Reranking Translation Candidates Produced by Several Bilingual Word Similarity Sources

Laurent Jakubina
RALI/DIRO
Université de Montréal
Montréal, Québec, Canada
jakubinl@iro.umontreal.ca

Philippe Langlais
RALI/DIRO
Université de Montréal
Montréal, Québec, Canada
felipe@iro.umontreal.ca

Abstract

We investigate the reranking of the output of several distributional approaches on the Bilingual Lexicon Induction task. We show that reranking an n-best list produced by any of those approaches leads to very substantial improvements. We further demonstrate that combining several n-best lists by reranking is an effective way of further boosting performance.

1 Introduction

Identifying translations in bilingual material — the Bilingual Lexicon Induction (BLI) task — is a challenge that has long attracted the attention of many researchers. One of the earliest approach to BLI (Rapp, 1995) is based on the assumption that words that are translations of one another show similar co-occurrence patterns. Many variants have been investigated. For instance, some authors reported gains by considering syntactically motivated co-occurrences, either with the use of a parser (Yu and Tsujii, 2009) or by relying on simpler POS patterns (Otero, 2007). Extensions to multiword expressions have also been proposed (Daille and Morin, 2008). See (Sharoff et al., 2013) for an extensive overview.

Recently, vast efforts have been dedicated to identify translations thanks to so-called word embeddings. The seminal work of Mikolov et al. (2013b) shows that learning a mapping between word embeddings learnt monolingually by the popular `Word2Vec` toolkit (Mikolov et al., 2013a) is an efficient solution. Since then, many practitioners have studied the BLI task as a mean to evaluate continuous word-representations (Coulmance et al., 2015; Vulić and Moens, 2015; Luong et al., 2015; Gouws et al., 2015; Duong et al., 2016). Those approaches differ in the type of data they can process (monolingual data, word-aligned parallel data, parallel sentence pairs, comparable documents). Nevertheless, learning to map individually trained word embeddings remains an extremely efficient solution that performs well on several BLI benchmarks. Read (Upadhyay et al., 2016; Levy et al., 2017) for two recent comparisons of several of those techniques.

Reranking the output of several BLI approaches has been investigated, mostly for translating terms of the medical domain, where dedicated approaches can be designed to capture correspondences at the morphemic level (Delpech et al., 2012; Harastani et al., 2013; Kontonatsios et al., 2014). A similar idea (generating candidate translations, then filtering them by rescoring) has been proposed in (Baldwin and Tanaka, 2004) for translating noun-noun compounds in English and Japanese. Also, Irvine and Callison-Burch (2013) show that monolingual signals (orthographic, temporal, etc.) can be used to train a classifier to distinghish good translations from erroneous ones.

In this paper, we investigate the reranking of n-best lists of translations produced by two embedding approaches (Mikolov et al., 2013b; Faruqui and Dyer, 2014) as well as a plain distributional approach (Rapp, 1995). We tested a large number of variants of those approaches, for the English-to-French translation direction. The investigation of other language pairs and other BLI approaches is left as future work. To the best of our knowledge, this is the first time reranking embedding-based BLI approaches is reported.

We present our reranking framework in Section 2, our experimental protocol in Section 3, and report experiments in Section 4. We analyze our results in Section 5 and summarize our contributions in Section 6.

2 Reranking

The RankLib[1] library offers the implementation of 8 Learning to Rank Algorithms. We trained each one in a supervised way to optimize precision at rank 1. We used a 3-fold cross-validation procedure where in each fold, 700 terms of the test set were used for training, and the remaining 300 ones served as a test set. For a source term s and a candidate translation t, we compute 3 sets of straightforward and easily extensible features:

Frequency features Four features recording the frequency of s (resp. t) in the source (resp. target) corpus, the difference between those two frequencies as well as their ratio.

String features Five features recording the length (counted in chars) of s and t, their difference, their ratio, and the edit-distance between the two. Edit-distance has been consistently reported to be a useful hint for matching terms.

Rank features For each n-best list considered, we compute 2 features: t's score in the list, as well as its rank. Whenever several n-best lists are reranked, we also add a feature that records the number of n-best lists t appears in as a candidate translation of s.

3 Experimental Protocol

3.1 Data sets

We trained each word's representation on the English and French versions of the Wikipedia dumps from June 2013. The English vocabulary contains 7.3M words forms (1.2G tokens) while the French vocabulary contains 3.6M forms (330M tokens).

One research avenue we explored in this study consisted in assessing the impact of words' frequency on the BLI performance. For this, we gathered two reference lists of words and their translations. One list, named Wiki$_{\leq 25}$, is populated with English words occurring 25 times or less in Wikipedia (English edition). There are 6.8M (92%) such words. Thus, this test set is more representative of a real-life setting. The other list, named Wiki$_{>25}$ contains words whose frequencies exceed 25. Both lists contain 1 000 words that we randomly picked from an in-house bilingual lexicon. Each one of those words had to have at least one of its approved translations belong to the French Wikipedia vocabulary.

Most recent studies on BLI focus on translating very frequent words, in keeping with the protocol described in (Mikolov et al., 2013b), which basically consists in translating 1 000 terms from the WMT11 dataset. Those terms' rank are between 5000 and 6000 when the terms are sorted in decreasing order of frequency (the most frequent 5k words are put aside in order to train the projection). We reproduced this setting for comparison purposes (list Euro$_{5-6k}$). Only 87.3% of the resulting pairs have both their source term in the English Wikipedia vocabulary and their approved translation in the French counterpart. For the sake of fairness, we report results of the embedding-based approaches on those terms only.

The main characteristics of our test sets are presented in Table 1. As an illustration of the difficulty of each test set, we measure the accuracy (@1) of a baseline that ranks candidates in increasing order of edit-distance with the source term. For some reasons, the Wikipedia test sets are easier than Euro$_{5-6k}$ for such an approach,.

	Frequency				
	min	max	avg	Cov (%)	@1
Wiki$_{>25}$	27	19.4k	2.8k	100.0	19.3
Wiki$_{\leq 25}$	1	25	10	100.0	17.6
Euro$_{5-6k}$	1	2.6M	33.6k	87.3	8.0

Table 1: Characteristics of our test sets. $Cov.$ is the percentage of source terms for which the reference translation is part of the French edition of Wikipedia.

3.2 Metrics

Each approach (see Section 4) has been configured to produce a ranked list of (at most) 100 candidate translations (in French). We measure their performance with accuracy at rank 1, 5, and 20; where accuracy at rank i (@i) is computed as the percentage of test words for which a reference translation is identified in the first i candidates proposed.

4 Experiments

4.1 Individual Approaches

We ran variants of an in-house implementation of (Rapp, 1995) exploring a number of meta-

[1] https://sourceforge.net/p/lemur/wiki/RankLib/

	INDIVIDUAL			1-RERANKED			n-RERANKED			
	@1	@5	@20	@1	@5	@20		@1	@5	@20
Wiki$_{>25}$										oracle: 69.3
Rapp	20.0	33.0	43.0	36.3$^{2.5}$	48.8$^{1.9}$	53.8$^{1.9}$	base	34.3$^{1.9}$	47.6$^{1.4}$	58.8$^{0.8}$
Miko	17.0	32.6	41.6	38.1$^{1.9}$	49.0$^{1.5}$	54.3$^{1.3}$	R+M	43.3$^{2.9}$	58.4$^{1.4}$	62.4$^{3.1}$
Faru	13.3	26.0	33.3	34.3$^{1.5}$	44.0$^{2.6}$	47.9$^{2.1}$	R+M+F	**45.6**$^{2.2}$	**59.6**$^{1.1}$	**64.0**$^{1.8}$
Wiki$_{\leq 25}$										oracle: 28.6
Rapp	2.6	4.3	7.3	8.6$^{1.2}$	9.4$^{0.8}$	10.2$^{1.0}$	base	10.7$^{0.6}$	15.9$^{1.2}$	21.8$^{0.7}$
Miko	1.6	4.6	10.6	16.6$^{2.2}$	19.0$^{1.5}$	20.1$^{1.4}$	R+M	18.9$^{2.01}$	22.0$^{1.3}$	23.6$^{2.2}$
Faru	1.6	2.6	5.0	7.9$^{2.2}$	8.7$^{2.5}$	8.9$^{2.7}$	R+M+F	**21.3**$^{1.86}$	**24.4**$^{1.7}$	**25.7**$^{1.9}$
Euro$_{5-6k}$										oracle: 84.4
Rapp	16.6	31.8	41.2	34.6$^{5.7}$	48.6$^{1.2}$	51.9$^{1.2}$	base	33.6$^{1.2}$	59.3$^{1.4}$	71.7$^{2.5}$
Miko	42.0	59.0	67.8	47.0$^{2.3}$	68.1$^{2.7}$	73.0$^{1.7}$	R+M	**49.5**$^{3.7}$	**68.7**$^{1.5}$	76.1$^{1.0}$
Faru	30.6	47.7	59.8	41.2$^{3.9}$	58.0$^{3.5}$	66.0$^{3.5}$	R+M+F	47.6$^{2.3}$	68.5$^{2.0}$	**76.2**$^{1.2}$

Table 2: Performance of each approach (left-hand side column) and their reranking (middle column), as well as the best reranking of 2 and 3 native n-best lists (right-hand side column). The reranked results are averaged over a 3-fold cross-validation procedure, the superscript indicates the standard deviation. `oracle` picks the reference translation among the 3 individual n-best lists.

parameters (window size, association measure, seed lexicon, etc.). We refer to this approach as `Rapp` hereafter. We studied a similar number of variants of (Mikolov et al., 2013b) — hereafter named `Miko` — training monolingual embeddings with `Word2Vec` (Mikolov et al., 2013b), varying among other things the model's architecture (skip-gram versus continuous bag-of-words), the optimization algorithm (negative sampling (5 or 10 samples) versus hierarchical softmax), and the context window size (6, 10, 20, 30). The largest embedding dimension for which we managed to train a model is 200 for the *cbow* architecture, and 250 for the *skg* architecture. We learnt the projection matrix with the implementation described in (Dinu and Baroni, 2015). We reproduced the approach of Faruqui and Dyer (2014) — henceforth `Faru` — thanks to the toolkit provided by the authors. We kept the embeddings that yielded the best performance for the `Miko` approach, and ran several configurations, varying the bilingual lexicon used, and tuning the $ratio$ parameter over the values 0.5, 0.8 and 1.0.

The best performance for the variants of each strategy we tested is reported in the first column of Table 2. On Wiki$_{>25}$, the `Rapp` approach delivers the best performance at rank 1, slightly outperforming the edit-distance baseline (@1 of 19.3). The drop in performance of all approaches on Wiki$_{\leq 25}$ is striking: the best one could only identify the translation of 2.6% of the test terms at rank 1. This clearly demonstrates the bias of the approaches tested in favor of frequent words. On the Euro$_{5-6k}$ test set, the two embedding approaches are rather good (@1 of `Miko` reaches 42%) and clearly outperform `Rapp`. This suggests that embeddings are very apt at capturing information for very frequent terms (test terms on Euro$_{5-6k}$ appear roughly 10 times more in Wikipedia than those in Wiki$_{>25}$). Our results are in line with those reported in (Mikolov et al., 2013b). We were more surprised by the lower performance yielded by `Faru`. It should be noted however that this model's gains, as reported in (Faruqui and Dyer, 2014), have been measured on monolingual tasks. The authors also built on top of embeddings learnt with the *skg* architecture, while we found it to be less accurate for our task.

4.2 Reranking Individual Approaches

The middle column in Table 2 reports the reranking of the n-best list produced by each individual approach. During calibration experiments, we found better rescoring performances with the *Random Forest* algorithm. We report results for this algorithm only.[2] We observe that reranking is

[2]Results were close with *LambaMart* (2 @1 points lost) and *Mart* (1.5 @1 points lost).

Wiki$_{>25}$	*Sing.*	*Cumulative*		Wiki$_{\leq 25}$	*Sing.*	*Cumulative*		Euro$_{5-6k}$	*Sing.*	*Cumulative*	
feat.	@1	@1	@100	feat	@1	@1	@100	feat	@1	@1	@100
Rank	33.0	33.0	66.0	String	16.6	16.6	26.6	Rank	46.2	46.2	81.3
+String	32.0	42.0	67.0	+Rank	6.6	20.3	26.3	+String	18.9	43.9	80.3
+ Freq	0.3	43.0	67.3	+Freq	0.0	20.3	26.6	+Freq	2.2	48.8	82.5

Table 3: Influence of the features used to train the reranker when combining `Rapp`, `Miko`, and `Faru`. Performances are averaged over a 3-fold cross-validation procedure. Each fold uses 700 pairs for training and 300 for testing. *Sing.* indicates the performance of individual features, while *Cumulative* indicates their cumulative performance. Features are listed in decreasing order of gains.

highly beneficial to each approach. For instance, when reranking the n-best list produced by `Miko`, @1 nearly doubles on Wiki$_{>25}$, and is 10 times higher on Wiki$_{\leq 25}$. It is also noteworthy that on Wiki$_{>25}$ all approaches, once reranked perform equally overall (@1 between 34 and 38) — `Miko` enjoying a slight advantage here — far better than the edit distance baseline.

4.3 Combining by Reranking

We conducted experiments aiming at combining several n-best lists with reranking. For comparison purposes, we implemented a naive combination approach that ranks a candidate translation higher if it is proposed in more n-best lists. Tied candidates are further sorted in increasing order of edit distance. The results of a few combinations are reported in the right column of Table 2.

Combining the n-best lists produced by the 3 native approaches leads to the best performance overall, except on Euro$_{5-6k}$ where not considering `Faru` leads to slight improvements in @1 and @5 metrics. This indicates that the reranker puts good use of multiple models. The gains over each reranked approach are impressive on Wiki$_{>25}$ (increase from 38.1% to 45.6%) and Wiki$_{\leq 25}$ (increase fom 16.6 to 21.3) and minor on Euro$_{5-6k}$(from 47.0% to 47.6%). We also observe that @20 obtained by the reranker is not very far from the oracle performance.

5 Analysis

In this section, we analyze the characteristics of the reranker we used to combine the 3 aforementioned approaches.

5.1 Training Size

Figure 1 shows the impact of the quantity of material used for learning the reranker, varying from 100 word pairs to 700. In this experiment, we always use the same 300 test words per test set. Increasing the training material increases performance for all test sets,[3] but even a small training set is enough to improve upon native approaches. In particular, using 200 training instances already yields a @1 of 36.6 on Wiki$_{>25}$, while the best native tops at 20.

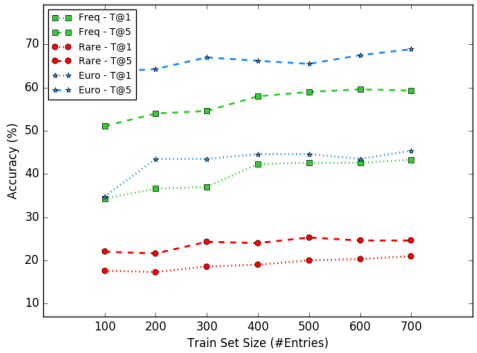

Figure 1: Influence of the training size (number of examples) on the performance of the reranker on Wiki$_{>25}$, Wiki$_{\leq 25}$ and Euro$_{5-6k}$.

5.2 Feature Selection

Table 3 shows the influence of the features used for training the reranker. On frequent terms (Wiki$_{>25}$ and Euro$_{5-6k}$) the rank-based features are the most useful ones, followed by the string-based features. The frequency-based features only help marginally. On Wiki$_{\leq 25}$, the string-based features are more useful. The performance of the reranker using only those features (16.6@1) is close to that of the baseline edit distance approach (17.6@1). Adding the rank-based features increases the performance slightly (20.3@1).

[3] On Wiki$_{\leq 25}$ however, the gains are very small.

5.3 Ranker Analysis

With a few exceptions, we observe that whenever at least 2 native approaches propose the reference translation first, the reranker keeps at the first position as well. When only one native approach is accurate at position 1, the results differ from one test set to another. It is only occasionally that the reranker will prefer the reference translation when none of the native approaches does. On Wiki$_{>25}$, this happens 130 times out of 300 cases, but on Euro$_{5-6k}$, it happened only 4 times over 132 cases, which is disappointing. Still, the average position of the reference translation in the reranker's output is clearly improving for all test sets, as shown in Table 4. The average number of positions gained by reranking is rather high, and outdoes an oracle that picks the n-best list in which the reference translation is best positioned. We note that the average rank of the `Rapp` approach is lower than that of the embedding approaches, for both Wikipedia test sets.

	Wiki$_{>25}$	Wiki$_{\leq 25}$	Euro$_{5-6k}$
`Rapp`	12.7	19.6	16.2
`Miko`	16.3	30.0	7.5
`Faru`	20.4	35.5	11.3
list-oracle	12.3	9.1	7.1
reranker	5.6	4.0	4.9

Table 4: Average rank of the reference translation. Terms for which the reference translation is not found in the first 100 positions are discarded.

5.4 Error Analysis

We manually inspected the first candidate produced by our best reranker (the one combining the 3 native approaches) for the first 100 test forms for which the candidate translation differs from the reference one. We encountered the following representative cases: morphological variants of the reference translation (*e.g. trompeur* / *trompeuse*, litt. *misleading*) — MORPHO; directly related translation, such as synonyms, antonyms, and cohyponyms — RELATED; loosely related to the reference (*e.g. gunman* / *poignardé*, litt. *stabbed*) — LOOSLY; English words – ENGLISH; translations that apparently have nothing to do with the source term (*e.g. judged /méritant*, litt. *worthy*) – JUNK; and translations that correspond to another sense of a polysemic term (*e.g. grizzly* / *grizzli*, while the reference translation is *grisonnant*, litt. *gray haired*) – POLYSEMY. The counts of each class for each test set are reported in Table 5.

We observe that the percentage of JUNK errors is much higher on Wiki$_{\leq 25}$, yet another illustration of the bias the approaches we tested have in favor of frequent terms. If we consider synonyms, morphological variants as well as polysemic cases to be correct, then the percentages of test forms that are redeemed reach 37% for Wiki$_{>25}$ and 50% for Euro$_{5-6k}$ of test forms that were counted wrong are indeed acceptable translations. On Wiki$_{\leq 25}$ however, this percentage is much lower (4%).

	Wiki$_{>25}$	Wiki$_{\leq 25}$	Euro$_{5-6k}$
MORPHO	18	3	26
RELATED	16	4	23
synonyms	*15*	*1*	*19*
antonyms	*1*	*2*	*2*
hyponym			*1*
cohyponym		*1*	*1*
POLYSEMY	4	0	5
LOOSLY	14	15	20
ENGLISH	21	6	7
JUNK	27	72	19

Table 5: Annotation of 100 translations produced (at rank 1) for each test set by the reranked output of the 3 native approaches.

6 Discussion

We have studied the reranking of three approaches to BLI. We reported significant improvements for all approaches, on all test sets. We also show that combining several n-best lists by reranking is a simple yet effective solution leading to even better performance. The gains were obtained by a random forest model learnt on a set of straightforward features, which leaves ample room for better feature engineering. While extra data must be used to train the reranker, we show that as few as 200 training examples often suffice to provide an appreciable boost in performance. As a future work we want to investigate whether similar gains can be obtained for other language pairs.

Acknowledgments

This work as been partly funded by the NSERC TRiBE grant.

References

Timothy Baldwin and Takaaki Tanaka. 2004. Translation by machine of complex nominals: Getting it right. In Takaaki Tanaka, Aline Villavicencio, Francis Bond, and Anna Korhonen, editors, *Second ACL Workshop on Multiword Expressions: Integrating Processing*, pages 24–31, Barcelona, Spain, July. Association for Computational Linguistics.

Jocelyn Coulmance, Jean-Marc Marty, Guillaume Wenzek, and Amine Benhalloum. 2015. Transgram, fast cross-lingual word-embeddings. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1109–1113, Lisbon, Portugal, September. Association for Computational Linguistics.

Béatrice Daille and Emmanuel Morin. 2008. An effective compositional model for lexical alignment. In *Proceedings of the Third International Joint Conference on Natural Language Processing: Volume-I*, pages 95–102, Hyderabad, India, January. Association for Computational Linguistics.

Estelle Delpech, Béatrice Daille, Emmanuel Morin, and Claire Lemaire. 2012. Extraction of domain-specific bilingual lexicon from comparable corpora: Compositional translation and ranking. In *Proceedings of COLING 2012*, pages 745–762, Mumbai, India, December. The COLING 2012 Organizing Committee.

Georgiana Dinu and Marco Baroni. 2015. Improving zero-shot learning by mitigating the hubness problem. *In ICLR 2015 Workshop Papers*, May.

Long Duong, Hiroshi Kanayama, Tengfei Ma, Steven Bird, and Trevor Cohn. 2016. Learning crosslingual word embeddings without bilingual corpora. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1285–1295, Austin, Texas, November. Association for Computational Linguistics.

Manaal Faruqui and Chris Dyer. 2014. Improving vector space word representations using multilingual correlation. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 462–471, Gothenburg, Sweden, April. Association for Computational Linguistics.

Stephan Gouws, Yoshua Bengio, and Greg Corrado. 2015. Bilbowa: Fast bilingual distributed representations without word alignments. In *Proceedings of the 32nd International Conference on Machine Learning (ICML-15)*, pages 748–756, Lille, France, July. JMLR.

Rima Harastani, Béatrice Daille, and Emmanuel Morin. 2013. Ranking translation candidates acquired from comparable corpora. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 401–409, Nagoya, Japan, October. Asian Federation of Natural Language Processing.

Ann Irvine and Chris Callison-Burch. 2013. Supervised bilingual lexicon induction with multiple monolingual signals. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 518–523, Atlanta, Georgia, June. Association for Computational Linguistics.

Georgios Kontonatsios, Ioannis Korkontzelos, Jun'ichi Tsujii, and Sophia Ananiadou. 2014. Combining string and context similarity for bilingual term alignment from comparable corpora. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1701–1712, Doha, Qatar, October. Association for Computational Linguistics.

Omer Levy, Anders Søgaard, and Yoav Goldberg. 2017. A strong baseline for learning cross-lingual word embeddings from sentence alignments. *arXiv preprint arXiv:1608.05426*.

Thang Luong, Hieu Pham, and D. Christopher Manning. 2015. Bilingual word representations with monolingual quality in mind. In *Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing*, pages 151–159, Denver, Colorado. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. *In ICLR 2013 Workshop Papers*, May.

Tomas Mikolov, Quoc V. Le, and Ilya Sutskever. 2013b. Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.

Pablo Gamallo Otero. 2007. Learning bilingual lexicons from comparable english and spanish corpora. In *Proceedings of Machine Translation Summit XI*, pages 191–198, Copenhagen, Denmark, September. European Association of Machine Translation.

Reinhard Rapp. 1995. Identifying word translations in non-parallel texts. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*, pages 320–322, Cambridge, Massachusetts, USA, June. Association for Computational Linguistics.

Serge Sharoff, Reinhard Rapp, and Pierre Zweigenbaum, 2013. *Overviewing Important Aspects of the Last Twenty Years of Research in Comparable Corpora*, pages 1–17. Springer Berlin Heidelberg, Berlin, Heidelberg.

Shyam Upadhyay, Manaal Faruqui, Chris Dyer, and Dan Roth. 2016. Cross-lingual models of word embeddings: An empirical comparison. In *Proceedings of the 54th Annual Meeting of the Association*

for Computational Linguistics (Volume 1: Long Papers), pages 1661–1670, Berlin, Germany, August. Association for Computational Linguistics.

Ivan Vulić and Marie-Francine Moens. 2015. Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 719–725, Beijing, China. Association for Computational Linguistics.

Kun Yu and Jun'ichi Tsujii. 2009. Extracting bilingual dictionary from comparable corpora with dependency heterogeneity. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers*, pages 121–124, Boulder, Colorado, June. Association for Computational Linguistics.