Multilingual Cognate Identification using Integer Linear Programming

Shane Bergsma and Grzegorz Kondrak
Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada, T6G 2E8
{bergsma,kondrak}@cs.ualberta.ca

Abstract

The identification of cognates in natural languages is a crucial part of automatic translation lexicon construction and other multilingual lexical tasks. We present new methods for multilingual cognate identification using the global inference framework of Integer Linear Programming. While previous approaches to cognate identification have focused on pairs of natural languages, we provide a methodology that directly forms *sets* of cognates across groups of languages. We show improvements over simple clustering techniques that do not inherently consider the transitivity of cognate relations. Furthermore, we show that formulations that jointly link cognates across groups of natural languages achieve higher performance than traditional pairwise approaches. We also describe applications of our technique to other important problems in multilingual natural language processing.

Keywords

Cognate Identification, Multilingual, Lexicon, Integer Linear Programming, Constrained Optimization

1 Introduction

Cross-language cognate information is an important component of multilingual lexical resources. Cognates are words with similar form and meaning across natural languages. For example, the word for "heart" in Italian (*cuore*), Spanish (*corazon*), Portuguese (*coracao*), and French (*coeur*) all derive from the Latin stem *cor*. Knowing that correspondence in word spelling can imply correspondence in word meaning has assisted researchers in automatically creating translation lexicons [17, 10], sentence-aligning bilingual corpora [24, 19], and finding word correspondences in statistical machine translation [13, 25]. Knowledge of cognates is also an important part of human second-language acquisition; for example, cognates have been used to assess the readability of foreign language text by new language learners [28].

Although impressive levels of performance have been achieved at cognate identification between pairs of languages [2], there has been little recognition that cognates are actually a multilingual phenomenon. Indeed, if we propose that *cuore* in Italian is cognate with *coeur* in French, and separately judge that *coeur* is cognate with *corazon* in Spanish, then we are implicitly saying that the Italian *cuore* and Spanish *corazon* are also cognate, since cognation is a transitive relation across languages. A natural question is whether pairwise cognate identification can be improved by considering transitivity in the identification process.

Most previous approaches to cognate identification assign scores to pairs of words across two languages. These scores indicate the likelihood the two words are cognate, and are usually based on either traditional measures of string similarity such as edit distance [15] and longest common subsequence ratio (LCSR) [19], or are provided by adaptive systems learned from annotated training data [26, 20, 2]. The fundamental issue with these approaches is that while their output is pairwise similarity scores, the more natural and useful output would be sets of cognates across languages.

In this paper, we propose a cognate identification technique that operates across groups of natural languages, and which directly produces cognate sets as output. We formulate the task as a constrained optimization and find the solution with the global inference technique of Integer Linear Programming (ILP). We maximize an objective function that incorporates the scores of all cognate decisions, subject to the constraints that these decisions should respect the transitivity of cognation across languages. That is, for all words in any output interconnected set, a positive cognate decision must exist between each pair of these words (forming *cliques* in the graph-theoretic sense). A similar approach has been taken to partitioning related pieces of information for natural language generation [1], and performing coreference resolution [5].

Our work differs from previous techniques that consider cognates over groups of natural languages. Lowe and Mazaudon [16] use linguist-supplied sound-change lists to construct protoforms of modern words (the "comparative method" of language reconstruction), and then link modern words into cognate sets that share a common protoform. Oakes [21] uses pairwise similarity to determine cognates in four languages, but does not explain how inconsistencies in cognate assignments are resolved. Kondrak et al. [12] use a *post hoc* set-formation algorithm to gather groups of highly-similar potential cognates, but the search for these sets is greedy and not evaluated for set recall. Our program finds a global optimum cognate set partitioning in a single step, without setting clustering parameters or otherwise requiring user supervision.

We show the benefits of our approach in two separate application domains. First, given that a collection of words with the same meaning has been identified across a group of natural languages, we partition these words into cognate sets. The ILP system is able to automatically identify cognate sets with a higher precision and recall than a comparison approach that forms sets from interconnected components with *post hoc* processing. Thus our system achieves higher performance while simultaneously having greater simplicity and ease of specification than previous approaches to this task.

Secondly, we show that our system also achieves higher precision and recall on the pairwise identification task compared to using a similarity measure alone. We demonstrate this both for finding cognates among words known to be translations, and for automatic translation lexicon induction, where cognates are to be identified between two lexicons based purely on orthographic similarity. For the latter situation, our results are directly applicable to the lexicon induction approach of Mann and Yarowsky [17]. We use an ILP formulation that includes natural constraints such as one-to-one cognate mappings and cross-language transitivity. Remarkably, this system achieves gains of up to twenty percent in precision (for equivalent levels of recall) over an unconstrained system that uses the string similarity measure alone. On the other hand, the automatic lexicon induction task also highlights some current computational limitations; we provide approximation strategies for cases where the the optimal solution cannot be found in reasonable time.

The paper is organized as follows. In Section 2, we describe the traditional pairwise approach to cognate identification and explain its limitations. Section 3 introduces our multilingual ILP approach to cognate clustering and describes experiments and results that validate the model empirically. In Section 4, we show how an extended version of the ILP formulation can be applied to automatic lexicon induction, and again analyze our formulation experimentally. Section 5 presents our ideas for future work, including new applications of cognate sets as bridge languages in the "multipath" approach of Mann and Yarowsky [17]. We also outline a multilingual ILP-based approach to word alignment in statistical machine translation.

2 Pairwise Cognate Discovery

In this section, we describe the key components of current cognate identification systems. As mentioned in the introduction, cognates are words with similar form and meaning across natural languages. In the linguistic sense, cognates may also include words with a common ancestor but which no longer have a consistent meaning. However, for practical purposes, most work in computational linguistics has focused on *translational* cognates: similarly-spelled words that have a common origin *and* interpretation.[1] Relevant work therefore includes not only systems that find ancestrally-related cognates [11], but loan-word borrowings [7] and even proper name transliterations [8]. These computational approaches to cognate identification generally consist of two key components:

1. a semantic indication of the likelihood the two words have the same meaning, and

2. an orthographic similarity measure based on the similarity of the words' spellings

For languages with available lexical resources, the first component may simply be a bilingual dictionary; words share the same meaning if and only if they are mutual translations in the translation lexicon. For resource-poor languages, semantic similarity models based on context or frequency similarities have been used instead [23], and for closely related languages, cognates have been detected without the use of any semantic similarity module at all: cognates are detected using only the orthographic string similarity measure [17]. In Section 3, we look at the case where a translation lexicon is available, while in Section 4 we consider using string similarity alone.

A typical orthographic similarity measure is an efficient, real-valued function of pairs of words from two different languages. The measure should return higher scores for pairs more likely to be cognate, and lower scores for words likely to be unrelated. To make decisions based on this measure, we must set a threshold and classify pairs above the threshold to be cognate, and those below to be unrelated. Higher thresholds result in higher-precision, lower-recall systems, while lower thresholds catch more cognates but with a greater number of false positives.

Melamed [19] uses the Longest Common Subsequence Ratio (LCSR) as the cognate orthographic similarity measure. The LCSR of two words is equal to the length of the longest common letter subsequence between the two words, divided by the length of the longer word. Hence the range of the LCSR function is between zero (for words sharing no letters) and one (for identical words). For example, the LCSR between the Italian word *cuore* and the French word *coeur* is 0.6. This word pair will be classified as cognate if the classification threshold is below 0.6. While dynamic programming algorithms exist to compute traditional similarity measures like LCSR efficiently, these untrained approaches do not capture the regular sound correspondences between a pair of languages that help identify words of common origin.

Furthermore, nothing about previous semantic and orthographic similarity measures requires that multilingual cognate decisions be consistent. Suppose we have identified a group of words with common meaning, such as the Romance language words for *just, right* given in Fig. 1, and we would like to form cognate sets based on their computed similarities. If we decide to label all words as cognate with LCSR greater than or equal to a threshold of 0.50, we are left with the following curious conclusion: *drept* and *direito* are cognate, as are *direito* and *derecho*, but *drept* and *derecho* are not. How can we best handle such inconsistencies?

A simple approach would be to add all links between all pairs of words that are in an interconnected subgraph. For a threshold of 0.50, this does correctly

[1] In all of our experiments, all gold-standard cognate sets are composed of words with common meaning that have been judged to be cognate by linguists (see Section 3.2).

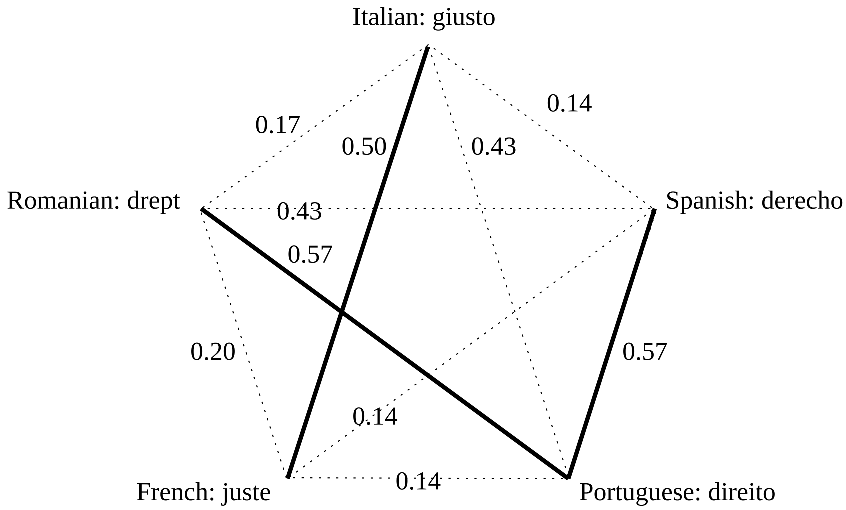

Fig. 1: *LCSR graph for Romance words corresponding to English* "just, right." *Links with LCSR* $\geq$ *0.50 are marked with solid lines.*

group the words into their true cognate sets. However, suppose the overall threshold for cognate decisions is not 0.50, but 0.43, adding the *drept* and *derecho* link as well as the *giusto* and *direito* pair. In this case, we have now interconnected all components of the graph, such that our approach will incorrectly classify words as diverse as the French *juste* and the Spanish *derecho* as cognates. Ideally, our computational procedure should know that it is better to add certain links (like *drept-derecho*) but not others (like *giusto-direito*), based on any implied cognate similarities. In the following section, we show how Integer Linear Programming provides such an approach.

3 Cognate Set Partitioning

In this application, we are presented with sets of words that share the same meaning, from a group of natural languages. Each set has one word from each language, like the set in Fig. 1. Our task is to identify which words in each set are cognate, that is, to partition the set into groups of cognates. Thus, we are automating the second step in cognate set formation; given words with common meaning, we use our orthographic similarity measure to detect those with a common origin. Determining cognates given known word correspondences is one of the fundamental tasks performed by linguists in the process of language reconstruction.

3.1 ILP Formulation

A *linear program* seeks to maximize an objective function over a set of variables, subject to a set of linear constraints. In Integer Linear Programming (ILP), these variables are further constrained to be integers. In general, finding solutions to integer linear programs is NP-hard [4, page 777], but in practice efficient solvers are available. We follow Roth and Yih [22] in using binary-$\{0, 1\}$ ILP variables to represent the decisions made by our system, and optimize as our objective function the sum of the costs/scores of the decisions that we make. The partitioning formulation we use is based on the work of Barzilay and Lapata [1].

We first consider how to use an ILP formulation for the standard pairwise approach to cognate identification. Suppose we have identified a set M of pairs of words that share a common meaning between two languages. Let $x_{\langle i,j\rangle}(m)$ be a binary variable representing a cognate decision for the mth pair of common-meaning words from language pair $\langle L_i, L_j\rangle$. The variable $x_{\langle i,j\rangle}(m)$ will be 1 when we affirm that the pair are cognates, and 0 when we decide the words are not. The standard approach is to classify each pair as cognate if their orthographic similarity is above a threshold, t. Let the similarity between the two words be $s_{\langle i,j\rangle}(m)$. We would like to associate a positive weight to an affirmative cognate decision when the pair has similarity above the threshold, and a negative weight to affirmative decisions when the similarity is below the threshold. The opposite should hold for negative decisions. Thus let the value of each positive decision be $c^+_{\langle i,j\rangle}(m) = (s_{\langle i,j\rangle}(m) - t)$ and the value of each negative decision be $c^-_{\langle i,j\rangle}(m) = (t - s_{\langle i,j\rangle}(m))$. If we are using LCSR as the similarity function with $t = 0.5$, the value of a positive cognate decision for the Italian word *cuore* and the French word *coeur* is $c^+_{I,F}(m) = 0.1$. The value of a negative decision is $c^-_{I,F}(m) = -0.1$. Our ILP formulation is to maximize the sum of the value scores over the $x_{\langle i,j\rangle}(m)$ variables:

$$max \sum_{m \in M} c^+_{\langle i,j\rangle}(m) x_{\langle i,j\rangle}(m) + c^-_{\langle i,j\rangle}(m)(1 - x_{\langle i,j\rangle}(m))$$

subject to:

$$x_{\langle i,j\rangle}(m) \in \{0, 1\} \qquad \forall m \in M$$

Subject to no further constraints, the solution to this optimization is simply having all variables be 1 when the orthographic similarity score is above the threshold (and the value is thus positive) and all variables be 0 when the orthographic similarity score is below the threshold (and the value thus negative), which is exactly the standard outcome. Note also, as in all approaches in this section, the assignment of variables for one meaning is independent of all others, and hence we could have solved $|M|$ independent integer linear programs and gotten the same output.

To enforce transitivity among decisions in sets of natural languages, we move from an ILP optimization for a pair of languages to one for all pairs within a set, adding transitivity requirements as constraints. Suppose our set M now includes words identified as having a common meaning among N different natural languages, e.g. $L_1, L_2 ... L_N$. Let S be the set of indices for all unique pairs of languages: $S = \{\langle i,j \rangle : 1 \leq i < j \leq N\}$. Our ILP formulation is now:

$$max \sum_{\langle i,j \rangle \in S} \sum_{m \in M} (c^+_{\langle i,j \rangle}(m) x_{\langle i,j \rangle}(m) + c^-_{\langle i,j \rangle}(m)(1 - x_{\langle i,j \rangle}(m)))$$

subject to:

$$\begin{aligned} x_{\langle i,j \rangle}(m) &\in \{0,1\} \\ x_{\langle i,j \rangle}(m) &\geq x_{\langle i,k \rangle}(m) + x_{\langle k,j \rangle}(m) - 1 \\ x_{\langle i,j \rangle}(m) &\geq x_{\langle i,k \rangle}(m) + x_{\langle j,k \rangle}(m) - 1 \\ x_{\langle i,j \rangle}(m) &\geq x_{\langle k,i \rangle}(m) + x_{\langle k,j \rangle}(m) - 1 \\ x_{\langle i,j \rangle}(m) &\geq x_{\langle k,i \rangle}(m) + x_{\langle j,k \rangle}(m) - 1 \\ \forall m \in M, \forall \langle i,j \rangle, &\langle i,k \rangle, \langle k,i \rangle, \langle j,k \rangle, \langle k,j \rangle \in S \end{aligned}$$

The new constraints explicitly require that if $x_{\langle i,k \rangle}(m)$ is 1 and $x_{\langle k,j \rangle}(m)$ is 1, then $x_{\langle i,j \rangle}(m)$ must also be 1 in order to satisfy the inequality, forcing the closing of the transitive link. Thus the output decisions must form fully-interconnected cliques between words wherever positive output decisions are made. Due to the constraints, the optimal solution may add links that are not present in the standard approach for the same threshold. For example, if the sum of the value scores for a set of interconnected positive links is greater than the negative values incurred by adding the transitive closure links, then these negative links will be added. In practice, however, we find the transitivity constraints have more of a conservative effect: links that would have been made in the standard approach are not made, because these would require adding quite negative values in order to satisfy transitivity. A lower threshold is needed to achieve the same recall, but this comes with higher precision.

There are many benefits to choosing an ILP approach for cognate clustering. We do not need to specify the number of clusters or perform any post-processing on our output; cognate clusters are formed naturally as the output that maximizes our objective function. Also, open-source and commercial linear programming solvers can find solutions to these kinds of problems quickly and efficiently. Finally, we can use advances in optimization and insights from other ILP applications to promote advances within the study of cognates.

3.2 Experiments

Our first set of experiments test the above formulation for finding sets of cognates in five Romance languages: Italian, Spanish, Portuguese, French, and Romanian. Our gold-standard data comes from the Comparative Indoeuropean Data Corpus [6]. The corpus contains word lists of 200 basic meanings for 95 speech varieties from the Indoeuropean family of languages, together with cognation judgements. Each word is represented in an orthographic form without diacritics using the 26 letters of the Roman alphabet. We extract all 200 sets of words and the corresponding cognate judgements for our five Romance languages.

For all approaches we use LCSR as our orthographic similarity measure, $s_{\langle i,j \rangle}(m)$, and vary the threshold, t, from 0.0 to 1.0. We choose LCSR because it is an efficiently-computable, unsupervised similarity measure, frequently used in cognate identification research [19, 26, 11, 10]. We use *lp_solve*, a free, open-source (integer) linear programming solver to perform the optimization.[2] It solves integer programs using the "Branch-and-bound" algorithm [22]. All cognate clustering optimizations returned a solution in less than a minute of computation.

We evaluate our approaches in two ways. First, we evaluate the ILP system at extracting the gold standard cognate sets. We calculate our system's *set precision,* π_s, as the proportion of sets proposed by our system which are also sets in the gold standard. The *set recall,* ρ_s, is the proportion of gold standard sets that our system correctly proposes. For a particular threshold, our evaluation measure is the *set F-score*:

$$Fscore = 2\frac{\pi_s \rho_s}{\pi_s + \rho_s}$$

We compare the ILP method to an approach that builds cognate sets from interconnected words in the LCSR-induced graph. That is, we link components that are above the LCSR threshold, and then propose as cognate sets all interconnected sub-graphs. For example, in Figure 1, the two sets would be $\{$*giusto,juste*$\}$ and $\{$*derecho,direito,drept*$\}$.

Our second evaluation considers the average pairwise precision and recall of the cognation decisions. This is the typical cognate identification evaluation for previous pairwise approaches [11, 20]. Precision is the percentage of pairwise positive cognate decisions that are also in the gold standard (i.e., the proposed pair are in the same gold-standard cognate set). Recall is the percentage of true pairwise cognate decisions that are also identified by our system. For a particular threshold, we calculate precision and recall separately for each of the ten language pairs, and return the average values. We compare the ILP system that makes these decisions jointly across all languages to a system that makes decisions based purely on the LCSR.

[2] Available at http://lpsolve.sourceforge.net/

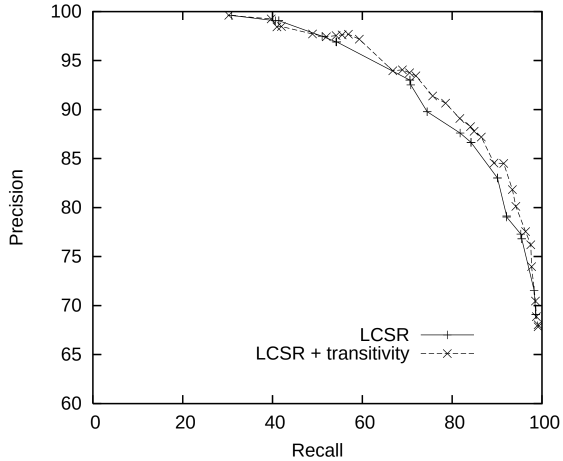

Fig. 2: *Pairwise average cognate identification precision-recall (%), within common-meaning sets, for the ten Romance languages pairs*

3.3 Results

We varied the range of the LCSR threshold and obtained maximum set F-score values of 61.6% by the ILP system and 57.7% by the system that builds sets from interconnected words. Thus with a simple, untrained measure, we can achieve approximately 4% higher maximum performance by considering transitivity constraints as part of a constrained optimization, rather than forcing transitive closure afterwards by having all interconnected components be in the same cognate set.

Beyond forming cognate sets, Fig. 2 shows that the ILP system can also achieve higher pairwise cognate identification precision than a pure LCSR-based system, for virtually all levels of identification recall. Thus even for the usual objective of detecting cognates between a single pair of languages, making those decisions within a joint optimization over a number of languages can actually achieve higher performance. This motivates the application we study in the following section, where we seek cognates between two languages without an indicator of semantic similarity.

4 Translation Lexicon Induction

We now expand our ILP formulation to consider the task of finding cognates between any two word pairs in a pair of natural languages. Like in the approach of Mann and Yarowsky [17], this can be used to automatically induce a translation lexicon; any pair of words are judged to be translations if they have a given level of orthographic similarity. The total set of translation pairs produced in this way gives the output lexicon.

4.1 ILP Formulation

We must now index our variables over all words in one language and all words in the other, not just for those with common meaning. Thus let $x_{\langle i,j\rangle}(u,v)$ again be a binary variable, but now representing a cognate decision between the uth and vth words in language L_i and language L_j, respectively. Also let the similarity and value scores range over u and v: $s_{\langle i,j\rangle}(u,v)$ and $c^{+/-}_{\langle i,j\rangle}(u,v)$. In Section 3.1, it was as if $u = v$ for all variables (and thus the only index needed was m). These expanded variables and functions can be used in place of their previous versions in the Section 3.1 formulations. Subject to no transitivity constraints, the optimizer will again set every cognate decision to true that is above the similarity threshold. Now, one word in one language could be linked to multiple words in another, if there are multiple pairs scoring above the similarity threshold. A more restrictive ILP formulation would constrain the decisions such that every word in one language can link to at most one other word in the paired language. We call this the one-to-one (1:1) constraint and encode it in the following formulation:

$$max \sum_{\langle i,j\rangle \in S} \sum_{u \in L_i} \sum_{v \in L_j} (c^{+}_{\langle i,j\rangle}(u,v) x_{\langle i,j\rangle}(u,v) + c^{-}_{\langle i,j\rangle}(u,v)(1 - x_{\langle i,j\rangle}(u,v)))$$

subject to:

$$\begin{aligned}
x_{\langle i,j\rangle}(u,v) &\in \{0,1\} \\
x_{\langle i,j\rangle}(u,v) &\geq x_{\langle i,k\rangle}(u,w) + x_{\langle k,j\rangle}(w,v) - 1 \\
x_{\langle i,j\rangle}(u,v) &\geq x_{\langle i,k\rangle}(u,w) + x_{\langle j,k\rangle}(v,w) - 1 \\
x_{\langle i,j\rangle}(u,v) &\geq x_{\langle k,i\rangle}(w,u) + x_{\langle k,j\rangle}(w,v) - 1 \\
x_{\langle i,j\rangle}(u,v) &\geq x_{\langle k,i\rangle}(w,u) + x_{\langle j,k\rangle}(v,w) - 1 \\
\sum_{t \in L_j} x_{\langle i,j\rangle}(u,t) &\leq 1 \\
\sum_{t \in L_i} x_{\langle i,j\rangle}(t,v) &\leq 1 \\
\forall u \in L_i, \forall v \in L_j, \forall w &\in L_k \\
\forall \langle i,j\rangle, \langle i,k\rangle, \langle k,i\rangle, \langle j,k\rangle, \langle k,j\rangle &\in S
\end{aligned}$$

The summation constraints ensure that at most one positive cognate decision is possible from all the words in one language to a single word in another. Without the transitivity constraints, our formulation is similar to the maximum-weight bipartite matching linear program given by Taskar et al. [25] for word alignment in statistical machine translation. Note that in our optimal solution, not every word in a given language will link with a word in one of the other languages; only those words with at least one positively-scoring potential pair-word will participate in a cognate pair. Note also that whether to enforce a one-to-one constraint depends on the ultimate application. For noisy translation lexicon induction, it makes sense to only output the single most likely translation for each word in each language. However, this constraint can be relaxed to link each word with at most two or more possible translations. In preliminary experiments for our task, a one-to-two constraint resulted in higher maximum recall at the expense of some precision (but still well above the precision of the pure LCSR approach).

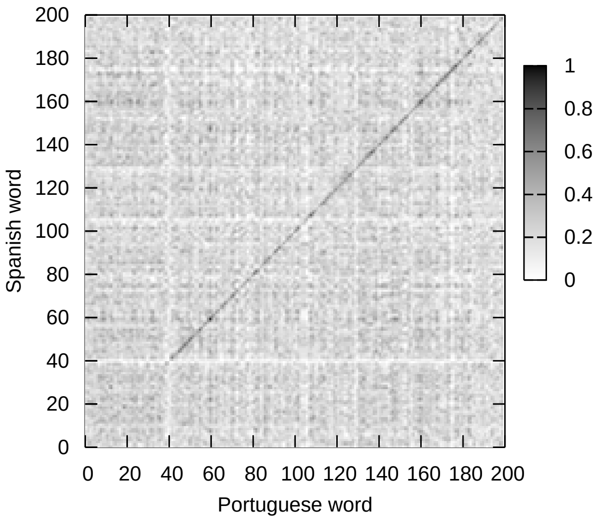

Fig. 3: *LCSR values for cross-product of Spanish-Portuguese lexicons*

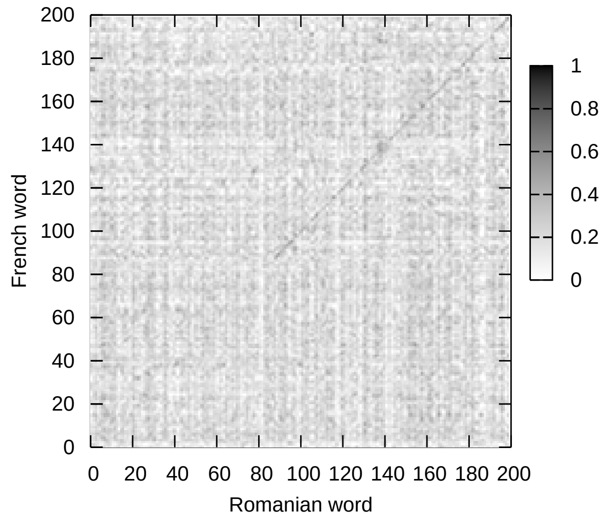

Fig. 4: *LCSR values for cross-product of French-Romanian lexicons*

4.2 Experiments

We evaluate the above approach on the same data used in Section 3, ignoring the common-meaning information and now having ILP variables for linking every word in every language to every word in every other language. Our evaluation metric is again the average cognate identification precision-recall across the ten Romance language pairs. Note that although we evaluate over pairs of languages, once again these pairwise decisions are actually made simultaneously over all ten language pairs by optimizing the given ILP formulation. Three systems are compared: defining cognate pairs in terms of LCSR alone (labelled *LCSR* in our figure), defining cognate pairs with the above ILP program but only using the one-to-one constraints (*LCSR + 1:1*), and then with both one-to-one and transitivity constraints (*LCSR + 1:1 + transitivity*).

The feasibility of lexicon induction without semantic information depends on the orthographic similarity of cognates in the specific Romance language pair. To illustrate, we provide temperature plots of the LCSR between every word pair for Spanish-Portuguese in Fig. 3 and French-Romanian in Fig. 4. Words with common meaning lie on the diagonal. The first 39 Spanish-Portuguese diagonal pairs and the first 88 French-Romanian diagonal pairs are not cognate while the remainder are. Lexicon induction can be visualized as a process whose goal is to assign "1" to the word pairs on the diagonal and "0" to off-diagonal entries. Clearly, this task is easier if the diagonal entries have higher LCSR than other pairs. Notice that in our figures, not only are there more Spanish-Portuguese cognates, but the ones which are cognate also seem to have a higher LCSR, facilitating the lexicon induction process.

Another difficulty presented by an ILP formulation to automatic lexicon induction is the sheer size of the problem. We now have 200^2 variables for each of the ten language pairs, so 400,000 in total. The total number of possible variable combinations is obviously quite large and thus we depend heavily on the efficient search of our ILP solver. Even more daunting, there are 200^3 transitivity constraints for each of the 30 unique triples of languages, for a total of 240 million constraints. Declaring these constraints in advance to our program is obviously infeasible.

We address this issue with two key techniques. First of all, it is clear that only a few of the 240 million transitivity constraints need be applied for a given optimal solution; our one-to-one constraint, for example, considerably limits the number of possible output pairings (and hence transitive triples) in the optimal solution. Thus instead of declaring the constraints in advance, we run our optimization, see which constraints are violated among our positive cognate decisions (which can be checked quite efficiently), and then add these constraints to the ILP for the next iteration. We run the algorithm until no new violations are detected. This is similar to the constraint generation approach used to detect SVM constraint violations by Tsochantaridis et al. [27]. Secondly, we reduce the burden of solving the ILP by instead solving the equivalent linear program relaxation (with variables now allowed to be any real-number between zero and one), and then rounding the output to the nearest integer. This is advantageous because, unlike integer programs, linear programs are solvable in polynomial time [4, page 777]. For their quadratic assignment approach to word alignment within translated sentences, Lacoste-Julien et al. [14] found that solving a relaxed ILP leads to no difference in performance from solving the original ILP formulation.

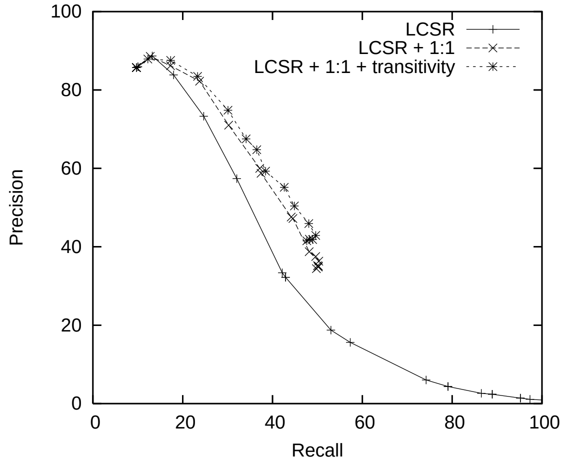

Fig. 5: *Pairwise average cognate identification precision-recall (%), for automatic lexicon induction, for the ten Romance languages pairs*

4.3 Results

In Fig. 5 we provide the experimental results. First of all, adding the one-to-one constraints strongly boosts performance over the pure *LCSR* system. At some levels of recall, gains in precision of over 20% are achieved. We found the gains in closely-related languages were greater than the gains in more distant pairs; however for a given threshold some improvement consistently occurred across all the languages. Note that the pure *LCSR* system can lower its threshold to the point where every word pair is deemed to be cognate. Hence very low precision values can be computed up to 100% recall. The constraints on the ILP systems, on the other hand, prevent arbitrarily adding low-precision links, making computing precision for higher levels of recall impossible. This is why the constrained solutions do not reach recall above 50% in Fig. 5.

The system that uses both one-to-one and transitivity constraints achieves even higher precision at equivalent values of recall. For example, for an LCSR threshold of 0.70 (30% recall and 75% precision on the plot), it has about 4% higher precision than the *LCSR + 1:1* system. For lower thresholds (all subsequent points on the plot with higher recall / lower precision), we found our ILP solver could simply not find an optimal *LCSR + 1:1 + transitivity* solution in reasonable time. For lower thresholds, there are more active constraints and more potentially high-scoring cognate pairings. Thus for thresholds below 0.70, we instead solve the linear program relaxation (as described in Section 4.2) and round the variables to zero or one. We find the relaxed *LCSR + 1:1 + transitivity* solution maintains its gains over the *LCSR + 1:1* system for all levels of recall.

These results strongly demonstrate the benefits of an ILP formulation for automatic lexicon induction. Using the same orthographic similarity measure as a naive approach that links words above the threshold, we can find cognates with higher precision and recall when using a constrained ILP formulation. We plan to investigate whether using more powerful ILP software and developing improved approximation algorithms can further boost performance.

5 Future Work

The ease of specification and the improvement in results demonstrated in the previous two sections are strong motivations for further work in developing lexical resources using Integer Linear Programming. Our next step will be to try our methods on other groups of languages, and to establish the relationship between the languages in the formulation and the cognate clustering performance. Beyond further analysis of our particular system, we now propose three especially promising general research directions.

First of all, note that aligning the words in five languages without any knowledge of meaning is perhaps an extreme situation; in practice, we may have further constraints on the alignment. For example, we may have Foreign-to-English translation dictionaries available. We could use these to constrain cognate groupings for words that have the same English translation. For the languages without translation lexicons, it would be a matter of adding them to the constrained cognate groupings based on optimizing their similarity to all the words in those sets, using our ILP formulation. This would effectively implement the multipath translation lexicon induction explored by Mann and Yarowsky [17]. They define a *bridge* word to be a word with a known English translation and a high orthographic similarity to the target word that we wish to translate. When there are multiple bridge words for a given target, their approach does not consider the similarity between the bridge words themselves, while our approach would integrate bridge word and target word similarities into one set-formation process.

Another potential application of our approach is for word alignment within statistical machine translation. This provides a very suitable problem for a transitivity-constrained ILP for several reasons. First of all, the scope is much smaller: word alignment only aligns words within aligned sentence pairs, not over entire vocabularies. Secondly, although this approach has previously been tackled on a pairwise basis, multilingual sets of sentences are readily accessible within multilingual corpora such as the Europarl corpus [9]. Finally, previous attempts at using maximum matching linear programs within word alignment have been quite successful [25, 14], but they have again been limited to pairwise cases. Like in cognate identification, separate pairwise word alignments between a set of natural language sentences implicitly suggest transitive links between interconnected words. It seems reasonable to expect that constraining these alignments to form consistent, equivalence-class word sets would allow for gains in word alignment performance.

Finally, we plan to investigate machine learned orthographic and semantic similarity models, such as those used by Bergsma and Kondrak [2], in place of simple LCSR. State-of-the-art learned models can more than double the performance of LCSR at finding translation pairs, and these gains should be additive with our improvements in clustering. Typically, partitioning and clustering using ILP have used Maximum Entropy-based pairwise models before finding the optimal sets with ILP [1, 5]. One drawback of these approaches is that they decouple the pairwise scoring and the clustering components of the set formation. It may instead be advantageous to use modern structured learning techniques to discriminatively derive the pairwise scoring functions that result in the best clustering performance. These kinds of structured learning approaches have proven successful in part-of-speech tagging [3], word alignment [25] and dependency parsing [18].

6 Conclusion

We have presented a multilingual approach to cognate identification that jointly optimizes cognate clustering across sets of natural languages. The technique of Integer Linear Programming is used to find solutions to a cognate-partitioning objective function, subject to natural constraints added to ensure consistency of decisions across languages. When words with common meaning have been identified *a priori* across five Romance languages, we have shown that our Integer Linear Programming approach to cognate set formation results in a four percent gain in cognate set formation F-score, as well as consistent gains in pairwise precision-recall across all language pairs. For the task of automatic lexicon induction, we have shown strong improvements in performance when using transitivity and one-to-one constraints. Although our fully-constrained formulation strains the computational limits of ILP, more work can be done in developing approximation algorithms or more efficient optimizations for our particular problem structure. Finally, we have outlined several possible future applications of ILP formulations in developing lexical resources for natural language processing and machine translation. In particular, our improvements in cognate set creation for lexicon induction have the potential to make a real impact on the development of electronic resources for resource-poor languages.

Acknowledgments

We gratefully acknowledge support from the Natural Sciences and Engineering Research Council of Canada, the Alberta Ingenuity Fund, and the Alberta Informatics Circle of Research Excellence.

References

[1] R. Barzilay and M. Lapata. Aggregation via set partitioning for natural language generation. In *HLT-NAACL*, pages 359–366, 2006.

[2] S. Bergsma and G. Kondrak. Alignment-based discriminative string similarity. In *ACL*, June 2007.

[3] M. Collins. Discriminative training methods for Hidden Markov Models: Theory and experiments with perceptron algorithms. In *EMNLP*, pages 1–8, 2002.

[4] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein. *Introduction to Algorithms.* The MIT Press and McGraw-Hill, second edition, 2001.

[5] P. Denis and J. Baldridge. Joint determination of anaphoricity and coreference using integer programming. In *NAACL-HLT*, pages 236–243, 2007.

[6] I. Dyen, J. B. Kruskal, and P. Black. An Indoeuropean classification: A lexicostatistical experiment. *Transactions of the American Philosophical Society*, 82(5), 1992.

[7] B.-O. Khaltar, A. Fujii, and T. Ishikawa. Extracting loanwords from Mongolian corpora and producing a Japanese-Mongolian bilingual dictionary. In *COLING-ACL*, pages 657–664, 2006.

[8] A. Klementiev and D. Roth. Named entity transliteration and discovery from multilingual comparable corpora. In *HLT-NAACL*, pages 82–88, 2006.

[9] P. Koehn. Europarl: A parallel corpus for statistical machine translation. In *MT Summit X*, pages 79–86, 2005.

[10] P. Koehn and K. Knight. Learning a translation lexicon from monolingual corpora. In *ACL Workshop on Unsupervised Lexical Acquistion*, 2002.

[11] G. Kondrak. Identifying cognates by phonetic and semantic similarity. In *NAACL*, pages 103–110, 2001.

[12] G. Kondrak, D. Beck, and P. Dilts. Creating a comparative dictionary of Totonac-Tepehua. In *ACL Special Interest Group for Computational Morphology and Phonology*, pages 134–141, 2007.

[13] G. Kondrak, D. Marcu, and K. Knight. Cognates can improve statistical translation models. In *HLT-NAACL*, pages 46–48, 2003.

[14] S. Lacoste-Julien, B. Taskar, D. Klein, and M. I. Jordan. Word alignment via quadratic assignment. In *HLT-NAACL*, pages 112–119, 2006.

[15] V. I. Levenshtein. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710, 1966.

[16] J. B. Lowe and M. Mazaudon. The reconstruction engine: A computer implementation of the comparative method. *Computational Linguistics*, 20(3):381–417, 1994.

[17] G. S. Mann and D. Yarowsky. Multipath translation lexicon induction via bridge languages. In *NAACL*, pages 151–158, 2001.

[18] R. McDonald, K. Crammer, and F. Pereira. Online large-margin training of dependency parsers. In *ACL*, pages 91–98, 2005.

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

[20] A. Mulloni and V. Pekar. Automatic detection of orthographic cues for cognate recognition. In *LREC*, pages 2387–2390, 2006.

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

[22] D. Roth and W.-T. Yih. A linear programming formulation for global inference in natural language tasks. In *CoNLL*, pages 1–8, 2004.

[23] C. Schafer and D. Yarowsky. Inducing translation lexicons via diverse similarity measures and bridge languages. In *CoNLL*, pages 207–216, 2002.

[24] M. Simard, G. F. Foster, and P. Isabelle. Using cognates to align sentences in bilingual corpora. In *Fourth International Conference on Theoretical and Methodological Issues in Machine Translation*, pages 67–81, 1992.

[25] B. Taskar, S. Lacoste-Julien, and D. Klein. A discriminative matching approach to word alignment. In *HLT-EMNLP*, pages 73–80, 2005.

[26] J. Tiedemann. Automatic construction of weighted string similarity measures. In *EMNLP-VLC*, pages 213–219, 1999.

[27] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *ICML*, 2004.

[28] S. Uitdenbogerd. Readability of French as a foreign language and its uses. In *Proceedings of the Australian Document Computing Symposium*, pages 19–25, 2005.