to address this in future work.

Qualitatively, the results are a source of optimism about parallel data from the Web. Looking monolingually, fluency of the English side is statistically comparable to that of English sentences from the FBIS parallel corpus (essentially at ceiling), and on average the Chinese Web data are judged somewhere between "fairly fluent" and "very fluent," with the median at "very fluent" (only the second judge found the fluency of the Chinese Web sentences to be significantly worse than the human-generated Chinese sentences, Mann-Whitney test, $p < 0.02$).

2.2.3 Optimizing Parameters Using Machine Learning. Based on experiments with several language pairs, it appears that STRAND's structure-based filter consistently throws out around one-third of the candidate document pairs it has found in order to maintain its precision in the 98–100% range. It does so by respecting parameter thresholds that were determined manually using English-Spanish development data; the same parameters seem to have worked reasonably well not only for English-Spanish, but also for English-French and English-Chinese pairs. It is possible, however, that classification can be tuned for better performance. In order to investigate this possibility, we took a machine-learning approach: We used the four structural values (*dp*, *n*, *r*, and *p*) as features characterizing each document pair and treated the problem as a binary decision task, using supervised learning to make an attempt at better predicting human judgments.

Using the English-French data, we constructed a ninefold cross-validation experiment using decision tree induction to predict the class assigned by the human judges. The decision tree software was the widely used C5.0 ⟨http://www.rulequest.com/demoeula.html⟩. We used a decision tree learner because it is transparent (it is easy to see which features are being used to classify page pairs). In addition, a decision tree produced by C5.0 can be translated into a fast C program that is a rapid classifier of document pairs.

Each fold had 87 test items and 174 training items; the fraction of good and bad pairs in each fold's test and training sets was roughly equal to the overall division (33% to 67%, respectively). Precision and recall results are reported in Table 1, together with baseline results from STRAND's untuned classifier as reported above.

Looking at the decision trees learned, we see that they are very similar to one another. In every case a tree that looked like the following was learned:

```
if dp > 37 then BAD
else
     if n > 11 then GOOD
     else ...
```

where the remaining branch involved various additional partitionings of the candidate pairs that had few aligned text chunks (small *n*) and relatively low (but perhaps unreliable) difference percentage (*dp*). That branch handled around 10% of the document set and was prone to overtraining. (The documents handled by this branch were mostly marked "bad" by the judges but appear to have been difficult to classify based

Table 1
Effects of parameter tuning.

	Precision	Recall
Untuned	1.000	0.686
Fold 1	0.875	1.000
Fold 2	0.857	0.667
Fold 3	1.000	1.000
Fold 4	1.000	0.923
Fold 5	1.000	0.875
Fold 6	1.000	1.000
Fold 7	0.889	0.667
Fold 8	1.000	0.889
Fold 9	1.000	0.545
Average	0.958	0.841

on the structural features.) Note that the learned classifiers were substantially different from the heuristic threshold used earlier.

Without tuning, the manually set parameters result in good document pairs' being discarded 31% of the time. Our cross-validation results indicate that tuning the parameters cuts that figure in half: Only 16% of the good pairs will be discarded, at a cost of admitting 4 false positives from every 100 candidate pairs.

This approach is quite general and uses only a minimum of language-dependent knowledge. The features we are using are the same for any language pair. The tuning process needs to be done only once per language pair and requires only a few hours of annotation from untrained speakers of both languages to obtain the small labeled sample.

2.3 Related Work

Several other systems for discovering parallel text, developed independently, can be described as operating within the same three-stage framework as STRAND.

Parallel Text Miner (PTMiner) (Chen and Nie 2000) exploits already-existing Web search engines to locate pages by querying for pages in a given language that contain links to pages that are likely to be in the other language of interest. Once bilingual sites are located, they are crawled exhaustively. In order to generate candidate pairs, PTMiner uses a URL-matching process similar to the one described above; for example, the French translation of a URL like ⟨http://www.foo.ca/english-index.html⟩ might be ⟨http://www.foo.ca/french-index.html⟩. PTMiner's matching process uses a mapping of language-specific prefixes and suffixes and does not handle cases in which URL matching requires multiple substitutions. PTMiner also applies a length filter and automatic language identification to verify that the pages are in the appropriate languages. Chen and Nie report a 95% precise English-French corpus of 118MB/135MB of text and a 90% precise English-Chinese corpus of 137MB/117MB of text, based on inspection.

Later versions of PTMiner include a final filtering stage to clean the extracted corpus; Nie and Cai (2001) independently used features similar to those described here to eliminate noisy pairs. Specifically, they used a file length ratio filter, the proportion of sentences that are aligned to empty (after a sentence alignment algorithm is applied), and a criterion that rewards sentence pairs that contain elements from a bilingual dictionary. Nie and Cai showed that hand-tuned combination of these criteria improved the quality of their parallel English-Chinese corpus by 8% (F-measure) at the text level.

Bilingual Internet Text Search (BITS) (Ma and Liberman 1999) starts with a given list of domains to search for parallel text. It operates by sampling pages from each domain and identifying their languages; if a domain is deemed to be multilingual, all pages on the site are crawled exhaustively. BITS appears to consider all possible combinations of Web page pairs in the two languages (i.e., the full cross product within each site) and filters out bad pairs by using a large bilingual dictionary to compute a content-based similarity score and comparing that score to a threshold. For each page pair, the similarity score is

$$similarity(A, B) = \frac{\text{number of translation token pairs}}{\text{number of tokens in A}} \tag{1}$$

Translation token pairs are considered within a fixed window (i.e., a distance-based measure of co-occurrence is used).[5] In addition to cross-lingual lexical matching, BITS filters out candidate pairs that do not match well in terms of file size, anchors (numbers, acronyms, and some named entities), or paragraph counts. Using an English-German bilingual lexicon of 117,793 entries, Ma and Liberman report 99.1% precision and 97.1% recall on a hand-picked set of 600 documents (half in each language) containing 240 translation pairs (as judged by humans). This technique yielded a 63MB parallel corpus of English-German.

Other work on Web mining has been done by Jinxi Xu of BBN (personal communication), who began with our STRAND implementation and added a module for automatically learning string substitution patterns for URLs and also implemented a different dynamic programming algorithm for assessing structural alignment. Xu used the modified STRAND to obtain 3,376 Chinese-English document pairs, which we evaluated formally (see above), determining that the set has 98% precision and 61% recall.

In addition, STRAND has been reimplemented by David Martinez and colleagues at Informatika Fakultatea in the Basque Country (personal communication), in order to perform exploratory experiments for discovering English-Basque document pairs.

It is worth noting that STRAND, PTMiner, and BITS are all largely independent of linguistic knowledge about the particular languages, and therefore very easily ported to new language pairs. With the exception of the use of a bilingual dictionary (in BITS and later versions of PTMiner), these systems require, at most, a set of URL substring patterns for the URL pattern-matching stage (e.g., *big5* $\sim$ *english* in the example above; see further discussion in Section 4.3), and a modest amount of monolingual data for training n-gram-based language identifiers (typically 50,000 to 100,000 characters of text per language).

Word-level translations are worth exploiting when they are available. In Section 3 we describe a bitext-matching process using a content-based similarity score grounded in information theory, and in Section 5 we show how structural and content-based criteria can be combined in order to obtain performance superior to that obtained using either method alone.

[5] Many details of this technique are left unspecified in Ma and Liberman (1999), such as the threshold for the similarity score, the distance threshold, and matching of non-one-word to one-word entries in the dictionary.

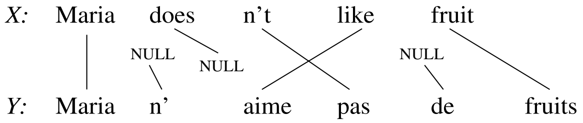


Figure 4
An example of two texts with links shown. There are seven link tokens, five of which are lexical (non-NULL) in *X* (the English side), six in *Y* (French).

3. Content-Based Matching

The approach discussed thus far relies heavily on document structure. However, as Ma and Liberman (1999) point out, not all translators create translated pages that look like the original page. Moreover, structure-based matching is applicable only in corpora that include markup, and there are certainly multilingual collections on the Web and elsewhere that contain parallel text without structural tags. Finally, other applications for translation detection exist, such as subdocument text alignment and cross-lingual duplicate detection (i.e., location of already-existing translations in a multilingual corpus). All these considerations motivate an approach to matching translations that pays attention to similarity of content, whether or not similarities of structure exist.

We present here a generic score of translational similarity that is based upon any word-to-word translation lexicon (hand-crafted or automatically generated, or a combination, and possibly highly noisy). The technique is shown to perform competitively to the structure-based approach of STRAND on the task of identifying English-French document translations.

3.1 Quantifying Translational Similarity

We define a cross-language similarity score, *tsim*, for two texts by starting with a generative, symmetric word-to-word model of parallel texts (Melamed's [2000] Method A). Let a **link** be a pair (x, y) in which x is a word in language L_1 and y is a word in L_2. The model consists of a bilingual dictionary that gives a probability distribution p over all possible link types. Within a particular link, one of the words may be NULL, but not both. In the generative process, a sequence of independent link tokens is sampled from the distribution. The model does not account for word order. An example of two texts with links is illustrated in Figure 4.

Next, we desire to compute the probability of the most probable link sequence that could have accounted for the two texts.[6] The probability of a link sequence is simply the product of the probabilities p of the links it contains. As noted by Melamed (2000), the problem of finding the best set of links is the maximum-weighted bipartite matching (MWBM) problem: Given a weighted bipartite graph $G = (V_1 \cup V_2, E)$ with edge weights $c_{i,j} (i \in V_1, j \in V_2)$, find a matching $M \subseteq E$ such that each vertex has at most one edge in M and $\sum_{e \in M} c_{i,j}$ is maximized. The fastest known MWBM algorithm runs in $O(ve + v^2 \log v)$ time (Ahuja, Magnati, and Orlin 1993). Applied to this problem, that is $O(\max(|X|, |Y|)^3)$, where X and Y are the text lengths in words.

To use MWBM to find the most probable link sequence, let the L_1 words be V_1 and the L_2 words be V_2. If two words x, y have $p(x, y) > 0$, an edge exists between them with weight $\log p(x, y)$. If a word x (or y) may link to NULL with nonzero probability, then that potential link is added as an additional edge in the graph between x (or y)

[6] Of course, all permutations of a given link sequence will have the same probability (since the links are sampled independently from the same distribution), so the order of the sequence is not important.

and a NULL vertex added to V_2 (or V_1). Each such x (or y) gets its own NULL vertex, so that multiple words may ultimately link to NULL. A sum of weights of links in a matching will be the log-probability of the (unordered) link sequence, and maximizing that sum maximizes the probability.

The similarity score should be high when many of the link tokens in the best sequence do *not* involve NULL tokens. Further, it should normalize for text length. Specifically, the score is

$$tsim = \frac{\log \Pr(\text{two-word links in best matching})}{\log \Pr(\text{all links in best matching})} \tag{2}$$

This score is an application of Lin's (1998) information-theoretic definition of similarity. Starting with a set of axioms relating intuitions about similarity to the mathematical notion of mutual information (Shannon 1948), Lin derives the measure

$$sim(X,Y) = \frac{\log \Pr(\text{common}(X,Y))}{\log \Pr(\text{description}(X,Y))} \tag{3}$$

where X and Y are any objects generated by a probabilistic model.

This technique of using a translation model to define translational similarity is generic to different sources of lexical translation information. An important feature is that it can be used with *any* symmetric translation model in which events can be divided into those that both sides of a bitext have in common and those that affect only one side.

The measure is simplified by assuming that all links in a given translation lexicon are equiprobable. The assumption reduces the formula for $tsim$ to

$$tsim = \frac{\text{number of two-word links in best matching}}{\text{number of links in best matching}} \tag{4}$$

The key reason to compute $tsim$ under the equiprobability assumption is that we need not compute the MWBM, but may find just the maximum cardinality bipartite matching (MCBM), since all potential links have the same weight. An $O(e\sqrt{v})$ (or $O(|X| \cdot |Y| \cdot \sqrt{|X| + |Y|})$ for this purpose) algorithm exists for MCBM (Ahuja, Magnati, and Orlin 1993). For example, if the matching shown in Figure 4 is the MCBM (for some translation lexicon), then $tsim(X,Y) = \frac{4}{7}$ under the simplifying assumption.

In earlier work (Smith 2002), we sought to show how multiple linguistic resources could be exploited in combination to recognize translation, and how the equiprobability assumption allowed straightforward combination of resources (i.e., set union of translation lexicon entries). In Section 3.2.1 we provide a clean solution to the problem of using unweighted translation lexicons along with probabilistic ones that improves performance over the earlier result.

This would appear to make the equiprobability assumption unnecessary (apart from concerns about computational expense). However, we found that, if $p(x,y)$ is set to the empirically estimated joint probability of the lexical link type (x,y), then performance turns out to be dismal. This is understandable: Using parameter estimation techniques like the one we used, a great deal of probability mass in the distribution p tends to go to frequent words, which are relatively uninformative with regard to whether texts are mutual translations. The equiprobability assumption helps to counteract this; in fact one could apply scoring techniques from information retrieval and cross-lingual information retrieval in weighting the lexicon. We leave this area of exploration to future work.

Melamed (2000) used a greedy approximation to MWBM called competitive linking. Competitive linking iteratively selects the edge with the highest weight, links the two vertices of the edge, then removes them from the graph. (Ties are broken at random.) A heap-based implementation of competitive linking runs in $O(\max(|X|,|Y|) \log \max(|X|,|Y|))$. Under the equiprobability assumption, all the weights are the same, so that competitive linking proceeds simply by randomly making legal links (those allowed by the translation lexicon) until no more can be made.

If definition (4) is applied to pairs of documents in the *same* language, with a "translation lexicon" defined by the identity relation, then *tsim* is a variant of resemblance (r), as defined by Broder et al. (1997) for the problem of monolingual duplicate detection, except that *tsim* has the advantage of being token-based rather than type-based, incorporating word frequency.

We have demonstrated that the *tsim* score can be used to extract translationally equivalent English-Chinese sentence pairs from even a noisy space with high precision (Smith 2002). It was also shown that combining multiple sources of word-level translation information (dictionaries, word-to-word translation models, cognates) had positive effects on performance on the sentence-matching task. These information sources were presumed to be extremely noisy (they are so presumed here, as well), though no independent evaluation was carried out on them. If the ad hoc translation lexicon induction techniques used here give good performance, then better techniques might lead to further improvement. In addition, the competitive linking approximation was shown to perform nearly as well as MCBM.

3.2 Experiment

We now apply our content-based similarity measure to the candidate pair classification task presented by STRAND. Recall that both the original STRAND classifier and those learned using decision tree methods, described in Section 2.2.3, employ only structural features of the documents to determine whether they are translations. Here we apply the *tsim* score to the same task and compare the results with those of the original STRAND classifier.

3.2.1 Translation Lexicon. The word-level translation lexicon is derived from several sources. We begin with an English-French dictionary (a total of 34,808 entries, 4,021 of which are not one-to-one).[7] Next, a word-to-word translation model (Melamed 2000) was trained on the dictionary. Note that the parameter estimation task here is very simple; in most cases the pairs are one-word to one-word, making the hidden link structure unambiguous (modulo NULLs). The training primarily served the purpose of breaking down multiword entries, informed by the rest of the entries, so as to obtain a fully one-word-to-one-word dictionary. The training procedure was an expectation-maximization (EM) procedure like that used by Melamed (2000), except that maximum weighted bipartite matching was used instead of competitive linking. Any entry containing a NULL was then removed.

We add to this dictionary a list of English-French cognate pairs, identified using the method of Tiedemann (1999). Tiedemann's approach involved learning language-specific character weights for the computation of weighted edit distance to measure cognateness. He used a list of known cognates to train the weights. We instead used

[7] This dictionary was generated by Gina Levow, who kindly made it available to us. It was derived from data available at ⟨http://www.freedict.com⟩ and contains morphological variants but not character accents.

Table 2
Fifteen randomly chosen cognate pairs. The noise is apparent.

English	French	English	French
closest	choses	TOXLINE	LIABLE
extensions	extension	RELATIONS	ENNEMIS
answer	passer	TARNOS	TARNOS
APPLICATION	PROTECTION	Generation	information
missions	maisons	Commerce	Community
proportion	prestation	private	prêtant
Fischer	Fischer	traditions	attributions
Anglais	dAlcatel		

the weighted translation pairs in a translation model lexicon built from the Bible.[8] The result is 35,513 word pairs from the corpus of Web pages under consideration. An additional set of 11,264 exact string matches were added (also from the corpus of Web pages). Qualitatively, these entries were highly noisy; a random selection of the cognate pairs is shown in Table 2. All of these word pairs were added to the dictionary, each with a count of one.

We took the enhanced dictionary with counts to define a Dirichlet prior, which is the conjugate prior to a multinomial distribution over discrete events (like the distribution p over link types we seek to estimate) (MacKay and Peto 1995). Such a prior is characterized by counts of all such events; when it is used in an EM procedure, these prior counts are added to those produced by the E step on every iteration. Intuitively, if a word pair (x, y) is expected to be a likely lexical word pair in the dictionary and cognate set, then models that make (x, y) probable are more likely (according to the prior). Therefore the expected count of (x, y) is increased at each iteration of training to the extent that the prior favors it.

Using the enhanced, weighted lexicon as a Dirichlet prior (containing 77,699 entries and a total count of 85,332), a word-to-word translation model (Melamed 2000) was trained on a verse-aligned Bible (15,548 verses, averaging 25.5 English words, 23.4 French words after tokenization). As before, we used the maximum weighted bipartite matching algorithm. The final lexicon consists of all word pairs with nonzero probability and contains 132,155 entries. Note that all word pairs in the enhanced dictionary are included; we have merely added to that dictionary by bootstrapping additional entries from the Bible.

3.2.2 Results. In order to compare $tsim$ with structural similarity scoring, we applied it to 325 English-French Web document pairs for which human evaluations were carried out in Section 2. As there is only one feature under consideration ($tsim$), the classifier must be a threshold on that value. At different thresholds, Cohen's κ score of agreement (with each of Resnik's (1999) two judges and their intersection) may be computed for comparison with STRAND, along with recall and precision against a gold standard (for which we use the intersection of the judges: the set of examples

[8] There is some circularity here; the cognates were derived using weighted word pairs from the Bible, then used again in the prior distribution. We note that the resources required to extract cognates in this way are no different from those required for the translation model.

Table 3
Comparison with STRAND. The test set contains 293 of the 326 pairs in Resnik's (1999) test set. The 32 development pairs were used to select manually the 0.44 threshold. N is the number of examples for which judgment comparison was possible in each case (human judges were sometimes undecided; those cases are ignored in computing κ).

Comparison	N	Pr(Agree)	κ	*prec*	*rec*	F
J1, J2	245	0.98	0.96			
J1, STRAND	250	0.88	0.70			
J2, STRAND	284	0.88	0.69			
J1 ∩ J2, STRAND	241	0.90	0.75	**1.000**	0.684	0.812
J1, $tsim(\tau = 0.44)$	249	0.96	0.92			
J2, $tsim(\tau = 0.44)$	283	0.95	0.88			
J1 ∩ J2, $tsim(\tau = 0.44)$	240	0.97	**0.92**	0.833	**0.921**	**0.875**

for which the judges agreed). The gold standard contained 86 page pairs marked as "good" by both judges and 174 page pairs marked as "bad" by both judges.[9]

Computing $tsim$ (MCBM on the words in the document pair) is not tractable for very large documents and translation lexicons. However, in preliminary comparisons, we found that representing $tsim$ for long documents by as few as their first 500 words results in excellent performance on the κ measure.[10] This allows $O(1)$ estimation of $tsim$ for two documents. Further, the competitive linking algorithm appears to be as reliable as MCBM, and it runs significantly faster in practice. The results reported here approximated $tsim$ in this way.

Of the 325 pairs, 32 were randomly selected as a development set, which we used to select manually a threshold $\tau = 0.44$. This value maximized the κ score against gold-standard human judgments on the development set.[11] κ scores against each judge and their intersection were then computed at that threshold on the test set (the remaining 293 pairs). These are compared to κ scores of the STRAND system (with original, untuned parameters), on the same test set, in Table 3. In every case, the $tsim$ classifier agreed more strongly with the human evaluations, and its F score is higher than that of STRAND. Figure 5 shows κ and the F measure plotted against τ.

In this application, the content-based classifier (at its approximate best performance, thresholding at 0.44) complements the structural classifier's high precision. Given two high-performing methods that use orthogonal information for identifying good candidate pairs (one using only structure, the other using only content), the natural question is whether the techniques can be combined for even better performance. We repeated the experiment presented in Section 2.2.3, adding the $tsim$ score as a feature. The same cross-validation setup was used, with the same division into folds. Precision and recall results are reported in Table 4.

[9] One additional pair was thrown out because it contained compressed data; it is assumed that this pair would not pass a language identification filter.

[10] This does, of course, run the risk of failing to identify that a page pair is similar only in the beginning, with diverging content in later text. When the content-based classifier is combined with a structural classifier, this problem is expected to be eliminated, since the structural method detects such cases. See the last experiment described in this section.

[11] One could select such a threshold to maximize any objective function over the development set. We note that this threshold differs from that reported in Smith (2002); it was chosen by the same procedure, though the translation lexicon is different, moving the distribution of $tsim$ scores into a higher range.

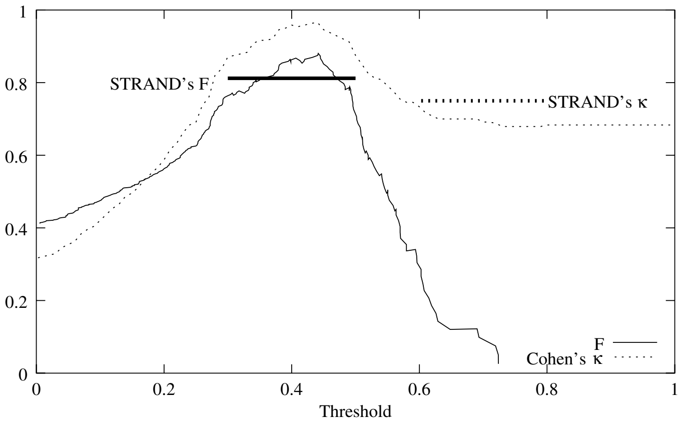


Figure 5
Performance measures as the threshold varies (all measures are on the test set): the κ agreement score with the two judges' intersection and *F* measure. Scores obtained by STRAND are shown as well.

Table 4
Effects of parameter tuning with the additional *tsim* feature.

	Precision	Recall
Untuned STRAND	1.000	0.686
Tuned STRAND (average)	0.958	0.841
tsim($\tau = .44$)	0.833	0.921
Fold 1	0.875	1.000
Fold 2	1.000	1.000
Fold 3	1.000	1.000
Fold 4	1.000	1.000
Fold 5	1.000	1.000
Fold 6	0.889	1.000
Fold 7	1.000	1.000
Fold 8	1.000	1.000
Fold 9	1.000	0.818
Average	0.974	0.980

The decision trees learned were once again all quite similar, with eight of the nine rooted as follows (a few of the trees differed slightly in the numerical values used as thresholds):

```
if tsim > 0.432 then GOOD
else
     if dp > 22.9 then BAD
     else ...
```

The remainder of each tree varied, and there was some evidence of overtraining.

These results clearly demonstrate the benefit of combining structural and content-based approaches. We next describe how we have adapted the STRAND architecture to the Internet Archive, in order to generate the candidate pairs on a scale that has previously been unattainable.

4. Exploiting the Internet Archive

One of the difficulties with doing research on Web mining is that large-scale crawling of the Web is a significant enterprise. Chen and Nie's (2000) PTMiner represents one solution, a carefully thought-out architecture for mining on a large scale. Here we present a different solution, taking advantage of an existing large-scale repository of Web pages maintained on an ongoing basis by an organization known as the Internet Archive.

4.1 The Internet Archive

The Internet Archive ⟨http://www.archive.org/web/researcher/⟩ is a nonprofit organization attempting to archive the entire publicly available Web, preserving the content and providing free access to researchers, historians, scholars, and the general public. Data come from crawls done by Alexa Internet, and hence they represent an industry-level resource of the sort not easily constructed within academia. At present, the Archive contains 120TB (terabytes) of data, by a conservative estimate, and it is growing at approximately 8TB per month. Text on the archive comprises over 10 billion Web pages, and the estimated duplicate rate is a factor of two (i.e., two copies of everything).[12]

The Internet Archive provides public access to the data via the Wayback Machine Web interface. As of this writing, a search for the ACL home page brings up links to 72 snapshots of that page dating back to June 7, 1997.[13] The reader can get to that page directly on the Wayback Machine using a URL that points to the Internet Archive and provides both the desired page and the time stamp indicating which snapshot to retrieve.[14]

The Archive also provides researchers with free, direct access to its data via accounts on their cluster. The data are stored on the disk drives of approximately 300 machines, each running some variety of UNIX, creating what is in essence one huge file system. This provides a researcher with the remarkable sensation of having the entire Web on his or her hard drive.

4.2 Properties of the Archive

Mining terabytes on the Archive presents a number of challenges:

- The Archive is a temporal database, but it is not stored in temporal order. Hence a document and its translation may be in files on different machines; a global merge of data is required for any hope of complete extraction.
- Extracting a document for inspection is an expensive operation involving text decompression.

[12] We are grateful to Paula Keezer of Alexa for these figures.

[13] http://www.cs.columbia.edu/∼acl/home.html

[14] http://web.archive.org/web/19970607032410/http://www.cs.columbia.edu/∼acl/home.html (this is a single, long URL.)

- The Archive's size makes it essential to keep computational complexity low.

On the other hand, as it turns out, aspects of the Archive's architecture make rapid development remarkably feasible:

- Almost all data are stored in compressed plain-text files, rather than in databases.
- The data relevant for our purposes are organized into archive files (arcfiles), which contain the stored pages, and index files, which contain plain text tuples $\langle$URL, time stamp, arcfile, offset, ...$\rangle$.
- A suite of tools exists for processing the archive (e.g., for extracting individual pages from archive files).
- The Archive's infrastructure for cluster computing makes it easy to write UNIX scripts or programs and run them in parallel across machines.

The last of these, the cluster computing tools,[15] is turned out to reduce drastically the time needed to port STRAND to the Archive, as well as the size of the STRAND code base. The centerpiece in Archive cluster computing is a parallelization tool called `p2`, which offers a UNIX command-line interface that allows one to specify (1) a parallelizable task, (2) a way to split it up, (3) a way to combine the results, and (4) a set of processors among which to divide the task. The `p2` tool divides up tasks intelligently, invoking each parallel computation on the local machine where the data reside.

4.3 STRAND on the Archive

In adapting STRAND's three-stage process to the Internet Archive, the primary challenge was in the first two steps, locating possible translations and matching them up to produce candidate document pairs. Structural filtering remained essentially unchanged.

Generating candidate pairs on the Archive involves the following steps:

1. Extracting URLs from index files using simple pattern matching
2. Combining the results from step 1 into a single huge list
3. Grouping URLs into buckets by handles
4. Generating candidate pairs from buckets

Steps 1 and 2 are performed via a parallel search operation plus combination of results; for example, extracting all URLs in the Hong Kong, Taiwan, or China domains (and their associated bookkeeping data) using a pattern like `/(.hk|.tw|.cn)/`.[16]

Step 3 is potentially tricky owing to computational complexity issues. As noted in Section 2.1.2, examining the cross product of a site's page sets in two different languages is potentially very expensive, and matching documents by similarity of URLs can represent a combinatoric process in the general case.

[15] The Archive intends to release these cluster tools under the GNU Public License.

[16] We also take advantage of the Archive's list of `.com` domains paired with the nation in which each is registered, making it possible to include commercial sites in the search without an explosion of irrelevant possibilities.

URLs:

`saudifrenchbank.com.sa/English/English.htm`

→ $\texttt{s}\underline{\texttt{a}}\texttt{udifr}\underline{\texttt{en}}\texttt{chb}\underline{\texttt{a}}\texttt{nk.com.s}\underline{\texttt{a}}\texttt{/}\underline{\texttt{English}}\texttt{/}\underline{\texttt{English}}\texttt{.htm}$

patterns removed:

`a e a a english english`

`saudifrenchbank.com.sa/Arabic/arabic.htm`

→ $\texttt{s}\underline{\texttt{a}}\texttt{udifr}\underline{\texttt{en}}\texttt{chb}\underline{\texttt{a}}\texttt{nk.com.s}\underline{\texttt{a}}\texttt{/}\underline{\texttt{Arabic}}\texttt{/}\underline{\texttt{arabic}}\texttt{.htm}$

patterns removed:

`a e a a arabic arabic`

Same handle for both URLs:

`sudifrchbnk.com.s//.htm`

Figure 6
Example of LSS subtraction.

We arrived at an algorithmically simple solution that avoids this problem but is still based on the idea of language-specific substrings (LSSs). The idea is to identify a set of language-specific URL substrings that pertain to the two languages of interest, (e.g., based on language names, countries, character codeset labels, abbreviations, etc.). For example, a set of LSSs for English-Arabic might be as follows (those containing numerals correspond to character code sets):

> 1256, 437, 864, 8859-1, 8859-6, a, ar, ara, arab, arabic, cp1256, cp437, cp864, e, en, eng, english, gb, iso, iso-8859-1, iso-8859-6, latin, latin-1, latin1, uk, us, usa

For each URL, we form a "handle" by subtracting any substrings that match (insensitive to case) any item on the LSS pattern list. The subtraction process is implemented reasonably efficiently: If there are p patterns with maximum length l, and the URL's length in characters is u, then the current implementation will do at most $p \times u$ string matches of length no more than l.[17] (Currently we use C `strcmp` for string matching.) In practice, this is extremely fast: We can generate handles for nearly 5,000 URLs per second on a six-year-old Sun Ultra 1 workstation.

Figure 6 illustrates handle generation on two real URLs. As one would hope, these two URLs produce the same handle, and as a result, they wind up in the same bucket in step 3.[18]

In step 4, the URLs in each bucket are used to generate candidate pairs by taking the cross product and keeping those URL pairs for which the URL bookkeeping data indicate pages that are in the correct languages. For example, given the bucket containing the two URLs in Figure 6, this step would generate a single pair consisting of

[17] We are grateful to Bill Pugh of the University of Maryland for suggesting this algorithm.

[18] Conceptually, they hash to the same bucket in a hash table; in practice on the Archive, it turns out to be more efficient to create buckets by doing a parallel sort of the entire URL set using the handle as the key, and then creating buckets based on identical handles' being on adjacent lines.

the URL for the English page and the URL for the Arabic page, assuming the language ID information associated with each URL confirmed it was in the proper language.[19]

At this point, the candidate generation process is complete. The final step is to apply STRAND's filtering step to each candidate pair, an operation that can itself be parallelized, since each candidate pair can be processed independently. The filtering pass will eliminate those page pairs (roughly 10% in our experience) whose URLs show similarity to each other but whose content and/or structure do not.

It is interesting to note that by taking advantage of the Archive's p2 cluster computing tool, together with its simple flat-text representations, adapting STRAND's candidate generation process resulted in a dramatic reduction in the size of the program, cutting it literally in half, as measured in lines of code.

5. Building an English-Arabic Corpus

In the previous sections, we have described methods and results for structural matching, for content-based matching, and for dramatically scaling up the number of candidate pairs that can be generated for any given language pair by using the industrial-strength Web crawls stored on the Internet Archive. In this section we put all these pieces together, describing an experiment in mining the Internet Archive to find English-Arabic parallel text. The language pair English-Arabic is of particular global importance, but resources for it, particularly bilingual text, have generally not been easy to obtain. Moreover, Arabic is far behind on the Web's exponential growth curve: Arabic text (as opposed to images) did not really start emerging on the Web until the release of Microsoft Windows 98, which provided Arabic support in its version of Internet Explorer.[20]

5.1 Finding English-Arabic Candidate Pairs on the Internet Archive

The input resources for our search for English-Arabic candidate pairs were a list of Internet domains likely to contain Arabic text.[21] The list included 24 top-level national domains for countries where Arabic is spoken by a significant portion of the population: Egypt (.eg), Saudi Arabia (.sa), Kuwait (.kw), etc. In addition, we used a list of .com domains known to originate in Arabic-speaking countries. This list provided an additional 21 specific domains (e.g., ⟨emiratesbank.com⟩, ⟨checkpoint.com⟩); note that the list is by no means exhaustive.

In the experiments we report here, we mined two crawls from 2001, comprising 8TB and 12TB (i.e., less than one-sixth of the Archive as it existed at the time of the mining effort in early 2002) spread over 27 machines. Our list of URLs with relevant domains, obtained through pattern matching in Archive index files, numbers 19,917,923 pages.[22] The language-specific substrings given earlier were subtracted from these URLs to generate handles, resulting in 786,880 buckets with an average of 25 pages per bucket. When all possible English-Arabic page pairs were generated from all

[19] The Internet Archive tags its data for language using standard *n*-gram language identification techniques.

[20] While this article was under review, a large Arabic-English parallel corpus of United Nations proceedings was released by the Linguistic Data Consortium ⟨http://www.ldc.upenn.edu⟩. Although this corpus is certainly of great import, its availability does not detract from the main point of this study. As noted in Section 1, UN parallel text from LDC is a specialized genre and is encumbered by fees and licensing restrictions. The experiment reported here provides, at minimum, a supplementary resource for English-Arabic, and it provides evidence for the viability of the approach.

[21] We are grateful to Nizar Habash for constructing this list.

[22] Pages with the same URL but different time stamps are counted separately; there were 10,701,622 unique URLs.

Table 5
English-Arabic structural classification results.

		Precision	Recall
Baseline (on the full set)		0.8993	1.0000
Without tuning	Fold 1	1.0000	0.0227
	Fold 2	1.0000	0.1364
	Fold 3	0.6667	0.0435
	Average	0.8889	0.0675
With tuning	Fold 1	0.9111	0.9318
	Fold 2	0.9302	0.9090
	Fold 3	0.9565	0.9565
	Average	0.9326	0.9324

buckets, the result was 8,294 candidate pairs. This number is lower than what might be expected, given the huge number of buckets, because many buckets were monolingual; note that only pairs of one English and one Arabic document are deemed to be candidates.

A random sample of two hundred candidate pairs was given to two human evaluators, bilingual in English and Arabic, who were asked (independently) to answer, for each pair, the question "Is this pair of pages intended to show the same material to two different users, one a reader of English and the other a reader of Arabic?" The judges' answers showed a Cohen's κ agreement of 0.6955, which is generally considered fair to good reliability. (Qualitatively, one judge was rather more strict than the other; when the stricter judge identified a page pair as valid translations, the less strict judge virtually always agreed.)

5.2 Evaluating Structure-Based Matching

Taking the set of 149 labeled pairs on which the two judges agreed (134 were marked "good," 15 "bad"), we carried out an evaluation of the full candidate set similar to the one for English-French discussed in Section 2.2.3. This was a threefold cross-validation experiment in which decision tree classifiers were tuned on the features extracted for each candidate pair by structure-based classification.[23] In addition to the four structural scores, we included two language identification confidence scores (one for the English page, one for the Arabic page); these were available as part of the Internet Archive's bookkeeping information for each URL and required no additional computation on our part. Table 5 shows precision and recall of each fold's classifier applied to the corresponding test set of page pairs. The value of the parameter-tuning process is dramatically confirmed by comparing the learned parameters with STRAND's default parameters (manually determined by Resnik [1999]).

Note, however, that the candidate generation system is highly precise to begin with; only around 10% of the pairs in the random sample of candidates were considered "bad" by both judges. A baseline system in which no filtering is done at all achieves 89.93% precision on the full labeled set (with 100% recall). Depending on the relative importance of precision and recall, these structure-based classifiers might be considered worse than that baseline.

[23] We did not use more than three folds in the cross-validation, since there were only 15 bad pairs and more folds would have made random division into folds that each contained both "good" and "bad" pairs difficult.

Upon inspection, we discovered that nearly 5,000 of the pairs in our candidate set were from a single domain, ⟨maktoob.com⟩. This site supports an online marketplace, and many of the pages discovered by our search were dedicated to specific merchandise categories within that service; a large portion of these were simply "no items available" and one or two similar messages. We ignored this domain completely in order to be conservative about the yield of page pairs, though we note that many of the pages within it are legitimate parallel text that could be extracted if a good duplicates filter were applied.[24]

In order to construct a final classifier, we trained a decision tree on all 149 of the manually judged examples on which both judges agreed. This was then applied to the candidate pairs, producing a set of 1,741 HTML document pairs hypothesized to be valid translations of one another. By way of simple duplicate detection, if a pair of URLs appeared multiple times (under case-insensitive matching), it was counted only once. (Note that when this occurs, the duplicate pair will differ by at least one time stamp, and therefore a more sophisticated technique for eliminating duplication might extract more text.) The remaining set contained 1,399 pairs.[25] Converting from HTML to plain text and tokenizing, the English documents in this corpus total approximately 673,108 tokens, with an average of 481 tokens per document; the Arabic side contains 845,891 tokens, averaging 605 tokens per document.[26]

5.3 Combining Structural and Content-Based Matching

We combined the structural and content-based approaches to detecting translations by adding the $tsim$ score to the set of structural features associated with each candidate pair, and then training a new decision tree classifier. Because Arabic is a highly inflected language with many surface forms, we found it necessary to use morphological preprocessing in order to make effective use of a dictionary. For English, we tokenized the text and used the WordNet lemmatizer to strip suffixes. The Arabic texts were tokenized at punctuation, then romanized and converted to root forms using a morphological analysis tool (Darwish 2002). This approximately halved the vocabulary size for the Arabic texts (from 89,047 types to 48,212 types).

The translation lexicon used to compute *tsim* contained 52,211 entries, each containing one English lemma and one Arabic root.[27] Of these, 16,944 contained two items that were both present in the candidate set of 8,294 Web page pairs. The approximations discussed in Section 3.2.2 were employed: Competitive linking on the first 500 words in each document was used to compute the score.

Carrying out the same cross-validation experiment (on the same random split of data), the combined structural and content-based classifier produced the results in Table 6. Also shown is the performance of the $tsim$-only classifier, assuming an

[24] One of our human evaluators confirmed that no other domains appeared to significantly dominate the candidate pairs as did ⟨maktoob.com⟩, providing some assurance that the rest of the data are diverse.

[25] Within that set, some documents (not pairs) were present multiple times; there were 1,385 unique (apart from case) English URLs and 1,385 unique (apart from case) URLs. Since this is only a 1% duplication rate for each language, we did not attempt to filter further.

[26] We converted HTML to text using the `lynx` browser, performed cleanups such as removing references, and tokenized using the tokenizers included with the Egypt statistical MT package (Al-Onaizan et al. 1999). Those tokenizers are somewhat aggressive about separating out punctuation, so, being aggressive in the opposite direction, we also tried counting only tokens containing at least one of `[A-Za-z]` (which excludes punctuation as well as dates, percentages, etc.). Using that very conservative counting method, the size of the English side is 493,702 words. Counting only tokens on the Arabic side that contained at least one non-numeric, nonpunctuation character yielded 745,480 words.

[27] This translation lexicon was used with the kind permission of Kareem Darwish.

Table 6
English-Arabic combined structural/content-based classification results. The baseline and content-only classifiers are on the full set, and the structure-only classifier is repeated for reference.

	Precision	Recall
Baseline	0.8993	1.0000
Structure only (tuned, average)	0.9326	0.9324
Content only (oracle threshold, $\tau = 0.058$)	0.7688	0.9925
Fold 1	0.9167	1.0000
Fold 2	0.9767	0.9545
Fold 3	0.9583	1.0000
Average	0.9506	0.9848

Table 7
Yield: The English-Arabic Internet Archive corpus, tokenized several ways.

	Tokenization method	English Tokens	Arabic Tokens
Without *tsim*	English lemmas, Arabic roots	620,826	641,654
	Egypt tokenizers (Al-Onaizan et al. 1999)	673,108	845,891
	Egypt tokenizers, words with letters	493,702	745,480
With *tsim*	English lemmas, Arabic roots	960,280	972,392
	Egypt tokenizers (Al-Onaizan et al. 1999)	1,021,839	1,326,803
	Egypt tokenizers, words with letters	800,231	1,213,066

optimal threshold is chosen. Averaged over three folds, the classifier achieved 95.06% precision and 98.48% recall (1.8% and 5.24% better than without *tsim*, respectively).

After building a single classifier on all 149 test pairs (the set on which both human judges agreed), we reclassified the entire candidate set. Ignoring again pages from the ⟨maktoob.com⟩ domain, 2,206 pairs were marked as translations. The same crude duplicate filter was applied, cutting the set back to 1,821 pairs.[28] Table 7 shows word counts for various tokenization schemes: the morphological analysis used for computing *tsim*, the Egypt tokenizer (which is aggressive), and counting only tokens with some alphabetic character from the Egypt tokenizer (a conservative approximation). The analogous results, using the classifier from Section 5.2, are shown for comparison. To summarize the results, using the content-based similarity score as a feature not only improved precision, it increased the size of the corpus (in words) by 51–63%, depending on the tokenization scheme.[29]

6. Future Work

A number of the techniques we have used to mine parallel data from the Web can be improved, and we suggest here some directions.

[28] There were 1,796 unique English URLs and 1,779 unique Arabic URLs, giving document duplication rates of 1.4% and 2.4%, respectively.

[29] A list of Wayback Machine URLs is available at ⟨http://umiacs.umd.edu/∼resnik/strand/⟩; a sample of the document pairs is included in Appendix A.

With respect to classifying document pairs as translations, the reader will notice that our approach to content-based cross-lingual similarity essentially boils down to a greedy matching of some of the words in a document pair using a dictionary. It remains to be seen whether weights in the dictionary can be exploited (Smith [2001] suggests that empirically estimated joint translation probabilities for word pairs are not useful). We suggest that the incorporation of scores from information retrieval (e.g., inverse document frequency) might be useful in discerning which lexicon entries are the strongest cues of translational equivalence. We also have not explored any filtering on the noisy translation lexicon; doing so might improve the quality of the $tsim$ score.

The competitive linking approximation (which, without weights, is essentially random matching of word pairs) and the use of only the initial portion of each document provide significant computational savings. In our experience, neither of these has significantly hurt the performance of $tsim$-based classifiers (as compared to finding the maximum cardinality bipartite matching and/or using the full documents), and in some cases competitive linking seems to improve performance. It is possible that some sample selection of words from document candidates might be profitable (e.g., creating a sample of size proportional to the document length, as opposed to a fixed size, or sampling only from content words, or sampling only words present in the dictionary).

Smith (2002) suggested a bootstrapping paradigm for the construction of parallel corpora. Beginning with a seed set of translation information (either parallel corpora or a bilingual dictionary), high-precision initial classifiers might be constructed using content and/or structural features (whichever are available). We might then iteratively select additional page pairs in which the current classifier has high confidence of translational equivalence, gradually increasing the pool of parallel data and at the same time expanding the bilingual lexicon. This approach to minimally supervised classifier construction has been widely studied (Yarowsky 1995), especially in cases in which the features of interest are orthogonal in some sense (e.g., Blum and Mitchell 1998; Abney 2002).

With respect to the generation of candidate pairs, we have described a progression from index-based searches on AltaVista to exhaustive matching of URLs on the Internet Archive. The combination of these approaches may be profitable, particularly for languages that are represented only very sparsely on the Web. For such languages, index-based searches on words from a language of interest might be used to identify sites potentially containing parallel text. Within such sites, it would likely be profitable to look for parallel documents in the full cross product of documents in the two languages of interest, obtained both on the Internet Archive and via crawling all pages on relevant sites.

Finally, we plan to utilize parallel texts mined from the Web in our work on machine translation and acquisition of bilingual lexicons, and in the creation of resources for new languages via projection of annotations from English.

7. Conclusions

Although efforts at discovering parallel text on the Web were first reported in 1998, Web-based parallel corpora appear to have had only a limited impact on the community. Three reasons for this suggest themselves.

Too few languages. Parallel text from the Web has been made available to the community in only a few pairs of languages. As of this writing, the STRAND Web site ⟨http://umiacs.umd.edu/∼resnik/strand/⟩, presenting URL pairs discovered via STRAND runs, contains collections only for English-French, English-Chinese, English-

Basque, and now English-Arabic, and we are not aware of any other efforts to disseminate Web-based parallel data publicly. Up to this point, it simply has not been easy to search the Web for parallel text in new language pairs. The most difficult part is finding the candidates: A year or two ago, we attempted to apply the original Web-based STRAND to the problem of finding English-Arabic text, and we were unable to locate enough search engine hits or sites to yield useful results.

Too little data. Very large Web-based parallel text collections are not available to the community. The largest appear to have been obtained by Chen and Nie (2000), who acquired collections on the order of 15,000 document pairs for English-French, English-German, English-Dutch, English-Italian, and English-Chinese using the PT-Miner system. However, these collections have not been made available to the general community.[30] In contrast, the STRAND collections, which are available to the community in the form of URL pairs, are modest in size: The English-Chinese collection contains fewer than 3,500 document pairs, and the English-French fewer than 2,500.

Difficulty with dissemination. Web-based collections are difficult to distribute. Standard mechanisms of the sort used by the LDC (a CD or downloadable file) are fraught with difficult legal issues, since, technically speaking, redistributing the actual content of Web pages could require permission from the author of every page. For example, presumably as a risk reduction strategy, the Web track for TREC-2002 (Text Retrieval Conference) limited its attention to the .gov domain and required the recipient of the data to sign a form that reduced the distributor's liability.[31] Similarly, the Google Programming Contest data set arrived with a limited-use license, indemnification from third-party claims, and a collection limited to the .edu domain, from which, presumably, authors are less likely to bring expensive lawsuits ⟨http://www.google.com/programming-contest/⟩.

A possible fourth reason may have to do with questions about the utility of the data. For example, a Web-based parallel collection may be unpredictable in terms of its coverage, and the community is well aware of the dangers of using training data that are not representative of the test domain. A solution to this problem might be to extract topically relevant subsets of the collection for particular domains or applications, but of course this requires a "more is better" approach in order to obtain subsets that are large enough to be useful.

The work reported in this article addresses each of these major problems. With respect to the number of language pairs, the Internet Archive offers us a huge sample of pages on the Web, and our techniques make it easy to explore that collection in an efficient way. Although it is probably impossible to crawl more than a small fraction of the Web, the Internet Archive is storing the results of commercial-scale Web crawling and has as its explicit mission the permanent storage of everything that can be found. The fact that we were able to find a substantial quantity of English-Arabic text (on the order of a million words per side, looking at less than a sixth of the Archive in 2002) offers the hope that it will be possible to find data for the less well-represented language pairs, if and when those data actually exist. Moreover, the final implementation we described here retains the almost entirely language-independent character of the original STRAND system, adding only the requirement of a reasonable transla-

[30] These data were made available only to participants in the iCLEF evaluations for interactive cross-language IR (C. Picci, personal communication).

[31] "The limitation on permitted use . . . is intended to reduce the risk of any action being brought by copyright owners, but if this happens the Organisation [recipient] agrees to bear all associated liability" ⟨http://www.ted.cmis.csiro.au/TRECWeb/access_to_data.html⟩.

tion lexicon. Therefore success in mining for parallel text in other languages depends primarily on whether the data exist in the Archive.

With regard to corpus size, we demonstrated that the recall of structural matching, and hence its yield, can be significantly improved by simple and automatic classifier construction, requiring only a few hours' work from a bilingual annotator to create the training material. These results are further improved by adding content-based similarity as a feature. Our success with English-Arabic, a language pair that is not one of those usually considered well represented on the Web, encourages us to believe that for other languages of interest, we will be similarly successful. We have also done a bit of exploration to gauge the potential of the Archive for better-represented language pairs, using English-Chinese as an example. By way of context, Chen and Nie (2000) reported that PTMiner found around 15,000 English-Chinese document pairs by crawling 185 sites in the .hk (Hong Kong) domain, with the run taking about a week. We did a STRAND search of the two Internet Archive crawls used in the English-Arabic study, seeking English-Chinese parallel text in multiple domains where Chinese is a dominant language (e.g., .hk, .tw, .cn). Our initial candidate pair set was generated in approximately 30 hours, and contains over 70,000 candidate page pairs. We are optimistic that this can be improved still further by expanding the search to include all sites that contain at least one Chinese document, regardless of the domain.

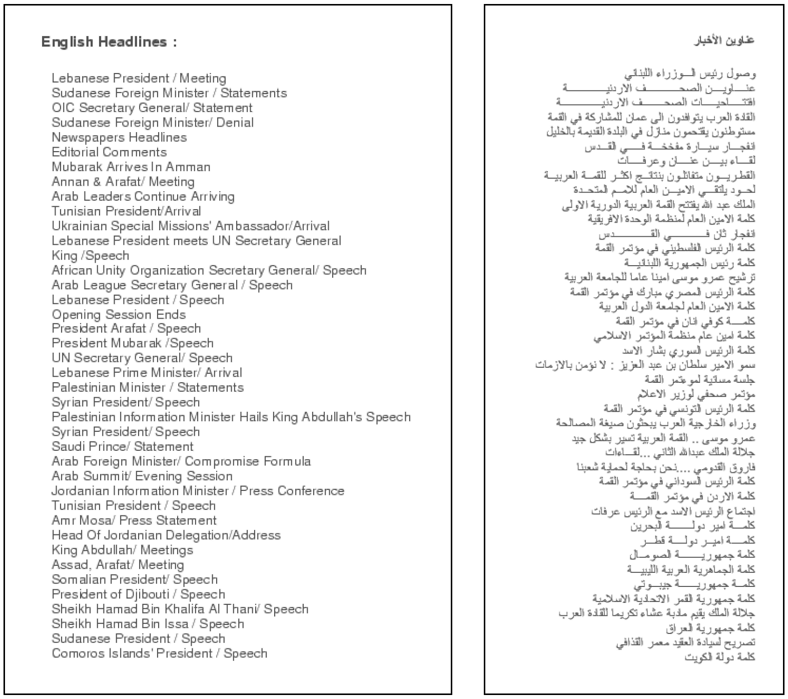
In terms of dissemination, the STRAND distribution mechanism models itself after Web search engines, distributing the URLs rather than the pages themselves. This places the legal burden on individual users, who are presumably safe under fair use provisions if they download pages for their individual use. Until recently the difficulty with this solution has been that the collection of URLs deteriorates over time as sites disappear, pages are reorganized, and underlying content changes: For example, in April 2002, we attempted to download the documents in the STRAND English-French, English-Chinese, and English-Basque collections, and we were able to access successfully only around 67%, 43%, and 40% of the URL pairs, respectively. However, the Internet Archive's Wayback Machine provides a way to distribute *persistent* URLs.

With regard to the quality of the data, in Section 2.2.2 we discussed two studies that demonstrate the utility of parallel Web data in acquiring translation lexicons for cross-language information retrieval. We also reported on the results of a human ratings study, which provided evidence that English-Chinese data mined from the Web contain reasonably fluent, reasonably translated sentence pairs. It is worth pointing out that, because STRAND expects pages to be very similar in structural terms, the resulting document collections are particularly amenable to sentence- or segment-level alignment. Indeed, just using dynamic programming to align the markup, ignoring the text, produces reasonable first-pass alignments of the intervening text as a side effect. We are currently adapting statistical text-based sentence alignment techniques to take advantage of the markup available in Web-based document pairs.

Ultimately, the utility of parallel data from the Web is a question that will need to be addressed in practice. The potential, of course, is as rich and diverse as the Web itself, and what we as researchers can do with it is an exciting question that remains to be answered.

Appendix A: Examples

The following page pairs (Figures 7–8) are representative of English-Arabic parallel corpus extracted from the Internet Archive. The text from the pages is shown in full. Note that the full corpus is available as a list of Wayback Machine URLs at ⟨http://umiacs.umd.edu/∼resnik/strand⟩. These pages show the generally high qual-



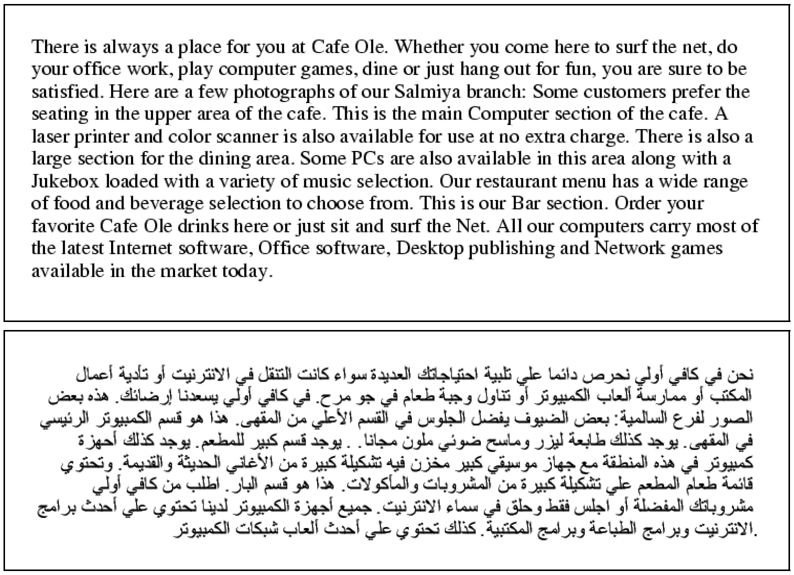
English Headlines :

Lebanese President / Meeting
Sudanese Foreign Minister / Statements
OIC Secretary General/ Statement
Sudanese Foreign Minister/ Denial
Newspapers Headlines
Editorial Comments
Mubarak Arrives In Amman
Annan & Arafat/ Meeting
Arab Leaders Continue Arriving
Tunisian President/Arrival
Ukrainian Special Missions' Ambassador/Arrival
Lebanese President meets UN Secretary General
King /Speech
African Unity Organization Secretary General/ Speech
Arab League Secretary General / Speech
Lebanese President / Speech
Opening Session Ends
President Arafat / Speech
President Mubarak /Speech
UN Secretary General/ Speech
Lebanese Prime Minister/ Arrival
Palestinian Minister / Statements
Syrian President/ Speech
Palestinian Information Minister Hails King Abdullah's Speech
Syrian President/ Speech
Saudi Prince/ Statement
Arab Foreign Minister/ Compromise Formula
Arab Summit/ Evening Session
Jordanian Information Minister / Press Conference
Tunisian President / Speech
Amr Mosa/ Press Statement
Head Of Jordanian Delegation/Address
King Abdullah/ Meetings
Assad, Arafat/ Meeting
Somalian President/ Speech
President of Djibouti / Speech
Sheikh Hamad Bin Khalifa Al Thani/ Speech
Sheikh Hamad Bin Issa / Speech
Sudanese President / Speech
Comoros Islands' President / Speech

عناوين الأخبار

وصول رئيس الـــوزراء اللبناني
عنـــاويـــن الصحـــــــف الاردنيـــــــــة
افتتـــاحيـــات الصحـــــــف الاردنيـــــــــة
القادة العرب يتوافدون الى عمان للمشاركة في القمة
مستوطنون يقتحمون منازل في البلدة القديمة بالخليل
انفجـــار سيـــارة مفخخـــة فـــــي القـــدس
لقـــاء بيـــن عنـــان وعرفـــات
القطريـــون متفائلون بنتائـــج اكثـــر للقمـــة العربيـــة
لحـــود يلتقـــي الاميـــن العام للامـــم المتحـــدة
الملك عبد الله يفتتح القمة العربية الدورية الاولى
كلمة الامين العام لمنظمة الوحدة الافريقية
انفجار ثان فـــــــي القـــــــــدس
كلمة الرئيس الفلسطيني في مؤتمر القمة
كلمة رئيس الجمهورية اللبنانيـــة
ترشيح عمرو موسى امينا عاما للجامعة العربية
كلمة الرئيس المصري مبارك في مؤتمر القمة
كلمة الامين العام لجامعة الدول العربية
كلمـــــة كوفي انان في مؤتمر القمة
كلمة امين عام منظمة المؤتمر الاسلامي
كلمة الرئيس السوري بشار الاسد
سمو الامير سلطان بن عبد العزيز : لا نؤمن بالازمات
جلسة مسائية لموعتمر القمة
مؤتمر صحفي لوزير الاعلام
كلمة الرئيس التونسي في مؤتمر القمة
وزراء الخارجية العرب يبحثون صيغة المصالحة
عمرو موسى .. القمة العربية تسير بشكل جيد
جلالة الملك عبدالله الثاني ...لقـــاءات
فاروق القدومي ....نحن بحاجة لحماية شعبنا
كلمة الرئيس السوداني في مؤتمر القمة
كلمة الاردن في مؤتمر القمـــة
اجتماع الرئيس الاسد مع الرئيس عرفات
كلمـــة امير دولـــــة البحرين
كلمـــة اميـــر دولـــة قطـــر
كلمة جمهوريـــــــة الصومـــال
كلمة الجماهرية العربية الليبيـــة
كلمـــة جمهوريـــــــة جيبـــوتي
كلمة جمهورية القمر الاتحادية الاسلامية
جلالة الملك يقيم مادبة عشاء تكريما للقادة العرب
كلمة جمهورية العراق
تصريح لسيادة العقيد معمر القذافي
كلمة دولة الكويت

Figure 7
Representative English-Arabic page pair: ⟨petranews.gov.jo:80/2001/86/tocen.htm⟩ (8 April 2001) and ⟨petranews.gov.jo:80/2001/86/tocar.htm⟩ (6 April 2001).



There is always a place for you at Cafe Ole. Whether you come here to surf the net, do your office work, play computer games, dine or just hang out for fun, you are sure to be satisfied. Here are a few photographs of our Salmiya branch: Some customers prefer the seating in the upper area of the cafe. This is the main Computer section of the cafe. A laser printer and color scanner is also available for use at no extra charge. There is also a large section for the dining area. Some PCs are also available in this area along with a Jukebox loaded with a variety of music selection. Our restaurant menu has a wide range of food and beverage selection to choose from. This is our Bar section. Order your favorite Cafe Ole drinks here or just sit and surf the Net. All our computers carry most of the latest Internet software, Office software, Desktop publishing and Network games available in the market today.

نحن في كافي أولي نحرص دائما علي تلبية احتياجاتك العديدة سواء كانت التنقل في الانترنيت أو تأدية أعمال
المكتب أو ممارسة ألعاب الكمبيوتر أو تناول وجبة طعام في جو مرح. في كافي أولي يسعدنا إرضائك. هذه بعض
الصور لفرع السالمية: بعض الضيوف يفضل الجلوس في القسم الأعلي من المقهى. هذا هو قسم الكمبيوتر الرئيسي
في المقهى. يوجد كذلك طابعة ليزر وماسح ضوئي ملون مجانا. . يوجد قسم كبير للمطعم. يوجد كذلك أجهزة
كمبيوتر في هذه المنطقة مع جهاز موسيقي كبير مخزن فيه تشكيلة كبيرة من الأغاني الحديثة والقديمة. وتحتوي
قائمة طعام المطعم علي تشكيلة كبيرة من المشروبات والمأكولات. هذا هو قسم البار. اطلب من كافي أولي
مشروباتك المفضلة أو اجلس فقط وحلق في سماء الانترنيت. جميع أجهزة الكمبيوتر لدينا تحتوي علي أحدث برامج
الانترنيت وبرامج الطباعة وبرامج المكتبية. كذلك تحتوي علي أحدث ألعاب شبكات الكمبيوتر.

Figure 8
Representative English-Arabic page pair: ⟨www.ole.com.kw:80/pictures.htm⟩ (9 April 2001) and ⟨www.ole.com.kw:80/picturesa.htm⟩ (17 April 2001).

ity of the corpus and also illustrate some of the potential difficulties with parallel Web data. For example, the Arabic page in the first pair includes an additional caption not present in the English side. These kinds of problems are expected to be overcome during sentence alignment processing.

Appendix B: Translation Ratings Criteria

For each item, participants were instructed to provide three ratings.
Quality of the English:

3. Very fluent: All of the English is comprehensible.
2. Fairly fluent: The major part of the English passes, but there are noticeable errors.
1. Barely fluent: Only part of the English meaning is understandable.
0. Unintelligible: Nothing or almost nothing of the English is comprehensible.

Quality of the Chinese:

3. Very fluent: All of the Chinese is comprehensible.
2. Fairly fluent: The major part of the Chinese passes, but there are noticeable errors.
1. Barely fluent: Only part of the Chinese meaning is understandable.
0. Unintelligible: Nothing or almost nothing of the Chinese is comprehensible.

Adequacy of translation:

4. The English and Chinese contain entirely the same meaning.
3. The English and Chinese contain mostly the same meaning.
2. The English and Chinese contain much of the same meaning.
1. The English and Chinese contain little of the same meaning.
0. The English and Chinese contain none of the same meaning.

Acknowledgments
This work has been supported in part by Department of Defense Contract RD-02-5700, DARPA/ITO Cooperative Agreement N660010028910, ONR MURI Contract FCPO.810548265, and National Science Foundation Grant EIA0130422. The second author is supported by a Fannie and John Hertz Foundation Fellowship. We would like to thank Nizar Habash, Kareem Darwish, Mona Diab, Usama Soltan, and David Smith for their assistance with Arabic, and Jason Eisner, Rebecca Hwa, Mark Liberman, Dan Melamed, Michael Nossal, and Doug Oard for helpful conversations. We are grateful to Yevgeniy Lyudovyk for allowing us to use his language identification software. We would also like to thank Sang Hong, Greg Marton, and Michael Subotin for assistance with implementation, and Brewster Kahle, Andy Jewell, Jad DeFanti, Brad Tofel, and Paula

Keezer for permitting and facilitating our use of the Internet Archive. Finally, we are indebted to several *Computational Linguistics* reviewers, whose comments helped us to greatly improve this article.

References

Abney, Steven. 2002. Bootstrapping. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, pages 360–367.

Ahuja, Ravindra K., Thomas L. Magnati, and James B. Orlin. 1993. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall, Englewood Cliffs, NJ.

Al-Onaizan, Yaser, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, I. Dan Melamed, Franz-Josef Och, David Purdy, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation. Technical report, Johns Hopkins University. Available at ⟨citeseer.nj.nec.com/al-onaizan99statistical.html⟩.

Beesley, Kenneth R. 1988. Language identifier: A computer program for automatic natural-language identification of on-line text. In D. L. Hammond, editor, *Language at the Crossroads: Proceedings of the 29th Annual Conference of the American Translators Association*, Medford, NJ, pages 47–54.

Blum, Avrim and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of the Eleventh Annual Conference on Computational Learning Theory*, pages 92–100, Madison, WI, July.

Broder, Andrei Z., Steven C. Glassman, Mark S. Manasse, and Geoffrey Zweig. 1997. Syntactic clustering of the Web. In *Proceedings of the Sixth International World-Wide Web Conference*, pages 391–404, Santa Clara, CA, April.

Brown, Peter F., John Cocke, Stephen A. Della Pietra, Vincent J. Della Pietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85.

Cabezas, Clara, Bonnie Dorr, and Philip Resnik. 2001. Spanish language processing at University of Maryland: Building infrastructure for multilingual applications. In *Proceedings of the Second International Workshop on Spanish Language Processing and Language Technologies (SLPLT-2)*. Available at ⟨ftp://ftp.umiacs.umd.edu/pub/bonnie/slplt-01.htm⟩.

Cavnar, William B. and John M. Trenkle. 1994. N-gram-based text categorization. In *Proceedings of Third Annual Symposium on Document Analysis and Information Retrieval (SDAIR-94)*, pages 161–175, Las Vegas, NV.

Chen, Jiang and Jian-Yun Nie. 2000. Parallel Web text mining for cross-language information retrieval. In *Recherche d'Informations Assistée par Ordinateur (RIAO)*, pages 62–77, Paris, April.

Church, Kenneth W. and Robert Mercer. 1993. Introduction to the special issue on computational linguistics using large corpora. *Computational Linguistics*, 19(1):1–24.

Dabbadie, Marianne, Anthony Hartley, Margaret King, Keith J. Miller, Widad Mustafa El Hadi, Andrei Popescu-Bellis, Florence Reeder, and Michelle Vanni. 2002. A hands-on study of the reliability and coherence of evaluation metrics. In *Workshop on Machine Translation Evaluation: Human Evaluators Meet Automated Metrics at the Third International Conference on Language Resources and Evaluation (LREC-2000)*, pages 8–16, Las Palmas, Canary Islands, Spain, May.

Darwish, Kareem. 2002. Building a shallow Arabic morphological analyser in one day. In *Proceedings of the Workshop on Computational Approaches to Semitic Languages*, pages 22–29, Philadelphia, July.

Davis, Mark and Ted Dunning. 1995. A TREC evaluation of query translation methods for multi-lingual text retrieval. In *Fourth Text Retrieval Conference (TREC-4)*, pages 483–498. NIST, Gaithersburg, MD.

Diab, Mona and Philip Resnik. 2002. An unsupervised method for word sense tagging using parallel corpora. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, Philadelphia, July.

Dunning, Ted. 1994. Statistical identification of language. Computing Research Laboratory Technical Memo MCCS 94-273, New Mexico State University, Las Cruces, NM. Available at ⟨http://citeseer.nj.nec.com/dunning94statistical.html⟩.

Gale, William A. and Kenneth W. Church. 1991. Identifying word correspondences in parallel texts. In *Fourth DARPA Workshop on Speech and Natural Language*, pages 152–157, Asilomar, CA, February.

Hunt, James W. and M. Douglas McIlroy. 1975. An algorithm for differential file comparison. Technical Memorandum 75-1271-11, Bell Laboratories, October.

Hwa, Rebecca, Philip Resnik, Amy Weinberg, and Okan Kolak. 2002. Evaluating translational correspondence using annotation projection. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 392–399, Philadelphia, July.

Ingle, Norman C. 1976. A language identification table. *The Incorporated Linguist*, 15(4):98–101.

Landauer, Thomas K. and Michael L. Littman. 1990. Fully automatic cross-language document retrieval using latent semantic indexing. In *Proceedings of the Sixth Annual Conference of the UW Centre for the New Oxford English Dictionary and Text Research*, pages 31–38, Waterloo, Ontario, October.

Lin, Dekang. 1998. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML-98)*, pages 296–304, Madison, WI.

Lopez, Adam, Michael Nossal, Rebecca Hwa, and Philip Resnik. 2002. Word-level alignment for multilingual resource acquisition. In *Workshop on Linguistic Knowledge Acquisition and Representation: Bootstrapping Annotated Language Data, at the Third International Conference on Language Resources and Evaluation (LREC-2000)*, Las Palmas, Canary Islands, Spain, June. Available at ⟨http://www.umiacs.umd.edu/∼hwa/lnhr02.ps⟩.

Ma, Xiaoyi and Mark Liberman. 1999. Bits: A method for bilingual text search over the Web. In *Machine Translation Summit VII*, September. Available at ⟨http://www.ldc.upenn.edu/Papers/MTSVII1999/BITS.ps⟩.

MacKay, David and Linda Peto. 1995. A hierarchical Dirichlet language model. *Journal of Natural Language Engineering,* 1(3):1–19.

Melamed, I. Dan. 1997. Automatic discovery of non-compositional compounds in parallel data. In *Proceedings of the 2nd Conference on Empirical Methods in Natural Language Processing (EMNLP-97)*, pages 97–108, Providence, RI, August.

Melamed, I. Dan. 2000. Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249.

Menezes, Arul and Stephen D. Richardson. 2001. A best-first alignment algorithm for automatic extraction of transfer mappings from bilingual corpora. In *Proceedings of the ACL 2001 Workshop on Data-Driven Methods in Machine Translation*, pages 39–46, Toulouse, France.

Nie, Juan-Yun and Jian Cai. 2001. Filtering noisy parallel corpora of Web pages. In *IEEE Symposium on Natural Language Processing and Knowledge Engineering*, pages 453–458, Tucson, AZ, October.

Oard, Douglas W. 1997. Cross-language text retrieval research in the USA. In *Third DELOS Workshop on Cross-Language Information Retrieval*, Zurich, March. European Research Consortium for Informatics and Mathematics, Sophia Antipolis, France. Available at ⟨http://www.ercim.org/publication/ws-proceedings/DELOS3/Oard.ps.gz⟩.

Och, Franz-Josef and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 295–302, Philadelphia, July.

Resnik, Philip. 1998. Parallel strands: A preliminary investigation into mining the Web for bilingual text. In *Proceedings of the Third Conference of the Association for Machine Translation in the Americas (AMTA-98).* Langhorne, PA, October 28-31. Lecture Notes in Artificial Intelligence 1529. Available at ⟨http://umiacs.umd.edu/∼resnik/pubs/amta98.ps.gz⟩.

Resnik, Philip. 1999. Mining the Web for bilingual text. In *Proceedings of the 37th Annual Meeting of the ACL*, pages 527–534, College Park, MD, June.

Resnik, Philip and I. Dan Melamed. 1997. Semi-automatic acquisition of domain-specific translation lexicons. In *Fifth Conference on Applied Natural Language Processing*, Washington, DC.

Resnik, Philip, Douglas Oard, and Gina Levow. 2001. Improved cross-language retrieval using backoff translation. In *Proceedings of the First International Conference on Human Language Technology Research (HLT-2001)*, pages 153–155, San Diego, CA, March.

Resnik, Philip, Mari Broman Olsen, and Mona Diab. 1999. The Bible as a parallel corpus: Annotating the "Book of 2000 Tongues." *Computers and the Humanities*, 33:129–153.

Riloff, Ellen, Charles Schafer, and David Yarowsky. 2002. Inducing information extraction systems for new languages via cross-language projection. In *Nineteenth International Conference on Computational Linguistics (COLING-2002)*, pages 828–834, Taipei, Taiwan, August.

Shannon, Claude E. 1948. A mathematical theory of communication. *Bell Systems Technical Journal*, 27:379–423, 623–656.

Smith, Noah A. 2001. Detection of translational equivalence. Undergraduate honors thesis, University of Maryland College Park. Available at ⟨http://nlp.cs.jhu.edu/∼nasmith/cmsc-thesis.ps⟩.

Smith, Noah A. 2002. From words to corpora: Recognizing translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 95–102, Philadelphia, July.

Tiedemann, Jörg. 1999. Automatic construction of weighted string similarity measures. In *Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 213–219, College Park, MD, June.

Yarowsky, David. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 189–196, Cambridge, MA.

Yarowsky, David and Grace Ngai. 2001. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics (NAACL-2001)*, pages 200–207, Pittsburgh, PA, June.

Yarowsky, David, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the First International Conference on Human Language Technology Research (HLT-2001)*, pages 161–168, San Diego, CA, March.