Bilingual Dictionary Induction as an Optimization Problem

Mairidan Wushouer[1], Donghui Lin[1], Toru Ishida[1], Katsutoshi Hirayama[2]

[1]Department of Social Informatics, Kyoto University, [2]Graduate School of Maritime Sciences, Kobe University
[1]Yoshida-Honmachi, Sakyo-Ku, Kyoto, 606-8501, Japan
[2]Fukaeminami-machi, Higashinada-ku, Kobe, 658-0022, Japan
mardan@ai.soc.i.kyoto-u.ac.jp, {lin,ishida}@i.kyoto-u.ac.jp, hirayama@maritime.kobe-u.ac.jp

Abstract

Bilingual dictionaries are vital in many areas of natural language processing, but such resources are rarely available for lower-density language pairs, especially for those that are closely related. Pivot-based induction consists of using a third language to bridge a language pair. As an approach to create new dictionaries, it can generate wrong translations due to polysemy and ambiguous words. In this paper we propose a constraint approach to pivot-based dictionary induction for the case of two closely related languages. In order to take into account the word senses, we use an approach based on semantic distances, in which possibly missing translations are considered, and instance of induction is encoded as an optimization problem to generate new dictionary. Evaluations show that the proposal achieves 83.7% accuracy and approximately 70.5% recall, thus outperforming the baseline pivot-based method.

Keywords: Bilingual Dictionary Induction, Weighted Partial Max-SAT, Constraint Satisfaction

1. Introduction

Bilingual dictionary (dictionary for short) is a valuable resource for many NLP tasks (Nakov and Ng, 2012). Unfortunately, high quality dictionaries are only available for well-resourced language pairs, such as English-French or English-Chinese; they remain unavailable for less-resourced language pairs like Uyghur and Kazakh. Hence researchers have investigated the issue of automatic creation of dictionaries: a dictionary is extracted from large scale parallel corpora (Tufiş et al., 2004; Fung and Church, 1994), and more recently, the utilization of comparable corpora has been tried (Haghighi et al., 2008; Yu and Tsujii, 2009) since parallel corpora are also scarce while, in the Internet era, monolingual data is readily available.

From the viewpoint of etymological closeness of languages, some studies directly tackled the creation of dictionaries of closely related language pairs such as Spanish and Portuguese (Schulz et al., 2004), using specific heuristics such as spelling. These studies, however, mainly focused on particular language pairs.

Another well-known approach, pivot-based induction, uses a widespread language as a bridge between less-resourced language pairs. Its naive implementation proceeds as follows. For each word in A language we take its translations in pivot language B from dictionary A-B, then for each such pivot translation, we take its translations in C language using B-C. This implementation yields highly noisy dictionaries containing incorrect translation pairs, because lexicons are generally intransitive. This intransitivity stems from polysemy and ambiguous words in the pivot language. To cope with the issue of divergence, previous studies attempted to select correct translation pairs by using semantic distances extracted from the inner structure of input dictionaries (Tanaka and Umemura, 1994) or by using additional external resources such as part of speech (Bond and Ogura, 2008), WordNet (István and Shoichi, 2009), comparable corpora (Kaji et al., 2008; Shezaf and Rappoport, 2010) and description in dictionary entities (Sjobergh, 2005).

Although the technique of adding resources to pivot-based induction is promising for improved performance (Shezaf and Rappoport, 2010), the basic methods that work with the inherent structure of input dictionaries must still be explored because: (1) It is essential for inadequately-resourced languages; (2) It is compatible with other approaches since they can be combined (Mairidan et al., 2013; Saralegi et al., 2012); (3) A large number of language resources including dictionaries are being accumulated as web services (Ishida, 2011), and the recent service computing technologies allow us to utilize existing language resources to create a new resource. (4) There is potential room for improving the quality of the induction when the missing translations are considered (Saralegi et al., 2011).

In this paper, we propose a constraint approach to the pivot-based dictionary induction to promote the quality of output dictionary *A*-*C*, where *A* and *C* are closely related languages (intra-family) while pivot language *B* is distant[1]. More precisely, we try to obtain semantic distance by constraining the types of connection in the structure of the input dictionaries based on a one-to-one assumption of intra-family language lexicons. Furthermore, Instances of pivot-based dictionary induction are represented by graphs, to which Weighted edges are added to represent missing translations. In this context, Weighted Partial Max-SAT framework (WPMax-SAT), an optimization extension of Boolean Satisfiability, is used to encode the graphs to generate optimal output dictionary. The reason for using the WPMax-SAT framework is that (1) the hidden facts such as whether a word pair is correct translation, whether a word or a translation pair is

[1]This limitation on language selection is not just because the closeness of languages is useful for detecting correct translation pairs, but the significant importance of dictionaries in making MT system for intra-family languages has been claimed by recent researches (Nakov and Ng, 2012). As to restricting pivot language to be distant, we consider the likeliness of having more information from the structure of the sources dictionaries.

missing from the dictionaries have binary states when they are unknown to machine, (2) automatic detection of correct translation pairs and missing translations whose states are bounded with certain weights can be seen as an optimization problem, which is to find most reliable translation pairs, while to add most probably missing translation, and (3) the constraints inferred from language similarity can easily be transformed to logic expression.

We designed a tool to implement the proposal using an open source SAT (Boolean Satisfiability) library. With this tool, we evaluated our approach by inducing a Uyghur-Kazakh dictionary from Chinese-Uyghur and Chinese-Kazakh dictionaries, where Uyghur and Kazakh are members of Turkic language family, while Chinese is a Sino-Tibetan language. The evaluation result revealed the efficiency of our proposal, whose detail can be found in Section 5.

The rest of the paper is organized as follows: In section 2, we discuss existing works on dictionary creation. Section 3 gives brief introduction to dictionary induction and pivot-based technique. Section 4 describes proposed method, while Section 5 details an experiment and analyzes its result. Finally, we end with the a discussion and conclusion.

2. Related Work

A very early attempt to create dictionary from existing dictionaries was by Tanaka (Tanaka and Iwasaki, 1996), who used a pivot language. They approached lexical intransitivity divergence through Inverse Consultation (IC). The IC method measures the intersection of two pivot word sets: the set of pivot translations of a word w in A language, and the set of pivot translations of each word in *C* language that is a candidate for being a translation of w. The IC method generally requires the intersection set contains at least two synonymous words. Variations of this method have also been proposed (Saralegi et al., 2011)(Kaji and Aizono, 1996)(Bond et al., 2001)(Ahn and Frampton, 2006), where dictionaries were extracted from non-aligned corpora, multiple input dictionaries and parallel corpora. A weakness of the IC method is that it relies on synonymous words to identify correct translations, which may result in low recall if the pivot language has few synonymous words or the input dictionaries are heavily incomplete.

With the assumption that more pivot languages could provide extra information for evaluating semantic distance of cross-lingual word pairs, one proposal requires more input dictionaries (Soderland et al., 2009). They represent the input dictionaries as an undirected graph, vertices reprehending the words from all the inputs, and edges representing translation pairs. The new translation pairs are induced based on cycles in the undirected graph, which means that the existence of multiple paths between two words in different languages. This multiple-pivot idea is similar to the IC method, but its use of multiple pivot languages eliminates its dependency on synonym-rich input dictionaries. The new problem is the need to find suitable multiple input dictionaries.

One study (Skoumalova, 2001) presented a method for inducing dictionary based on transduction models of cognate pairs via a bridge language, where dictionaries within language families are induced using probabilistic string edit distance models, while dictionaries for arbitrary distant language pairs are then generated by combination of these intra-family translation models and one or more cross-family dictionaries. In this study, relatively high accuracy was reported when generating English-Portuguese, English-Norwegian and English-Ukrainian dictionaries.

Most of existing approaches are promising when there are extra language resources that offer word sense data or permit semantic distance between cross-lingual word pairs to be assessed. Yet, Tanaka (Tanaka and Umemura, 1994) has been the only study to try to create dictionaries from just wo input dictionaries, with pivot synonymous words a the only information to support semantic distance determination (hence it is often seen as baseline method for the evaluations). In our work we also focus on creating dictionaries from only pairs of input dictionaries since many languages still lack adequate languages resources. Our approach, modeling complete structure of input dictionaries, can handle the incompleteness of input dictionaries to some extent, and performs better in the case of two closely related languages. To best of our knowledge, our work is first attempt to apply SAT technology to dictionary construction; one study tackled the ambiguity problem of word selection in machine translation by using a similar method (Matsuno and Ishida, 2011).

3. Pivot-based dictionary induction

Let $D_{l_1-l_2}$ denote the dictionary of l_1 and l_2 languages. Connecting $D_{\text{A-B}}$ and $D_{\text{B-C}}$ via *B* language forms a graph with at least one connected component, which we, following (Soderland et al., 2009), call a *transgraph*.

A transgraph is defined as an undirected graph $G = \{V, E\}$*, in which vertex* $w^l \in V$ *is a word in a language* $l \in \{A, B, C\}$*, and an edge* $e(w_i^{l_1}, w_j^{l_2})$ *is a translation relation of its endpoint words.* V^{A}, V^{B} and V^{C} denote $\bigcup w_i^{\text{A}}$, $\bigcup w_i^{\text{B}}$ and $\bigcup w_i^{\text{C}}$, respectively.

It should be noted that although $D_{\text{A-B}}$ and $D_{\text{B-C}}$ are usually directional, for example, $D_{\text{A-B}}$ was made with the intention to translate words in A language to B language , ignoring directionality is possible, because it is not only accordance with the *reversibility principle* found in lexicographic literature (Tomaszczyk, 1986), but the initial noisy dictionary $D_{\text{A-C}}$ would provide the most complete candidate set possible. Thus, $D_{l_1\text{-}l_2}$ is identical to $D_{l_2\text{-}l_1}$ either in terms of *transgraph*s or the set of translation pairs it contains. The small *transgraph* in Fig. 1-a is used as an example in this paper.

Using pivot language is well known in research on machine translation (Tanaka et al., 2009). However, in this context, the idea of pivot-based induction is to induce new dictionary $D_{\text{A-C}}$ from existing $D_{\text{A-B}}$ and $D_{\text{B-C}}$, where a pair of words in *A* and *C* languages is added to $D_{\text{A-C}}$ if they have same translation in *B*, or accordingly, if they are adjacent to the same pivot word in a *trangraph*. Such a $D_{\text{A-C}}$ may include both correct and incorrect translation pairs. Taking Uyghur-English-Kazakh as an example, the English word *tear* is the translation of the Uyghur word *yash*, but only in the sense of liquid from the eyes. Further translating *tear* into Kazakh yields both the correct translation *jash* and an incorrect one,

jirtiw (to rip). Identifying such an incorrect translation is challenging (see Fig. 1), because, unfortunately, most dictionaries lack comparable information about senses in their entries. So it is not possible to map entries and translation equivalents according to their corresponding senses. As an alternative, most previous studies try to guide this mapping according to semantic distances extracted from the dictionaries themselves or external resources.

One can create a dictionary of two languages just by propagating their lexicons. This dictionary would have highest recall and lowest accuracy (precision). It is important to note that the basic pivot approach can often be the first and easiest step to increasing the accuracy of such a dictionary. In many cases, the accuracy obtained from first step is so low enough that the resulting dictionary is far from practical.

3.1. One-to-one Assumption and Constraints

Intra-family languages share a significant number of cognates[2]. A cognate pair is usually a direct translation (one-to-one equivalent) (Melamed, 2000). A classical lexicostatistical study of 15 Turkic languages[3] showed that the cognates shared by these languages scale from 44% to 94% of their lexicons, from which we may assume that relatively large parts of their lexicons are one-to-one mapping.

Taking account such facts, we make a following assumption: *lexicons of intra-family languages are one-to-one mapping*[4]. That is, if *A* and *C* are intra-family, for any w^{A} there exists a unique w^{C}, such that they have exactly same meaning. Such pair is called a one-to-one pair, and denoted by $\mathbb{O}(w^{\mathrm{A}}, w^{\mathrm{C}})$. Accordingly, $\neg\mathbb{O}(w^{\mathrm{A}}, w^{\mathrm{C}})$ denotes other than one-to-one pair. We sometimes use the term one-to-one pair candidate to refer a pair of words whose state of being one-to-one mapping has yet to be determined.

Although such assumption may be strong for the general case, we consider it is quite reasonable for the case of intra-family languages. Moreover, a similar assumption is used in the parallel corpora approach has already been evaluated by Melamed (Melamed, 2000) and Vulic (Vulić and Moens, 2012).

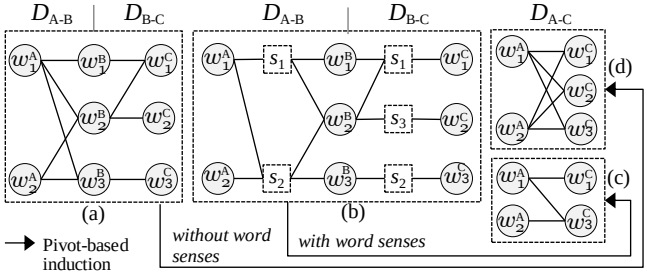

Figure 1: Pivot-based dictionary induction and ambiguity problem: w_2^{C} can be the translation of w_2^{B} for *s*ense(s_3) difference from sense(s_1) for which w_2^{B} is the translation of w_1^{A}.

[2]Words that are derived from same origin, and similar in both spelling and meaning (e.g. "segiz" [Kazakh], "säkkiz" [Uyghur] for English gloss "eight").

[3]http://turkic-languages.scienceontheweb.net

[4]The utilization of similar assumption in the parallel corpora approach has been evaluated (Melamed, 2000; Vulić and Moens, 2012)

We instantiate the one-to-one assumption by the following constraints.

Constraint 1 (*Candidate Generation*): *Given the pair of words w^{A} and w^{C} in a* transgraph*, they can be one-to-one pair candidate iff they are in the same* transgraph *and connected via at least one pivot word.*

In other words, a word pair will be one-to-one pair candidate and subjected to further evaluation only if they share at least one sense in the pivot language. For instance, in Fig. 1.a, the one-to-one candidates are all the six possible combination between $\{w_1^{\mathrm{A}}, w_2^{\mathrm{A}}\}$ and $\{w_1^{\mathrm{C}}, w_2^{\mathrm{C}}, w_3^{\mathrm{C}}\}$. This constraint may raise a question on the potential one-to-one pair candidates which are hidden because of data incompleteness (missing pivot words or translations). However, we ignored this for the simplicity.

Constraint 2 (*Symmetry*): *Given the pair of words w^{A} and w^{C} in a* transgraph*, if they are one-to-one pair, then they should be symmetrically connected through pivot word(s).*

In other words, a one-to-one pair must share same meaning(s) in the pivot language. Note that a path through a pivot word might maintain at least one common word sense along the edges. This constraint is formulated as follows:

$$\mathbb{O}(w_i^{\mathrm{A}}, w_j^{\mathrm{C}}) \rightarrow \bigwedge_{w_k^{\mathrm{B}} \in V^{\mathrm{B}}_{w_i^{\mathrm{A}}} \cap V^{\mathrm{B}}_{w_j^{\mathrm{C}}}} e(w_i^{\mathrm{A}}, w_k^{\mathrm{B}}) \wedge \bigwedge_{w_p^{\mathrm{B}} \in V^{\mathrm{B}}_{w_i^{\mathrm{A}}} \cap V^{\mathrm{B}}_{w_j^{\mathrm{C}}}} e(w_j^{\mathrm{C}}, w_p^{\mathrm{B}}) \tag{1}$$

Where $V^{\mathrm{B}}_{w_i^{\mathrm{A}}}$ and $V^{\mathrm{B}}_{w_j^{\mathrm{C}}}$ are the meaning sets of w_i^{A} and w_j^{C} in *B* language, respectively. For example, in Fig. 1.a, if $(w_1^{\mathrm{A}}, w_1^{\mathrm{C}})$ is one-to-one pair, then the 6 edges: $e(w_1^{\mathrm{A}}, w_1^{\mathrm{B}})$, $e(w_1^{\mathrm{A}}, w_2^{\mathrm{B}})$, $e(w_1^{\mathrm{A}}, w_3^{\mathrm{B}})$, $e(w_1^{\mathrm{C}}, w_1^{\mathrm{B}})$, $e(w_1^{\mathrm{C}}, w_2^{\mathrm{B}})$ and $e(w_1^{\mathrm{C}}, w_3^{\mathrm{B}})$ must exist in the *transgraph*, among which $e(w_1^{\mathrm{C}}, w_3^{\mathrm{B}})$ is not present. We consider such an edge is possibly missing, which means that the corresponding translation might not have been included in input dictionary when it was built.

Constraint 3 (*Uniqueness*): *Given the pair of words w^{A} and w^{C} in a* transgraph*, if they are one-to-one pair, then they should be unique, such that all other candidates involving w^{A} or w^{C} are not one-to-one pairs.*

For example, in Fig. 1.a, if $(w_1^{\mathrm{A}}, w_1^{\mathrm{C}})$ is a one-to-one pair, then we assert that $(w_1^{\mathrm{A}}, w_2^{\mathrm{C}})$, $(w_1^{\mathrm{A}}, w_3^{\mathrm{C}})$ and $(w_2^{\mathrm{A}}, w_1^{\mathrm{C}})$ are not one-to-one pairs. This constraint is formulated as follows:

$$\mathbb{O}(w^{\mathrm{A}}, w^{\mathrm{C}}) \rightarrow [\bigwedge_{i=2}^{n} \overline{\mathbb{O}}(w^{\mathrm{A}}, w_i^{\mathrm{C}}] \wedge [\bigwedge_{j=2}^{m} \overline{\mathbb{O}}(w^{\mathrm{C}}, w_j^{\mathrm{A}}] \tag{2}$$

3.2. Data Incompleteness

The completeness of input dictionaries is seldom guaranteed: (1) a pivot word is missing so that some translation pair for the $D_{\text{A-C}}$ are not identified, (2) a non-pivot word is missing (a vertex $w_i^{\mathrm{A}} \in V^{\mathrm{A}}$ or $w_i^{\mathrm{A}} \in V^{\mathrm{C}}$ is missing in a *transgraph*), or (3) a translation $tr(w_i^{\mathrm{A}}, w_j^{\mathrm{B}})$ or $tr(w_i^{\mathrm{C}}, w_j^{\mathrm{B}})$ is missing (an edge is missing in a *transgraph*).

Apparently, first two problems cannot be resolved without additional resources. So they are not considered in this paper. The third one, however, is vital because the one-to-one assumption demands a symmetric connection between one-to-one pairs in the *transgraph*, so that any possible missing edge breaks the symmetric connection between w_i^{C} and w_j^{B}

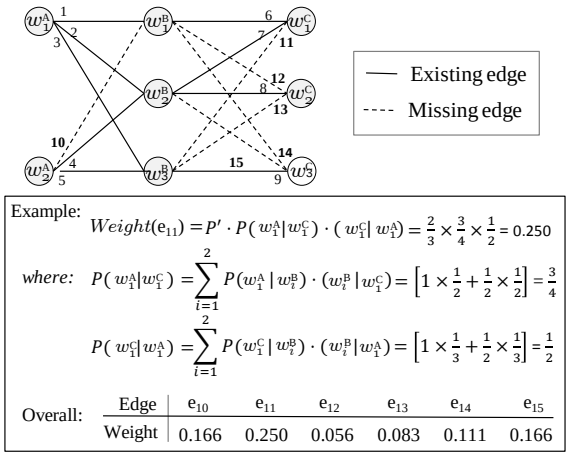

Figure 2: A *transgraph* with possibly missing edges and their weights. Note that dotted lines represent the missing edges $e_{10} \sim e_{15}$;$P(w_1^l|w_2^l)$ is the probability that w_1^l is the translation of w_2^l; P' is the maximum probability of w^{A} or w^{C} having its one-to-one equivalent in a *transgraph*.

which ensures that the pair $(w_i^{\text{A}}, w_i^{\text{C}})$ cannot be detected as a one-to-one pair. Moreover, missing edges are hard to avoid since the input dictionaries are usually independently created, and their completeness is seldom guaranteed. However, adding missing edges into the *transgraph* makes it complete, and, thus, makes induction more accurate. We assign probability value as a weight, $Weight(e)$, to the missing edge e, indicating the likelihood of it being incorrectly missed. Although many methods are proposed for such calculation, we employ a simple statistical method (Nakov and Ng, 2012) for the sake of simplicity as in Formula 3 [5]. However, one can extend our method by adopting different calculation or even using external knowledge to gain more accurate weight.

$$Pr(w^{\text{A}}, w^{\text{C}}) = Pr^{max} \cdot Pr(w^{\text{A}}|w^{\text{C}}) \cdot Pr(w^{\text{C}}|w^{\text{A}}) \quad (3)$$
$$where \quad Pr^{max} = \frac{Min(|V^{\text{A}}|,|V^{\text{C}}|)}{Max(|V^{\text{A}}|,|V^{\text{C}}|)}$$
$$Pr(w^{\text{A}}|w^{\text{C}}) = \sum Pr(w^{\text{A}}|w_i^{\text{B}}) \cdot Pr(w_i^{\text{B}}|w^{\text{C}})$$
$$Pr(w^{\text{C}}|w^{\text{A}}) = \sum Pr(w^{\text{C}}|w_i^{\text{B}}) \cdot Pr(w_i^{\text{B}}|w^{\text{A}})$$

Notice that Pr^{max} represents the maximum probability that $w^{\text{A}} \in V^{\text{A}}$ or $w^{\text{C}} \in V^{\text{C}}$ has a one-to-one equivalent in the *transgraph*; i and j are the indexes of pivot translation. Fig. 2 shows the edges that are considered to be missing in the *transgraph* given in Fig. 1.a; their weights obtained by Formula 3. A weight calculation is simply demonstrated in Fig. 2.

3.3. Objective Function

Edge e is allowed to be added into the *transgraph* if it has non-zero probability p of having been incorrectly missed. If it is added, then a certain cost, $1 - p$ (or $1 - Weight(e)$) , is to be paid. We define the process of extracting one-to-one pairs from a *transgraph* as an optimization problem; the objective is to extract as many one-to-one pairs as possible while minimizing the cost of edge addition, where cost is defined as the probability that an edge does not exist (or turns out to be not missing).

We used a Boolean Optimization framework, WPMax-SAT, to formulate the induction to generate the optimal one-to-one pair set, since the facts that whether a pair is one-to-one mapping, whether an edge has been missing, and constraints can easily be represented by Boolean variables and expressions. In the next section, we will describe how we formalize this problem within the Weighted Partial Max-SAT framework, and then evaluate CNF (Conjunctive Normal Form) formulas to obtain one-to-one pairs.

4. SAT-based Formulation

4.1. Preliminaries

Boolean Satisfiability (SAT) is the problem of finding, if it exists, an assignment to the variables that satisfies the Boolean formula expressed in CNF (Conjunctive Normal Form) (Biere et al., 2009). A *literal* is a Boolean variable υ or its negation $\overline{\upsilon}$; a *clause* is a disjunction (logical OR) of literals (e.g., $\upsilon_1 \vee \overline{\upsilon}_2 \vee \overline{\upsilon}_3$). Each clause consists of ORed literals (a Boolean variable or its negation). A CNF φ is the conjunction (logical AND) of m clauses $c_1, \ldots, c_m$, where c_i is a disjunction of k_i literals. The φ is *satisfied* if it evaluates to 1 (*TRUE*), such that all $c_i \in \varphi$ are evaluate to 1. There are several extensions to the SAT problem. One such extension of interest is Weighted partial Max-SAT (WPMax-SAT) (Fu and Malik, 2006) which aims to satisfy a partial set of clauses. In a WPMax-SAT problem, clauses are assigned weights, and are separated into hard and soft types. Hard clauses have maximum (usually represented by infinity) weights and all must be satisfied, while soft clauses need to be satisfied such that the sum of the weights of the satisfied soft clauses is maximized or sum of the weights of the unsatisfied (falsified) is minimized.

Formally, a WPMax-SAT is multiset of weighted clauses $\varphi = \{(c_1, \omega_1), \ldots, (c_m, \omega_m), (c_{m+1}, \infty), (c_{m+m'}, \infty)\}$, where the first m clauses (φ^+) are soft and last m' clauses (φ^∞) are hard. WCNF formula φ is the problem of finding an assignment to $\mathbb{V}$ that minimizes the cost of the assignment on φ. If the cost is infinity, it means that we must falsify a hard clause, and say that the multiset is unsatisfiable.

4.2. Encoding

As a first step of casting our problem in WPMax-SAT form, we apparently need a variable to denote whether a given word pair is a one-to-one pair. Moreover, another variable is also needed to represent whether an edge is missing, since the identification of a one-to-one pair requires the existence of particular edges. Overall, we say x and y to denote one-to-one pair candidates and edges in the *transgraph*, respectively:

- $x_{i,j}$, representing pair $(w_i^{\text{A}}, w_j^{\text{C}})$, turns `TRUE` if it is one-to-one pair; turns `FALSE` otherwise.
- $y_{i,j}^{\text{A}}$, representing an edge $e(w_i^{\text{A}}, w_j^{\text{B}})$, turns `TRUE` if it must exist; turns `FALSE` otherwise.

[5] Using the formula **??** the maximum probability value might be exceed 1. If it is the case, the obtained probability values of all the candidates have to be normalized to the range from 0 to 1 dividing by number of pivot word in a given *transgraph*.

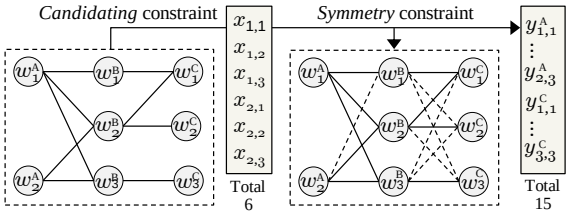

Figure 3: The process of creating variables for a $transgraph$.

- $y^{\mathrm{C}}_{i,j}$, represents an edge $e(w^{\mathrm{C}}_i, w^{\mathrm{B}}_j)$, turns TRUE if it must exist; turns FALSE otherwise.

Note that Y^l_{ext} and Y^l_{mis}, $l \in \{A, C\}$ are used to denote the set of preexisting edges and missing edges, respectively. Fig. 3 illustrates how variables are created for a *transgraph*.

Before evaluating a WPMax-SAT problem by using a solver, it must usually be encoded to CNF. There are several ways of encoding most problems (Biere et al., 2009), yet the choice of encoding can be as important as the choice of search algorithm. However for our problem, a resolution approach based on simple Boolean algebra rules such as $\upsilon_1 \rightarrow \upsilon_2 \wedge \upsilon_3 \Leftrightarrow (\neg\upsilon_1 \vee \upsilon_2) \wedge (\neg\upsilon_1 \vee \upsilon_3)$, is used, because we consider it is most appropriate way to encode the constraints in our problem.

We use hard clauses to encode all the constraints that must be satisfied, and an apparent constraint: *a preexisting edge cannot be deleted* (this constraint is added because preexisting edges need to be protected from being deletion, since they are assumed to be created by humans). Meanwhile, the missing edges are encoded with soft clauses since adding an edge is not mandatory. In the following clause formulations, φ^{∞} indicates hard, while φ^{+} indicates soft.

(1) Hard clauses in encoding to prevent edge deletion:

$$\varphi^{\infty}_1 = [\bigwedge (y^{\mathrm{A}}_{i,j}, \infty)] \wedge [\bigwedge (y^{\mathrm{C}}_{i,j}, \infty)]$$
$$\text{where } y^{\mathrm{A}}_{i,j} \in Y^{\mathrm{A}}_{ext},\ y^{\mathrm{C}}_{i,j} \in Y^{\mathrm{C}}_{ext}$$

(2) Soft clauses in encoding for addition of missing edges:

$$\varphi^{+} = [\bigwedge (\neg y^{\mathrm{A}}_{i,j}, 1 - \omega^{\mathrm{A}}_{i,j})] \wedge [\bigwedge (\neg y^{\mathrm{C}}_{i,j}, 1 - \omega^{\mathrm{C}}_{i,j})]$$
$$\text{where } y^{\mathrm{A}}_{i,j} \in Y^{\mathrm{A}}_{mis},\ y^{\mathrm{C}}_{i,j} \in Y^{\mathrm{C}}_{mis},\ \omega \text{ is the weight}$$

(3) Hard clauses encoding Symmetry Constraint:

$$\varphi^{\infty}_2 = [\bigwedge (\neg x_{i,j} \vee y^{\mathrm{A}}_{i,j}, \infty)] \wedge [\bigwedge (\neg x_{i,j} \vee y^{\mathrm{C}}_{i,j}, \infty)]$$

(4) Hard clauses encoding Uniqueness Constraint:

$$\varphi^{\infty}_3 = [\bigwedge_{j \neq k} (\neg x_{i,j} \vee \neg x_{i,k}, \infty)] \wedge [\bigwedge_{i \neq k} (\neg x_{i,j} \vee \neg x_{k,j}, \infty)]$$

4.3. Finding Solution

CNF formula $\varphi = \varphi^{+} \wedge \varphi^{\infty}_1 \wedge \varphi^{\infty}_2 \wedge \varphi^{\infty}_3$ can be evaluated by a Max-SAT solver to output an optimal variable assignment (solution). However, any satisfiable assignment on φ ends up with minimum cost, equally zero, because no hard clause in φ requires x variables to evaluate to TRUE (doing so may need edge addition that eventually increases the cost of assignment). However, we resolve this by adding a new hard clause whose constraint is that there at least ONE x variable must evaluate to TRUE. This clause is disjunction of all the x variables: $\varphi^{\infty}_4 = (\bigvee x_{i,j}, \infty)$.

Therefore, the complete CNF becomes $\varphi = \varphi^{+} \wedge \varphi^{\infty}_1 \wedge \varphi^{\infty}_2 \wedge \varphi^{\infty}_3 \wedge \varphi^{\infty}_4$; solving it returns an optimal assignment with minimum cost[6], which equals 0 when no edge is added, or exceeds 0 when the most probably missing edge(s) is added.

In an optimal assignment, we can have single variable $x_{m,k} \in \bigcup x_{i,j}$ evaluated to TRUE, while all others, if available, are falsified. In this case, the corresponding pair $(w^{\mathrm{A}}_m, w^{\mathrm{C}}_k)$ is considered to be the most reliably correct one-to-one pair. We add it into output dictionary $D_{\text{A-C}}$, and re-generate φ by reflecting the awareness of $\mathbb{O}(w^{\mathrm{A}}_m, w^{\mathrm{C}}_k)$, which can be encoded by a new hard clause $(x_{m,k}, \infty)$. The re-generated φ is again evaluated by the solver to identify one more one-to-to pair. The same process is iterated until φ becomes unsatisfiable, at which point the output dictionary is complete (as in Algorithm 1).

We describe how two one-to-one pairs are extracted from the example *transgraph* in Fig. 1-a after three iterations. Before solving the problem, a corresponding $\varphi = \varphi^{+} \wedge \varphi^{\infty}_1 \wedge \varphi^{\infty}_2 \wedge \varphi^{\infty}_3$ is formed, where $\varphi^{\infty}_4 = x_{1,1} \vee x_{1,2} \vee x_{1,3} \vee x_{2,1} \vee x_{2,2} \vee x_{2,3}$.

1. φ is evaluated: an optimal solution is found, where $x_{1,1}$ is assigned to TRUE, since the cost, 0.750, of adding the edge e_{11} is the minimum. Then the fact that $x_{1,1}$ is TRUE represents a new hard constraint and forms corresponding clause $(x_{1,1}, \infty)$ which further becomes a part of φ. Meanwhile, φ^{∞}_4 updates to $x_{1,1} \vee x_{1,2} \vee x_{1,3} \vee x_{2,1} \vee x_{2,2} \vee x_{2,3}$ to prevent deadlock. This iteration produces the one-to-one pair $(w^{\mathrm{A}}_1, w^{\mathrm{C}}_1)$.

2. φ is evaluated: an optimal solution is found, where the variable $x_{2,3}$ is assigned to TRUE, since the cost, 0.834, of adding the edge e_{15} is the minimum. Likewise, $x_{2,3}$ = TRUE represents a new hard constraint and forms corresponding clause $(x_{2,3}, \infty)$ which is attached to φ. Meanwhile, φ^{∞}_4 becomes $x_{1,2} \vee x_{1,3} \vee$

Algorithm 1 Extracting one-to-one pairs from a *transgraph*

Input: G – a *transgraph*
Output: R – Set of one-to-one pairs

```
(φ, map) ← encode G to CNF
/* φ = φ+ ∧ φ1∞ ∧ φ2∞ ∧ φ3∞ ∧ φ4∞
   where φ4∞ = (⋁ x_{i,j}, ∞)*/
X ← ∅
while φ is satisfied do
    A ← take an optimal assignment on φ
    x_{m,k} ← take x_{i,j} ∈ A ,where x_{i,j} ∉ X ∧ x_{i,j} = 1
    X ← X ∪ {x_{m,k}}
    φ4∞ ← φ4∞ − x_{m,k} /*exclude x_{m,k} from ⋁ x_{i,j}*/
    φ ← φ ∧ (x_{m,k}, ∞) /* create new clause and add to φ */
end while
return R ← map(X)
```

[6]Notice that the optimal assignment may not be unique, since several assignments may have equally minimum cost. If it is the case, solver selects one randomly.

$x_{2,1} \vee x_{2,2}$. This iteration produces the one-to-one pair $(w_2^{\mathrm{A}}, w_3^{\mathrm{C}})$.

3. φ is evaluated: no solution is found (problem is unsatisfiable) because, in this case, any attempt to have an x variable assigned `TRUE` violates Uniqueness Constraint imposed by φ_3^{∞}.

5. Experiment

We designed a tool (see a screen-shot in Fig. 5) to implement the proposal using sat4j[7] as the default solver due to its flexibility in integration with third-party software. With this tool, we evaluated our approach by inducing $D_{ug\text{-}kk}$ from $D_{zh\text{-}ug}$ and $D_{zh\text{-}kk}$ (see Table 3 for details), where ug (Uyghur) and kk (Kazakh) are Turkic languages, while zh (Chinese) belongs to the Sino-Tibetan language family.

5.1. Tool Implementation

The highlights of the tool's main features are as follows.

- Provides many options for pre-processing the input dictionaries.
- Displays *transgraph*s using dynamic graph components (see Fig. 4), so that users can easily observe the induction process and even interact with *transgraph*s to manually modify their structure (e.g. annotating known one-to-one pairs or adding missing edges).
- Produces comprehensive statistics of the structure of input dictionaries, *transgraph*s, CNF encoding, solutions and some other details such as computational performance.
- Bilingual human experts can use it to evaluate automatically selected sample pairs easily.

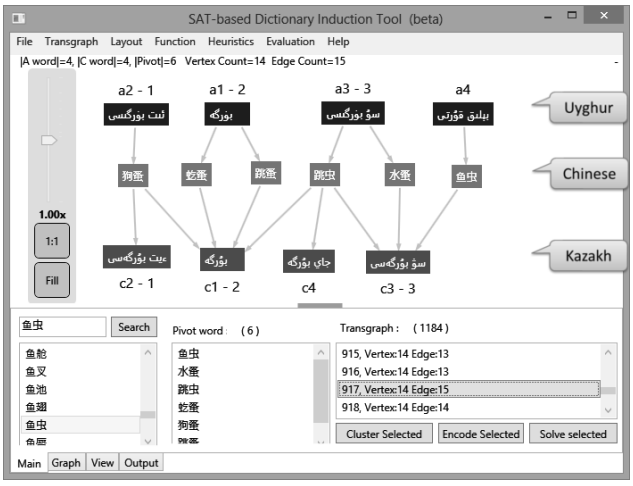

Figure 4: A screen-shot: evaluation of a *transgraph* with 14 vertices, which resulted in three one-to-one pairs with 100% accuracy and 75% recall.

[7]Library of SAT and Boolean Optimization solver: http://www.sat4j.org

Dictionary	zh words	ug / kk words	Translation Pair
$D_{zh\text{-}ug}$	52, 478	70, 989	118,805
$D_{zh\text{-}kk}$	52, 478	102, 426	232,589

Table 1: Details of dictionaries for experiment

transgraph		$\lvert V^{zh}\rvert$	$\lvert V^{ug}\rvert$	$\lvert V^{kk}\rvert$	Edge
#1 ∼ #1183	Smallest	2	2	3	6
	Largest	13	21	27	71
#1184		35,539	47,893	66,693	287,966

Table 2: *transgraphs* constructed from dictionaries

5.2. Experiment Settings

Connecting $D_{zh\text{-}ug}$ and $D_{zh\text{-}kk}$ resulted in 12,393 *transgraphs*. Among them, we selected only 1184, each of which involves at least two pivot words (see Table 4). Because, in theory, our approach does not make sense to others (all the possible assignments always have equal cost, so a random selection is enough). Surprisingly, one of them, #1184, is remarkably large; it contains 69% of all vertices and 82% of all edges. Encoding this large *transgraph*, however, resulted in a CNF formula with 16,879,348 variables and 46,059,686 clauses. We were unable to evaluate it using sat4j solver in our experimental hardware environment[8] due to its high computational complexity. Hence for experimental purposes, we further partitioned *transgraph* #1184 into 150 smaller subgraphs using a graph partitioning algorithm (Dhillon et al., 2005). Since the goal of graph partitioning is to minimize the number of edges that cross from one subgroup of vertices to another, we consider adopting such an algorithm is reasonable. However, as one can implement our proposal with more efficient SAT solvers, and rerun the experiment with greater hardware capacity, the step of partitioning may not be needed. Also, since graph partitioning is not a focus in our work, we have not made any comparison study on relevant algorithms with our data. Instead, one that offered easy integration with our tool was preferred.

We independently processed these two groups of *transgraphs* (1183 in Group I, and 150 in Group II), and evaluated the induction result of each group. The overall values were also calculated by averaging. To measure the recall, we set an upper bound value that represents the maximum number of possible one-to-one pairs available in a *transgraph*. It is given by $Min(|V^{\mathrm{A}}|, |V^{\mathrm{C}}|)$ for a *transgraph* with $|V^{\mathrm{A}}|$ words in A language and $|V^{\mathrm{C}}|$ in C language. Moreover, if there are n *transgraph*s, the overall upper bound value should equal $\sum_{i=1}^{n} Min(|V_{g_i}^{\mathrm{A}}|, |V_{g_i}^{\mathrm{C}}|)$. To evaluate accuracy, samples were evaluated by bilingual human experts.

5.3. Result and Analysis

Fig. 5 illustrates distribution of maximum expected and actual extracted one-to-one pairs from *transgraphs* in each

[8]Hardware – CPU: Intel(R) Core(TM) i5 2.40GHz ; 8GB RAM Software – Dictionary induction tool with Sat4j 2.3 & Java 1.7 & .Net 4.0

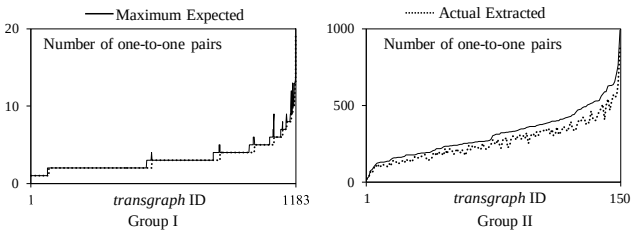

Figure 5: Distribution of number of maximum expected and actual extracted one-to-one pairs among ordered *transgraphs* in Group I & II.

group; we can observe an extraction with relatively high precision in almost every *transgraph* when we assume that recall one-to-one pairs are 100% accurate. Overall, however, 84.2% of maximum expected one-to-one pairs are extracted as the details shown in Table 5.

In order to evaluate the accuracy of the extracted one-to-one pairs, we randomly selected 3×100 samples from the sets of one-to-one pairs extracted from each group, receptively, and asked an ug-kk bilingual human to judge whether they are indeed correctly mapped as one-to-one. As a result, 237 (79%) out of 300 for Group I, and 251 (84%) out of 300 for Group II were determined to be correct. Thus, our method roughly yielded 70.5 % overall recall as shown in Table 5.

Nonetheless, it is not reasonable to directly compare these numbers with those in related works and reach a conclusion on the efficiency of our approach, since the experimental language pairs and resources chosen in each similar research are not quite the same. In response, we processed our 1183+150 *transgraphs* with the IC method, used as a baseline in related works (Shezaf and Rappoport, 2010), because it is a well-known approach to creating new dictionary from just two input dictionaries with no extra information.

IC examines the two pivot word sets: set of pivot translations of a word w^{A}, and the set of pivot translations of each w_i^{C} word that is a candidate for being translation to w^{A}. The more closely they match, the better the candidate is. Since the IC was not intended to create one-to-one mapping dictionaries, it allows multiple translations for a word, and translation sharing among multiple words. However, in our implementation of IC, we only leave a top ranked translation candidate w_i^{C} for each w_j^{A}, and if several candidates are equally ranked as top, we conduct random selection to reduce the possibility of translations to be shared with multiple words. As a result, output of IC method turned out to be overall 60% accurate, that is roughly 10.5% lower than the result of our proposal.

transgraphs	Maximum Expected	Actual Extracted	Accuracy	Recall
Group I	3877	3708 (95.6%)	79.0%	75.5%
Group II	47,893	39,907 (83.0%)	84.0%	70.0%
Overall	51,770	43,615 (84.2%)	83.7%	70.5%

Table 3: Overview of the experimental result

6. Conclusion

Bilingual dictionaries of many language pairs yet to be created. Such work has been challenging because many language pairs severely lack useful language resources like a parallel corpus, and even comparable corpora. To provide an efficient, robust and accurate dictionary creation method for poorly resourced language pairs, we presented a constraint approach to pivot-based dictionary induction, where a new dictionary of closely related language pair is induced from two exiting dictionaries using a distant language as a pivot. In our approach, the lexical intransitivity divergence is approached by modeling instance of induction as an optimization problem (WPMax-Sat is used for modeling), where the new dictionary is produced as a solution of the problem. We also considered data incompleteness to some extent. An experiment showed feasibility of our approach in practice. However, we note following points: (1) The problem may also be tackled by maximum weighted bipartite matching (Cheng et al., 1996) as well as Integer Linear Programming (ILP), which are left as our future work, as we will continue to explore more efficient modeling approaches and algorithms for dictionary induction; (2) There is a potential of including spelling as additional information; (3) More comparisons are expected to find whether the method can indeed rely purely on the inherent structure and still outperform the methods that utilize cheap external resources such as monolingual data; (4) The one-to-one assumption may be too strong for the general case, but we consider it is reasonable for the case of intra-family languages to reduce the complexity of the problem, while we continue to explore clearer evidence, and how to manage exceptions; (5) Applying the proposal to extra-family language pairs is also promising.

Acknowledgment

This research was partially supported by Service Science, Solutions and Foundation Integrated Research Program from JST RISTEX, and a Grant-in-Aid for Scientific Research (S) (24220002) from Japan Society for the Promotion of Science.

7. References

Ahn, Kisuh and Frampton, Matthew. (2006). Automatic generation of translation dictionaries using intermediary languages. In *Proceedings of the International Workshop on Cross-Language Knowledge Induction*, pages 41–44. Association for Computational Linguistics.

Biere, Armin, Heule, Marijn J. H., van Maaren, Hans, and Walsh, Toby, editors. (2009). *Handbook of Satisfiability*, volume 185 of *Frontiers in Artificial Intelligence and Applications*. IOS Press.

Bond, Francis and Ogura, Kentaro. (2008). Combining linguistic resources to create a machine-tractable japanese-malay dictionary. *Language Resources and Evaluation*, 42(2):127–136.

Bond, Francis, Yamazaki, Takefumi, Sulong, Ruhaida Binti, and Okura, Kentaro. (2001). Design and construction of a machine–tractable japanese-malay lexicon. In *ANNUAL MEETING OF THE ASSOCIATION FOR NATURAL LANGUAGE PROCESSING*, volume 7, page 1.

Cheng, Y, Wu, Victor, Collins, Robert, Hanson, A, and Riseman, E. (1996). Maximum-weight bipartite matching technique and its application in image feature matching. In *SPIE Conference on Visual Communication and Image Processing*, pages 1358–1379.

Dhillon, Inderjit, Guan, Yuqiang, and Kulis, Brian. (2005). A fast kernel-based multilevel algorithm for graph clustering. In *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 629–634. ACM.

Fu, Zhaohui and Malik, Sharad. (2006). On solving the partial max-sat problem. In *Theory and Applications of Satisfiability Testing-SAT 2006*, pages 252–265. Springer.

Fung, Pascale and Church, Kenneth Ward. (1994). K-vec: A new approach for aligning parallel texts. In *Proceedings of the 15th Conference on Computational Linguistics-Volume 2*, pages 1096–1102.

Haghighi, Aria, Liang, Percy, Berg-Kirkpatrick, Taylor, and Klein, Dan. (2008). Learning bilingual lexicons from monolingual corpora. *Proceedings of ACL-08: HLT*, pages 771–779.

Ishida, Toru. (2011). *The Language Grid*. Springer.

István, Varga and Shoichi, Yokoyama. (2009). Bilingual dictionary generation for low-resourced language pairs. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 2-Volume 2*, pages 862–870. Association for Computational Linguistics.

Kaji, Hiroyuki and Aizono, Toshiko. (1996). Extracting word correspondences from bilingual corpora based on word co-occurrences information. In *Proceedings of the 16th conference on Computational linguistics-Volume 1*, pages 23–28. Association for Computational Linguistics.

Kaji, Hiroyuki, Tamamura, Shin'ichi, and Erdenebat, Dashtseren. (2008). Automatic construction of a japanese-chinese dictionary via english. In *LREC*, volume 2008, pages 699–706.

Mairidan, Wushouer, Donghui, Lin, and Ishida, Toru. (2013). A heuristic framework for pivot-based bilingual dictionary induction. In *Proceedings of third International Conference on Culture and Computing*, September.

Matsuno, Jun and Ishida, Toru. (2011). Constraint optimization approach to context based word selection. In *Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Three*, pages 1846–1851. AAAI Press.

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

Nakov, Preslav and Ng, Hwee Tou. (2012). Improving statistical machine translation for a resource-poor language using related resource-rich languages. *Journal of Artificial Intelligence Research*, 44(1):179–222.

Saralegi, Xabier, Manterola, Iker, and Vicente, Iñaki San. (2011). Analyzing methods for improving precision of pivot based bilingual dictionaries. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 846–856. Association for Computational Linguistics.

Saralegi, Xabier, Manterola, Iker, and San Vicente, Iñaki. (2012). Building a basque-chinese dictionary by using english as pivot. In *LREC*, pages 1443–1447.

Schulz, Stefan, Markó, Kornél, Sbrissia, Eduardo, Nohama, Percy, and Hahn, Udo. (2004). Cognate mapping: A heuristic strategy for the semi-supervised acquisition of a spanish lexicon from a portuguese seed lexicon. In *Proceedings of the 20th international conference on Computational Linguistics*, page 813. Association for Computational Linguistics.

Shezaf, Daphna and Rappoport, Ari. (2010). Bilingual lexicon generation using non-aligned signatures. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 98–107. Association for Computational Linguistics.

Sjobergh, Jonas. (2005). Creating a free digital japanese-swedish lexicon. In *Proceedings of PACLING*, pages 296–300. Citeseer.

Skoumalova, Hana. (2001). Bridge dictionaries as bridges between languages. *International Journal of Corpus Linguistics, 6, Special Issue*, 95(105):11.

Soderland, Stephen, Etzioni, Oren, Weld, Daniel S, Skinner, Michael, Bilmes, Jeff, et al. (2009). Compiling a massive, multilingual dictionary via probabilistic inference. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1*, pages 262–270. Association for Computational Linguistics.

Tanaka, Kumiko and Iwasaki, Hideya. (1996). Extraction of lexical translations from non-aligned corpora. In *Proceedings of the 16th conference on Computational linguistics-Volume 2*, pages 580–585. Association for Computational Linguistics.

Tanaka, Kumiko and Umemura, Kyoji. (1994). Construction of a bilingual dictionary intermediated by a third language. In *Proceedings of the 15th Conference on Computational Linguistics - Volume 1*, COLING '94, pages 297–303, Stroudsburg, PA, USA. Association for Computational Linguistics.

Tanaka, Rie, Murakami, Yohei, and Ishida, Toru. (2009). Context-based approach for pivot translation services. In *IJCAI*, pages 1555–1561.

Tomaszczyk, Jerzy. (1986). The bilingual dictionary under review. In *Zurilex'86 Proceedings: Papers Read at the Euralex International Congress, University of Zurich*, pages 289–297.

Tufiş, Dan, Barbu, Ana Maria, and Ion, Radu. (2004). Extracting multilingual lexicons from parallel corpora. *Computers and the Humanities*, 38(2):163–189.

Vulić, Ivan and Moens, Marie-Francine. (2012). Detecting highly confident word translations from comparable corpora without any prior knowledge. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 449–459. Association for Computational Linguistics.

Yu, Kun and Tsujii, Junichi. (2009). Bilingual dictionary extraction from wikipedia. *Proceedings of Machine Translation Summit XII*, pages 379–386.