# Determining Recurrent Sound Correspondences by Inducing Translation Models

**Grzegorz Kondrak**
Department of Computer Science
University of Toronto
Toronto, Ontario, Canada M5S 3G4

## Abstract

I present a novel approach to the determination of recurrent sound correspondences in bilingual wordlists. The idea is to relate correspondences between sounds in wordlists to translational equivalences between words in bitexts (bilingual corpora). My method induces models of sound correspondence that are similar to models developed for statistical machine translation. The experiments show that the method is able to determine recurrent sound correspondences in bilingual wordlists in which less than 30% of the pairs are cognates. By employing the discovered correspondences, the method can identify cognates with higher accuracy than the previously reported algorithms.

## 1 Introduction

Genetically related languages often exhibit recurrent sound correspondences (henceforth referred to simply as correspondences) in words with similar meaning. For example, *t:d*, *θ:t*, *n:n*, and other known correspondences between English and Latin are demonstrated by the word pairs in Table 1. Word pairs that contain such correspondences are called *cognates*, because they originate from the same protoform in the ancestor language. Correspondences in cognates are preserved over time thanks to the regularity of sound changes, which normally apply to sounds in a given phonological context across all words in the language.

The determination of correspondences is the principal step of the comparative method of language reconstruction. Not only does it provide evidence for the relatedness of languages, but it also makes it possible to distinguish cognates from loan words and chance resemblances. However, because manual determination of correspondences is an extremely time-consuming process, it has yet to be accomplished for many proposed language families. A system able to perform this task automatically from unprocessed bilingual wordlists could be of great assistance to historical linguists. The *Reconstruction Engine* (Lowe and Mazaudon, 1994), a set of programs designed to be an aid in language reconstruction, requires a set of correspondences to be provided beforehand.

| English | Latin | |
|---|---|---|
| **t** ɛ n | **d** e k e | ‘ten’ |
| **t** ū | **d** u o | ‘two’ |
| ī **t** | e **d** | ‘eat’ |
| **t** ū **θ** | **d** e n **t** | ‘tooth’ |
| **n** ɛ s **t** | **n** i **d** | ‘nest’ |
| **n** ī | g e **n** | ‘knee’ |
| **n** ɛ **f** j ū | **n** e **p** o t | ‘nephew’ |
| **f** u **t** | **p** e **d** | ‘foot’ |
| **f** ō **m** | s **p** u **m** | ‘foam’ |
| w u **l f** | **l** u **p** | ‘wolf’ |

Table 1: Examples of English–Latin cognates exhibiting correspondences. The corresponding phonemes shown in boldface originate from a single proto-phoneme.

The determination of correspondences is closely related to another task that has been much studied in computational linguistics, the identification of cognates. Cognates have been employed for sentence and word alignment in bitexts (Simard et al., 1992; Melamed, 1999), improving statistical machine translation models (Al-Onaizan et al., 1999), and inducing translation lexicons (Koehn and Knight, 2001). Some of the proposed cognate identification algorithms implicitly determine and employ correspondences (Tiedemann, 1999; Mann and Yarowsky, 2001).

Although it may not be immediately apparent, there is a strong similarity between the task of matching phonetic segments in a pair of cognate words, and the task of matching words in two sentences that are mutual translations (Figure 1). The

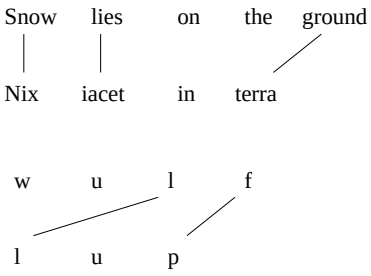


Figure 1: The similarity of word alignment in bitexts and phoneme alignment between cognates.

consistency with which a word in one language is translated into a word in another language is mirrored by the consistency of sound correspondences. The former is due to the semantic relation of synonymy, while the latter follows from the principle of the regularity of sound change. Thus, as already asserted by Guy (1994), it should be possible to use similar techniques for both tasks.

The primary objective of the method proposed in this paper is the automatic determination of correspondences in bilingual wordlists, such as the one in Table 1. The method exploits the idea of relating correspondences in bilingual wordlists to translational equivalence associations in bitexts through the employment of models developed in the context of statistical machine translation, The second task addressed in this paper is the identification of cognates on the basis of the discovered correspondences. The experiments to be described in Section 6 show that the method is capable of determining correspondences in bilingual wordlists in which less than 30% of the pairs are cognates, and outperforms comparable algorithms on cognate identification. Although the experiments focus on bilingual wordlists, the approach presented in this paper could potentially be applied to other bitext-related tasks.

## 2 Related work

In a schematic description of the comparative method, the two steps that precede the determination of correspondences are the identification of cognate pairs (Kondrak, 2001), and their phonetic alignment (Kondrak, 2000). Indeed, if a comprehensive set of correctly aligned cognate pairs is available, the correspondences could be extracted by simply following the alignment links. Unfortunately, in order to make reliable judgments of cognation, it is necessary to know in advance what the correspondences are. Historical linguists solve this apparent circularity by guessing a small number of likely cognates and refining the set of correspondences and cognates in an iterative fashion.

Guy (1994) outlines an algorithm for identifying cognates in bilingual wordlists which is based on correspondences. The algorithm estimates the probability of phoneme correspondences by employing a variant of the $\chi^2$ statistic on a contingency table, which indicates how often two phonemes co-occur in words of the same meaning. The probabilities are then converted into the estimates of cognation by means of some experimentation-based heuristics. The paper does not contain any evaluation on authentic language data, but Guy's program COGNATE, which implements the algorithm, is publicly available. An experimental evaluation of COGNATE is described in Section 6.

Oakes (2000) describes a set of programs that together perform several steps of the comparative method, from the determination of correspondences in wordlists to the actual reconstruction of the protoforms. Word pairs are considered cognate if their edit distance is below a certain threshold. The edit operations cover a number of sound-change categories. Sound correspondences are deemed to be regular if they are found to occur more than once in the data. The paper describes experimental results of running the programs on a set of wordlists representing four Indonesian languages, and compares those to the reconstructions found in the linguistic literature. Section 6 contains an evaluation of one of the programs in the set, JAKARTA, on the cognate identification task.

## 3 Models of translational equivalence

In statistical machine translation, a translation model approximates the probability that two sentences are mutual translations by computing the product of the probabilities that each word in the target sentence is a translation of some source language word. A model of translation equivalence that determines the word translation probabilities can be *induced* from bitexts. The difficulty lies in the fact that the mapping, or alignment, of words between two parts of a bitext is not known in advance.

Algorithms for word alignment in bitexts aim at discovering word pairs that are mutual translations. A straightforward approach is to estimate the likelihood that words are mutual translations by computing a similarity function based on a co-occurrence

statistic, such as mutual information, Dice coefficient, or the $\chi^2$ test. The underlying assumption is that the association scores for different word pairs are independent of each other.

Melamed (2000) shows that the assumption of independence leads to invalid word associations, and proposes an algorithm for inducing models of translational equivalence that outperform the models that are based solely on co-occurrence counts. His models employ the *one-to-one* assumption, which formalizes the observation that most words in bitexts are translated to a single word in the corresponding sentence. The algorithm, which is related to the expectation-maximization (EM) algorithm, iteratively re-estimates the *likelihood scores* which represent the probability that two word types are mutual translations. In the first step, the scores are initialized according to the $G^2$ statistic (Dunning, 1993). Next, the likelihood scores are used to induce a set of one-to-one *links* between word tokens in the bitext. The links are determined by a greedy *competitive linking* algorithm, which proceeds to link pairs that have the highest likelihood scores. After the linking is completed, the link counts are used to re-estimate the likelihood scores, which in turn are applied to find a new set of links. The process is repeated until the translation model converges to the desired degree.

Melamed presents three translation-model estimation methods. Method A re-estimates the likelihood scores as the logarithm of the probability of jointly generating the pair of words $u$ and $v$:

$$score_A(u,v) = \log \frac{links(u,v)}{\sum_{u',v'} links(u',v')}$$

where $links(u,v)$ denotes the number of links induced between $u$ and $v$. Note that the co-occurrence counts of $u$ and $v$ are not used for the re-estimation,

In Method B, an explicit noise model with auxiliary parameters $\lambda^+$ and $\lambda^-$ is constructed in order to improve the estimation of likelihood scores. $\lambda^+$ is a probability that a link is induced between two co-occurring words that are mutual translations, while $\lambda^-$ is a probability that a link is induced between two co-occurring words that are not mutual translations. Ideally, $\lambda^+$ should be close to one and $\lambda^-$ should be close to zero. The actual values of the two parameters are calculated by the maximum likelihood estimation. Let $cooc(u,v)$ be the number of co-occurrences of $u$ and $v$. The *score* function is defined as:

$$score_B(u,v) = \log \frac{B(links(u,v) \,|\, cooc(u,v), \lambda^+)}{B(links(u,v) \,|\, cooc(u,v), \lambda^-)}$$

where $B(k \,|\, n,p)$ denotes the probability of $k$ being generated from a binomial distribution with parameters $n$ and $p$.

In Method C, bitext tokens are divided into classes, such as content words, function words, punctuation, etc., with the aim of producing more accurate translation models. The auxiliary parameters are estimated separately for each class.

$$score_C(u,v \,|\, Z = class(u,v)) = \log \frac{B(links(u,v) \,|\, cooc(u,v), \lambda_Z^+)}{B(links(u,v) \,|\, cooc(u,v), \lambda_Z^-)}$$

## 4 Models of sound correspondence

Thanks to its generality and symmetry, Melamed's parameter estimation process can be adapted to the problem of determining correspondences. The main idea is to induce a model of sound correspondence in a bilingual wordlist, in the same way as one induces a model of translational equivalence among words in a parallel corpus. After the model has converged, phoneme pairs with the highest likelihood scores represent the most likely correspondences.

While there are strong similarities between the task of estimating translational equivalence of words and the task of determining recurrent correspondences of sounds, a number of important modifications to Melamed's original algorithm are necessary in order to make it applicable to the latter task. The modifications include the method of finding a good alignment, the handling of null links, and the method of computing the alignment score.

For the task at hand, I employ a different method of aligning the segments in two corresponding sequences. In sentence translation, the alignment links frequently cross and it is not unusual for two words in different parts of sentences to correspond. In contrast, the processes that lead to link intersection in diachronic phonology, such as *metathesis*, are quite sporadic. The introduction of the *no-crossing-links constraint* on alignments not only leads to a dramatic reduction of the search space, but also makes it possible to replace the approximate competitive-linking algorithm of Melamed with a variant of the well-known dynamic programming algorithm (Wagner and Fischer, 1974; Kondrak,

2000), which computes the *optimal* alignment between two strings in polynomial time.

*Null links* in statistical machine translation are induced for words on one side of the bitext that have no clear counterparts on the other side of the bitext. Melamed's algorithm explicitly calculates the likelihood scores of null links for every word type occurring in a bitext. In diachronic phonology, phonological processes that lead to insertion or deletion of segments usually operate on individual words rather than on particular sounds across the language. Therefore, I model insertion and deletion by employing a constant *indel* penalty for unlinked segments.

The alignment score between two words is computed by summing the number of induced links, and applying an indel penalty for each unlinked segment, with the exception of the segments beyond the rightmost link. The exception reflects the relative instability of word endings in the course of linguistic evolution. In order to avoid inducing links that are unlikely to represent recurrent sound correspondences, only pairs whose likelihood scores exceed a set threshold are linked. All correspondences above the threshold are considered to be equally valid. In the cases where more than one best alignment is found, each link is assigned a weight that is its average over the entire set of best alignments (for example, a link present in only one of two competing alignments receives the weight of 0.5).

## 5 Implementation

The method described above has been implemented as a C++ program, named CORDI, which will soon be made publicly available. The program takes as input a bilingual wordlist and produces an ordered list of correspondences. A model for a 200-pair list usually converges after 3–5 iterations, which takes only a few seconds on a Sparc workstation. The user can choose between methods A, B, and C, described in Section 3, and an additional Method D. In Method C, phonemes are divided into two classes: non-syllabic (consonants and glides), and syllabic (vowels); links between phonemes belonging to different classes are not induced. Method D differs from Method C in that the syllabic phonemes do not participate in any links.

Adjustable parameters include the indel penalty ratio $d$ and the minimum-strength correspondence threshold $t$. The parameter $d$ fixes the ratio between the negative indel weight and the positive weight assigned to every induced link. (A lower ratio causes the program to be more adventurous in positing sparse links.) The parameter $t$ controls the tradeoff between reliability and the number of links. In Method A, the value of $t$ is the minimum number of phoneme links that have to be induced for the correspondence to be valid. In methods B, C, and D, the value of $t$ implies a likelihood score threshold of $t \cdot \log \frac{\lambda^+}{\lambda^-}$, which is a score achieved by a pair of phonemes that have $t$ links out of $t$ co-occurrences. In the experiments reported in Section 6, $d$ was set to 0.15, and $t$ was set to 1 (sufficient to reject all non-recurring correspondences). In Method D, where the lack of vowel links causes the linking constraints to be weaker, a higher value of $t = 3$ was used. These parameter values were optimized on the development set described below.

## 6 Evaluation

### 6.1 The data for experiments

The experiments in this section were performed using a well-known list of 200 basic meanings that are considered universal and relatively resistant to lexical replacement (Swadesh, 1952). The Swadesh 200-word lists are widely used in linguistics and have been compiled for a large number of languages.

The development set consisted of three 200-word list pairs adapted from the Comparative Indoeuropean Data Corpus (Dyen et al., 1992). The corpus contains the 200-word lists for a number of Indoeuropean languages together with cognation judgments made by a renowned historical linguist Isidore Dyen. Unfortunately, the words are represented in the Roman alphabet without any diacritical marks, which makes them unsuitable for automatic phonetic analysis. The Polish–Russian, Spanish–Romanian, and Italian–Serbocroatian were selected because they represent three different levels of relatedness (73.5%, 58.5%, and 25.3% of cognate pairs, respectively), and also because they have relatively transparent grapheme-to-phoneme conversion rules. They were transcribed into a phonetic notation by means of *Perl* scripts and then stemmed and corrected manually.

The test set consisted of five 200-word lists representing English, German, French, Latin, and Albanian, compiled by Kessler (2001) As the lists contain rich phonetic and morphological information, the stemmed forms were automatically converted from the XML format with virtually no extra pro-

cessing. The word pairs classified by Kessler as doubtful cognates were assumed to be unrelated.

### 6.2 Determination of correspondences in word pairs

Experiments show that CORDI has little difficulty in determining correspondences given a set of cognate pairs (Kondrak, 2002) However, the assumption that a set of identified cognates is already available as the input for the program is not very plausible. The very existence of a reliable set of cognate pairs implies that the languages in question have already been thoroughly analyzed and that the sound correspondences are known. A more realistic input requirement is a list of word pairs from two languages such that the corresponding words have the same, well-defined meaning. Determining correspondences in a list of synonyms is clearly a more challenging task than extracting them from a list of reliable cognates because the non-cognate pairs introduce noise into the data. Note that Melamed's original algorithm is designed to operate on aligned sentences that are guaranteed to be mutual translations.

| | *cooc* | *links* | *score* | valid |
|---|---|---|---|---|
| r:r | 26 | 24 | 158.7 | yes |
| n:n | 24 | 23 | 154.2 | yes |
| t:d | 18 | 18 | 122.4 | yes |
| k:k | 12 | 11 | 72.5 | yes |
| s:s | 11 | 10 | 65.7 | yes |
| f:p | 9 | 9 | 61.2 | yes |
| m:m | 10 | 9 | 58.9 | yes |
| d:t | 10 | 8 | 49.8 | no |
| l:l | 14 | 9 | 49.7 | yes |
| h:k | 7 | 7 | 47.6 | yes |

Table 2: English–Latin correspondences discovered by CORDI in noisy synonym data.

In order to test CORDI's ability to determine correspondences in noisy data, Method D was applied to the 200-word lists for English and Latin. Only 29% of word pairs are actually cognate; the remaining 71% of the pairs are unrelated lexemes. The top ten correspondences discovered by the program are shown in Table 2. Remarkably, all but one are valid. In contrast, only four of the top ten phoneme matchings picked up by the $\chi^2$ statistic are valid correspondences (the validity judgements are my own).

### 6.3 Identification of cognates in word pairs

The quality of correspondences produced by CORDI is difficult to validate, quantify, and compare with the results of alternative approaches. However, it is possible to evaluate the correspondences indirectly by using them to identify cognates. The likelihood of cognation of a pair of words increases with the number of correspondences that they contain. Since CORDI explicitly posits correspondence links between words, the likelihood of cognation can be estimated by simply dividing the number of induced links by the length of the words that are being compared. A minimum-length parameter can be set in order to avoid computing cognation estimates for very short words, which tend to be unreliable.

| $r_i$ | word pair | cognate? | $i$ | $p_i$ |
|---|---|---|---|---|
| 1 | /hɑrt/:/kord/ | yes | 1 | 1.00 |
| 2 | /hɑt/:/kalid/ | no | | |
| 3 | /snō/:/niw/ | yes | 2 | 0.66 |

Table 3: An example ranking of cognate pairs.

The evaluation method for cognate identification algorithms adopted in this section is to apply them to a bilingual wordlist and order the pairs according to their scores (refer to Table 3). The ranking is then evaluated against a gold standard by computing the *n*-point average precision, a generalization of the 11-point average precision, where $n$ is the total number of cognate pairs in the list. The *n*-point average precision is obtained by taking the average of $n$ precision values that are calculated for each point in the list where we find a cognate pair: $p_i = \frac{i}{r_i}, i = 1, \ldots, n$, where $i$ is the number of the cognate pair counting from the top of the list produced by the algorithm, and $r_i$ is the rank of this cognate pair among all word pairs. The *n*-point precision of the ranking in Table 3 is $(1.0 + 0.66)/2 = 0.83$. The expected *n*-point precision of a program that randomly orders word pairs is close to the proportion of cognate pairs in the list.

| Languages | | Method | | | |
|---|---|---|---|---|---|
| | | A | B | C | D |
| Polish | Russian | .989 | .994 | .994 | .986 |
| Romanian | Spanish | .898 | .948 | .948 | .875 |
| Italian | Serbocr. | .499 | .455 | .527 | .615 |

Table 4: Average cognate identification precision on the development set for various methods.

| Languages | | Proportion of cognates | COGNATE | JAKARTA | Method | | | |
|---|---|---|---|---|---|---|---|---|
| | | | | | A | B | C | D |
| English | German | .590 | .878 | .888 | .936 | .957 | .952 | .950 |
| French | Latin | .560 | .867 | .787 | .843 | .914 | .838 | .866 |
| English | Latin | .290 | .590 | .447 | .584 | .641 | .749 | .853 |
| German | Latin | .290 | .532 | .518 | .617 | .723 | .736 | .857 |
| English | French | .275 | .324 | .411 | .482 | .528 | .545 | .559 |
| French | German | .245 | .390 | .406 | .347 | .502 | .487 | .528 |
| Albanian | Latin | .195 | .449 | .455 | .403 | .432 | .568 | .606 |
| Albanian | French | .165 | .306 | .432 | .249 | .292 | .319 | .437 |
| Albanian | German | .125 | .277 | .248 | .156 | .177 | .154 | .312 |
| Albanian | English | .100 | .225 | .227 | .302 | .373 | .319 | .196 |
| **Average** | | **.283** | **.484** | **.482** | **.492** | **.554** | **.567** | **.616** |

Table 5: Average cognate identification precision on the test set for various methods.

Table 4 compares the average precision achieved by methods A, B, C, and D on the development set. The cognation judgments from the Comparative Indoeuropean Data Corpus served as the gold standard.

All four methods proposed in this paper as well as other cognate identification programs were uniformly applied to the test set representing five Indoeuropean languages. Apart from the English–German and the French–Latin pairs, all remaining language pairs are quite challenging for a cognate identification program. In many cases, the gold-standard cognate judgments distill the findings of decades of linguistic research. In fact, for some of those pairs, Kessler finds it difficult to show by statistical techniques that the surface regularities are unlikely to be due to chance. Nevertheless, in order to avoid making subjective choices, CORDI was evaluated on all possible language pairs in Kessler's set.

Two programs mentioned in Section 2, COGNATE and JAKARTA, were also applied to the test set. The source code of JAKARTA was obtained directly from the author and slightly modified according to his instructions in order to make it recognize additional phonemes. Word pairs were ordered according to the confidence scores in the case of COGNATE, and according to the edit distances in the case of JAKARTA. Since the other two programs do not impose any length constraints on words, the minimum-length parameter was not used in the experiments described here.

The results on the test set are shown in Table 5. The best result for each language pair is underlined. The performance of COGNATE and JAKARTA is quite similar, even though they represent two radically different approaches to cognate identification. On average, methods B, C, and D outperform both comparison programs. On closely related languages, Method B, with its relatively unconstrained linking, achieves the highest precision. Method D, which considers only consonants, is the best on fairly remote languages, where vowel correspondences tend to be weak. The only exception is the extremely difficult Albanian–English pair, where the relative ordering of methods seems to be accidental. As expected, Method A is outperformed by methods that employ an explicit noise model. However, in spite of its extra complexity, Method C is not consistently better than Method B, perhaps because of its inability to detect important vowel-consonant correspondences, such as the ones between French nasal vowels and Latin /n/.

## 7 Conclusions and future work

I have presented a novel approach to the determination of correspondences in bilingual wordlists. The results of experiments indicate that the approach is robust enough to handle a substantial amount of noise that is introduced by unrelated word pairs. CORDI does well even when the number of non-cognate pairs is more than double the number of cognate pairs. When tested on the cognate-identification task, CORDI achieves substantially higher precision than comparable programs. The correspondences are explicitly posited, which means that, unlike in some statistical approaches, they can be verified by examining individual cognate pairs. In contrast with approaches that assume a rigid alignment based on the syl-

labic structure, the models presented here can link phonemes in any word position.

Currently, I am working on the incorporation of complex correspondences into the cognate identification algorithm by employing Melamed's (1997) algorithm for discovering non-compositional compounds in parallel data. Such an extension would overcome the limitation of the one-to-one model, in which links are induced only between individual phonemes. Other possible extensions include taking into account the phonological context of correspondences, combining the correspondence-based approach with phonetic-based approaches, and identifying correspondences and cognates directly in dictionary-type data.

The results presented here prove that the techniques developed in the context of statistical machine translation can be successfully applied to a problem in diachronic phonology. The transfer of methods and insights should also be possible in the other direction.

## Acknowledgments

Thanks to Graeme Hirst, Radford Neal, and Suzanne Stevenson for helpful comments, to Michael Oakes for assistance with JAKARTA, and to Gemma Enriquez for helping with the experimental evaluation of COGNATE. This research was supported by the Natural Sciences and Engineering Research Council of Canada.

## References

Y. Al-Onaizan, J. Curin, M. Jahr, K. Knight, J. Lafferty, D. Melamed, F. Och, D. Purdy, N. Smith, and D. Yarowsky. 1999. Statistical machine translation. Technical report, Johns Hopkins University.

Ted Dunning. 1993. Accurate methods for the statistics of surprise and coincidence. *Computational Linguistics*, 19(1):61–74.

Isidore Dyen, Joseph B. Kruskal, and Paul Black. 1992. An Indoeuropean classification: A lexicostatistical experiment. *Transactions of the American Philosophical Society*, 82(5). Word lists available at http://www.ldc.upenn.edu/ldc/service/comp-ie.

Jacques B. M. Guy. 1994. An algorithm for identifying cognates in bilingual wordlists and its applicability to machine translation. *Journal of Quantitative Linguistics*, 1(1):35–42. MS-DOS executable available at http://garbo.uwasa.fi.

Brett Kessler. 2001. *The Significance of Word Lists*. Stanford: CSLI Publications. Word lists available at http://spell.psychology.wayne.edu/∼bkessler.

Philipp Koehn and Kevin Knight. 2001. Knowledge sources for word-level translation models. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, pages 27–35.

Grzegorz Kondrak. 2000. A new algorithm for the alignment of phonetic sequences. In *Proceedings of NAACL 2000: 1st Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 288–295.

Grzegorz Kondrak. 2001. Identifying cognates by phonetic and semantic similarity. In *Proceedings of NAACL 2001: 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 103–110.

Grzegorz Kondrak. 2002. *Algorithms for Language Reconstruction*. Ph.D. thesis, University of Toronto. Available at http://www.cs.toronto.edu/∼kondrak.

John B. Lowe and Martine Mazaudon. 1994. The reconstruction engine: a computer implementation of the comparative method. *Computational Linguistics*, 20:381–417.

Gideon S. Mann and David Yarowsky. 2001. Multipath translation lexicon induction via bridge languages. In *Proceedings of NAACL 2001: 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 151–158.

I. Dan Melamed. 1997. Automatic discovery of non-compositional compounds in parallel data. In *Proceedings of the Second Conference on Empirical Methods in Natural Language Processing*, pages 97–108.

I. Dan Melamed. 1999. Bitext maps and alignment via pattern recognition. *Computational Linguistics*, 25(1):107–130.

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

Michael P. Oakes. 2000. Computer estimation of vocabulary in protolanguage from word lists in four daughter languages. *Journal of Quantitative Linguistics*, 7(3):233–243.

Michel Simard, George F. Foster, and Pierre Isabelle. 1992. Using cognates to align sentences in bilingual corpora. In *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 67–81, Montreal, Canada.

Morris Swadesh. 1952. Lexico-statistical dating of prehistoric ethnic contacts. *Proceedings of the American Philosophical Society*, 96:452–463.

Jörg Tiedemann. 1999. Automatic construction of weighted string similarity measures. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, College Park, Maryland.

Robert A. Wagner and Michael J. Fischer. 1974. The string-to-string correction problem. *Journal of the Association for Computing Machinery*, 21(1):168–173.