Reranking Candidate Lists for Improved Lexical Induction

Laurent Jakubina and Philippe Langlais

Université de Montréal
CP 6128 Succursale Centre-Ville
H3C3J7 Montreal, Québec, Canada
rali.iro.umontreal.ca

Abstract. Identifying translations in bilingual material — also referred to as the Bilingual Lexicon Induction (BLI) task — is a challenge that has attracted many researchers since a long time. In this paper, we investigate the reranking of two types of state-of-the-art approaches that have been used for the task. We test our reranker on four language pairs (translating from English), analyzing the influence of the frequency of the source terms we seek to translate. Our reranking approach almost invariably leads to performance gains for all the translation directions we consider.

1 Introduction

Identifying translations in bilingual material — also referred to as the Bilingual Lexicon Induction (BLI) task — is a challenge that has attracted many researchers since a long time. One of the earliest approaches [1] relies on the assumption that words in translation relation show similar co-occurrence patterns. Many variants of this approach have been investigated [2]. Some authors have for instance reported gains by considering syntactically motivated co-occurrences, either by the use of a parser [3] or simpler Part of Speech patterns [4]. Extensions to multiword expressions have also been proposed [5].

Recently, there has been a wealth of works dedicated to identify translations thanks to so-called word embeddings. In [6] the authors describe two models implemented in the popular `Word2Vec` toolkit, that efficiently train vector-representations of the words in a large monolingual collection of texts. In [7], it is further shown that a mapping between word embeddings learnt independently for each language can be trained by making use of a (small) seed bilingual lexicon. Since then, many practitioners have studied the BLI task as a mean to evaluate continuous word-representations [8–12]. While approaches differ in the type of data they can digest (monolingual data, word-aligned parallel sentence pairs, parallel sentence pairs) it is still fair to say that training monolingual word embeddings, then learning a mapping between the two vector spaces obtained is an extremely efficient solution that performs rather well on several BLI benchmarks. Read [13, 14] for two comparisons of several of those techniques.

Learning to discriminate good from bad translations has been investigated in [15]. In this work, the authors show that monolingual signals (orthographic,

temporal, topical, etc.) can be efficiently used to train in a supervised way a binary classifier. For this approach to work, some metadata is required, such as the time stamp of (pre-collected) news texts, or the list of inter-language linked Wikipedia pages.[1] We are more interested in this work to approaches that do not exploit such metadata.

Reranking the output of several BLI approaches has been investigated by some authors [16–18], mostly for translating terms of the medical domain, where dedicated approaches can be designed to capture correspondences at the morphemic level.[2] A similar idea has been proposed in [19] for translating noun-noun compounds in English and Japanese. In those works, the rerankers are prescribed (basically, a weighted average of native scores), while in [20], a reranker is trained in a supervised way to rerank or merge n-best lists. While interesting, this last approach has only been tested on the English-French language pair.

In this study, we revisit and extend the work of [20] by considering three other language pairs: English-Romanian, English-Spanish, and English-German. We provide evidences that the reranking approach is apt for all these language pairs. We distribute our datasets and bilingual lexicons, so that further experiments can be performed.[3]

2 Native BLI Methods

We consider three native approaches to BLI, all of them being projective in the sense that they all map mono-lingually acquired representations (trained or not), thanks to a (small) seed bilingual lexicon.

2.1 Plain Projective Approach (`Rapp`)

We tested variants of [1] where each word of interest is represented mono-lingually by a so-called context vector, that is, the words it co-occurs with. A co-ocurrence can be encoded by a boolean (present/absent), by its frequency, or by a real measuring its association strength: point-wise mutual information (PMI), log-likelihood ratios (LLR) and discontinuous odd-ratios (ODR) are popular weighting schemas. Given a word to translate, its vector representation is projected using a seed bilingual lexicon: each co-ocurrent word is simply looked-up in the bilingual dictionary, and the sanctioned translations are added to the vector representation in the target language. A similarity measure (typically cosine) is used for ranking target vectors according to their similarity to the projected vector.

We ran more than 110 configurations, varying typical hyper-parameters among which the context window size (6, 15, 20, 30), the association measure (PMI, LLR, ORD), as well as meta-parameters that control the way the target vectors

[1] The authors demonstrate the potential of the approach on 22 language pairs, but the datasets used in their work are unfortunately not available.

[2] Similarly to the orthographic signal used in [15].

[3] `http://rali.iro.umontreal.ca/rali/en/bli-dataset`

are computed. We used the cosine similarity measure in all configurations, and projected source vectors according to a large in-house bilingual lexicon with no overlap with the test material.

Since many contextual words are typically absent from this lexicon, the resulting projected vector can be rather sparse. To overcome this situation to some extent, words in the context vector unknown for the seed bilingual lexicon are added to the projected vector. We observe this strategy to be beneficial in practice since unknown words encompass proper names, numerical entities or acronyms that often are invariant across languages.

2.2 Linear Projection (`Miko`)

In [7], the authors propose to train a linear transformation between independently trained source and target embeddings, thanks to a seed lexicon. We reproduced variants of this approach, training monolingual embeddings with `Word2Vec`[4] toolkit [7]. We ran over 160 configurations, varying the architecture of the model (Skip-gram (*skg*) or Continuous Bag-of-Words (*cbow*)), the optimization algorithm (Negative Sampling (5 or 10 samples) or Hierarchical Softmax), 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.[5] We learnt the projection matrix with the implementation described in [21]. We built 6 bilingual lexicons of different characteristics: 5 from our in-house bilingual lexicon, that is, 2k of low frequency word pairs, 5k, 15k and 30k of randomly picked word pairs, and 5k of highly frequent word pairs; as well as one lexicon populated with 5k of highly frequent word pairs, as in [7].

2.3 Canonical Correlation Analysis (`Faru`)

In [22], the authors propose a technique based on canonical correlation analysis which transforms two existing monolingual embeddings so as to maximize the correlation between representations of words paired in a bilingual lexicon. It has been shown to improve independently trained monolingual embeddings on a number of monolingual benchmarks.

This approach only changes the way monolingual embeddings are mapped, so we used the embeddings we identified the most suited for the `Miko` approach. We used the implementation accompanying [22].[6] We ran a number of configurations, varying the bilingual lexicon used (as previously described), and tuning the *ratio* parameter over the values 0.5, 0.8 and 1.0.

[4] `https://code.google.com/p/word2vec/`
[5] Our cluster could accommodate up to 64 Gb of memory.
[6] `https://github.com/mfaruqui/crosslingual-cca`

3 Reranking

We used the RankLib[7] library to access the implementation of 8 Learning to Rank Algorithms (MART, RankNet, RankBoost, AdaRank, Coordinate Ascent, LambdaMART, ListNet, and Random Forests). We optimized each algorithm in a supervised way thanks to precision at rank 1. For a source term s and a candidate translation t, we compute 3 families of features:

Frequency features 4 features recording the (raw) frequency of s (resp. t) in the source (resp. target) corpus, the difference between those two frequencies as well as their ratio. Frequency is one of the signals used in [15].

String features 5 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 repetitively reported to be a useful clue for matching terms. It has been also used as a useful signal in [15].

Rank features for each n-best list considered, we compute 2 features: the score of t in the list, as well as its rank. Whenever several n-best lists are reranked, we also add a feature which records the number of n-best lists in which t appears as a candidate translation of s.

Those features are straightforward, do not require any metadata, and are of course largely extensible. But as we shall see, they already yield gains, for the four language pairs we studied here.

4 Experimental Protocol

4.1 Monolingual Collections

We consider the task of identifying the translation of English terms into 4 languages: French, German, Romanian and Spanish. We extracted the text from those dumps thanks to `WikiExtractor`[8]. The monolingual datasets used for computing distributional representations are Wikipedia dumps, which means that a fair number of article pairs are indeed comparable[9], although we do not use this information specifically. The main characteristics of the Wikipedia dumps we collected for each language are reported in Table 1. One specificity of our experimental protocol, is that we seek the translation of a source term among all the token types present in the target Wikipedia collection. As can be seen in the last line of Table 1, this represents a rather large set of candidate translations (in the order of millions), which poses technical challenges. This choice departs from most studies where heuristics are being used to reduce the list of candidate terms among which a translation is searched for. A typical experimental setting consists in considering only target words that happen frequently enough (say at

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

[8] `https://github.com/attardi/wikiextractor`

[9] Two documents are said comparable if they address the same topic, without being in translation of each other.

	English	German	French	Spanish	Romanian
#tokens	4 075.7M	1425.6M	1220.3M	866.5M	130.8M
#types	8.6M	8.0M	3.7M	3.2M	1.2M

Table 1. Main characteristics of the Wikipedia material. The French dump is from June 2013, while for the other languages we took the dumps of July 2017.

least five times) in the target collection, as in [7] where the size of the target vocabulary they consider is in the order of a hundred thousand words, an order of magnitude less than in our setting. We believe or choice to be more faithful to the zipfian nature of (even massive) texts collections, where most token types actually occur rarely.

4.2 Test Sets

Following [20], we gathered for each language pair three reference lists of words and their translations, each one populated with source (English) terms of various frequency in Wikipedia. One named $\text{Wiki}_{\leq 25}$, is populated with English words occurring less than 26 times in English Wikipedia. Actually 92% of the tokens in the English collection have this property. Another set, named $\text{Wiki}_{>25}$, gathers words with frequency in (English) Wikipedia higher than 25. Last, since most recent studies on BLI focus on translating highly frequent words, we reproduced this setting here, following the protocol described in [7]. Basically, it consists in translating 1 000 terms of the WMT11 dataset[10], which rank between 5000 and 6000 when sorted in decreasing order of frequency (the first 5k top-frequent words being spared to train the projection). We name the resulting dataset Euro_{5-6k} hereafter.

For the English-French language pair, and for each test set, we randomly picked 1000 English words among those belonging to an in-house general bilingual lexicon, and for which one of their sanctioned translations belongs to the French Wikipedia vocabulary. This way, we eliminate the numerous proper names present in our lists (especially for low-frequency words): translating proper names may involve transliteration [23], an interesting problem which is not the focus of this study.

For the other language pairs, we did not have such a general bilingual lexicon. Therefore, we followed [7] and resorted to `Google Translate`[11] to translate the English words of our test sets. We removed pairs where the does not belong to the target Wikipedia vocabulary. Often, this application produces a translation identical to the source word. While it might happen that a word and its translation are identical in a given language pair, most often, this is a mistake of the application. We are not aware of any study that paid attention to this

[10] `www.statsmt.org`

[11] `https://translate.google.com/`

phenomenon. We found it problematic enough in our case[12] that we decided to remove from our test sets all the pairs involving a translation identical to its source word. Further, it happens that `Google Translate` does not translate a word, or that the translation produced is not part of the Wikipedia vocabulary. We removed those entries from our test sets. The main characteristics of our datasets are presented in Table 2.

	German			Spanish			Romanian			French
	ϕ	=	T/*test*	ϕ	=	T/*test*	ϕ	=	T/*test*	T/*test*
$\text{Wiki}_{>25}$	79	208	500/213	272	26	500/202	201	148	500/151	700/300
$\text{Wiki}_{\leq 25}$	276	317	300/107	385	149	300/166	131	484	300/85	700/300
Euro_{5-6k}	239	78	500/183	299	28	500/173	319	57	500/124	700/300

Table 2. Number of pairs of words per target langage considered. ϕ stands for the number of English words we could not translate with `Google Translate` or for which the translation was not part of the target Wikipedia; = indicates the number of pairs where the translation was identical to the source term; and T/*test* stands for the split of the remaining pairs into TRAIN and TEST datasets.

4.3 Reranking Protocol

For reranking experiments, we kept a number of entries of our test sets for training the classifier, and the remaining ones for testing. Based on the size of our datasets, we kept 700 entries for French, 500 for German and Spanish, and 300 for Romanian. Because this represents small quantities of material, we resorted to a 3-fold cross-validation procedure and report the average performance over the 3 folds. The results across folds are actually fairly stable.

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

5 Experiments

5.1 Calibration

For the English-French language pair, we compared hundreds of variants and came to the conclusion that there is no free lunch: for better performance, each approach should be adjusted to the specificity of the test words. For instance, for the `Rapp` approach, the variant performing the best on $\text{Wiki}_{>25}$ is one with a

[12] For instance, 32% (resp. 50.8%) of the rare words we submitted to Google Translate for the German (resp. Romanian) test set received a translation identical to the source term.

window size of 6 (3 words before and after), and using PMI, while on $\text{Wiki}_{\leq 25}$, the best configuration is obtained by considering a window size of 30 and the discontinuous odd-ratio association measure. Similarly, for the `Miko` approach, the best configuration on frequent words ($\text{Wiki}_{>25}$ and Euro_{5-6k}) is obtained by training embeddings with the *cbow* architecture, *negative sampling* (10 samples), and a window size of 10. For learning the translation matrix, we found the best results with a bilingual lexicon of size 5k, containing frequent words[13], as suggested in [7]. For $\text{Wiki}_{\leq 25}$, however, a *skip-gram* architecture with a *hierarchical softmax*, and a window size of 20 yields the best performance.

For the French-English language pair, the best performing configuration for each test set has been selected in the figures reported in this study. While this overestimates the performance on this language pair, our focus in this study is on reranking. Therefore, we assume that native approaches have been tuned specifically. For the other language pairs however, we kept the best configurations identified on the English-French language pair, which might not be optimal. This provides some data points on what happens when fine tuning is not performed.

	German		**French**		**Spanish**		**Romanian**	
	native	rerank	native	rerank	native	rerank	native	rerank
Euro_{5-6k}								
`Rapp`	0.2	0.2	16.6	**34.6**	0.9	**3.1**	0.9	**2.5**
`Miko`	38.1	**39.0**	42.0	**47.0**	36.8	**37.0**	24.6	**26.9**
`Faru`	5.2	**21.6**	30.6	**41.2**	6.5	**22.9**	0.1	**3.2**
$\text{Wiki}_{>25}$								
`Rapp`	3.6	1.8	20.0	**36.3**	4.3	**8.0**	4.1	**5.1**
`Miko`	19.1	**20.2**	17.0	**38.1**	18.1	**21.5**	5.4	**9.5**
`Faru`	4.0	**14.6**	13.3	**34.3**	3.1	**15.5**	0.2	**3.2**
$\text{Wiki}_{\leq 25}$								
`Rapp`	1.0	0.4	2.6	**8.6**	1.0	**1.5**	0.2	**0.5**
`Miko`	0.2	**1.2**	1.6	**16.6**	0.4	**3.1**	0.0	0.0
`Faru`	0.7	**4.6**	1.6	**5.0**	0.9	**9.1**	0.0	**2.5**

Table 3. @1 of the native approaches and their (individual) reranking.

5.2 Native approaches and their Reranking

Table 3 reports the results of the native approaches, as well as their individual reranking. This table calls for several comments. A first thing to note is the

[13] For Euro_{5-6k}, we took the top 5k frequent words of the Europarl corpus, while for $\text{Wiki}_{>25}$, we took the 5k most frequent words of Wikipedia intersected with our in-house lexicon. Using a single lexicon for both test sets markedly decreases performance.

overall good performance of `Miko` on the Euro_{5-6k} test set. Variations do occur with the target language, Romanian being the most difficult to deal with. One explanation for this is the relatively small size of the Wikipedia Romanian collection, where many target words are seen only a few times, which puzzles the approach somehow. Still, the precision at rank 1 of `Miko` is 24.6%, while the two other approaches culminate at less than 1%! Another observation is the overall bad performance of the `Rapp` approach. Only on the $\text{Wiki}_{>25}$ corpus, for the English-French translation direction, does it outperform `Miko` by 3 absolute @1 points. Similarly, the results of `Faru` are disappointing. This contradicts the observation made by their authors who reported better results. This might be explained by the different nature of the tasks they tested.

A second striking observation is the very disappointing results of all the approaches on the $\text{Wiki}_{\leq 25}$ test set, where for as less as 1% of the test words, could a translation be identified at rank 1. Enlarging the candidate list raises this figure slightly[14], but not to a satisfactory level. This clearly indicates that seeking the translation of unfrequent words in a large collection of texts is by far an unsolved problem.

A third, and more positive, observation is the overall stable performance of the reranking approach. Gains are not spectacular, but (to a few exceptions) consistent across test sets, language pairs, and approaches. Considering the very light features set we considered, this somehow comes to a surprise, and calls for more feature engineering. The gains observed while reranking the n-best list produced by the `Faru` approach are actually rather impressive. For instance, reranking increases @1 from 6.5% to 22.9% for the English-Spanish language pair.

Figure 1 shows the average rank of the reference translation before and after reranking, for all the approaches and data sets for both the English-German and English-Spanish translation directions (similar patterns are observed for the English-Romanian language pair). Terms for which the reference translation was not in the native 100-best list were not considered. It is clear that the average rank of the reference translation does decrease significantly after reranking, often drastically. There is one notable exception for German, with the `Rapp` approach on the Euro_{5-6k} test set, where the rank increased from 9 to 44 after reranking. The reason why it is so still needs to be investigated[15]. This data point apart, we were rather astonished of the gains in ran k obtained by our simple reranker. Further analysis is provided in Section 5.4.

5.3 Combining Native n-best Lists by Reranking

Despite the stable gains we obtained when reranking individual n-best lists, there is no reason not to rerank all the candidate lists produced for a given test word. This is what we investigate hereafter. Table 4 reports the results of the best native approach per test set (the one recording the best @1), its individual

[14] The highest @20 (10.4%) is recorded by `Faru` when translating into Spanish.
[15] Will be done for the final version.

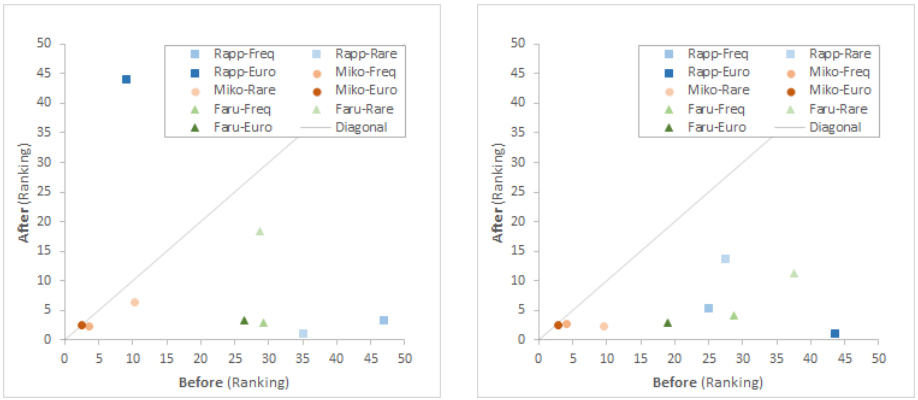

Fig. 1. Average rank of the reference translation before and after reranking, for each approach and test set for the English-German translation task (left) and the English-Spanish task (right). Average are computed on the only terms for which the reference translation was proposed in the top 100 candidate list of the native approach.

reranking, as well as the reranking of the candidates produced by the three approaches (`R+M+F`). The latter case leads overall to the best performance, with one notable exception for the English-German language pair, where reranking the three candidate lists leads to a slight loss in @1 (37.87% versus 38.1%). Most of the time, reranking the output of the three approaches is preferable to reranking the best approach only. This demonstrates the complementarity of the approaches, and the efficiency of the simple rank features we exploited (see Section 3). The most impressive gains are observed for the $\text{Wiki}_{\leq 25}$ and $\text{Wiki}_{>25}$ data sets, a test case scenario we think more interesting, for the reason previously mentioned. For instance, on $\text{Wiki}_{\leq 25}$, when seeking French translations, the best native approach (`Rapp`) performs 2.6% @1 while `R+M+F` achieves 21.3%.

Looking at the performance of an oracle that picks the reference translation whenever it exists in one of the three n-best lists, we realize there is still a gap between the reranking gains obtained and the ones we could possibly achieve. The simplicity of the feature set we considered here is certainly to be blamed.

5.4 Analysis

We analyzed the output of the different approaches and their reranking. We observed in Section 5.2 that native approaches are weak when the source frequency is low. We also noted that native approaches are typically puzzled by cases where the frequency of the source term in the source collection differs significantly from the frequency of its translation in the target collection, especially when the source frequency is much lower than the target one. Overall, the approach of [7] is the one that delivers the most coherent candidate lists: can-

	German		French		Spanish		Romanian	
	@1	@5	@1	@5	@1	@5	@1	@5
Euro_{5-6k}								
best native	M 38.1	**51.9**	M 42.0	59.0	M 36.8	46.1	M 24.6	**37.9**
best reranked	M **39.0**	49.8	M 47.0	68.1	M 37.0	46.3	M **26.9**	37.8
R+M+F	37.8	50.0	**47.6**	**68.5**	**37.3**	**46.7**	26.1	37.8
oracle	*63.6*		*84.4*		*52.3*		*49.0*	
$\text{Wiki}_{>25}$								
best native	M 19.1	27.8	R 20.0	33.0	M 18.1	26.0	M 5.4	10.1
best reranked	M 20.2	27.9	M 38.1	49.0	M 21.5	27.2	M 9.5	13.3
R+M+F	**21.1**	**29.6**	**45.6**	**59.6**	**23.9**	**30.7**	**13.0**	**17.8**
oracle	*49.6*		*69.3*		*42.1*		*26.0*	
$\text{Wiki}_{\leq 25}$								
best native	R 1.0	2.6	R 2.6	4.3	R 0.9	1.6	R 0.2	0.7
best reranked	F 4.6	6.0	M 16.6	19.0	F **9.1**	9.4	F 2.5	3.2
R+M+F	**5.0**	**6.4**	**21.3**	**24.4**	8.9	**10.2**	**3.2**	**4.1**
oracle	*14.1*		*28.6*		*14.1*		*5.7*	

Table 4. @1 and @5 of the best native approach (according to @1), the best reranked native approach, as well as the reranking of the candidates produced by the 3 approaches (R+M+F). R stands for Rapp, M for Miko, and F for Faru. *oracle* picks the reference translation whenever it exists in one n-best list.

didates are often synonyms, antonyms or syntactic derivations of the expected translation.

We identified two patterns that characterize the reranker. First, it tends to prefer source and target pairs that have small edit-distance. Second, and to a less extent, it does prefer pairs of words where the difference between the source and target frequency is small.

Table 5 shows two examples of the top-4 candidates produced by each native approach, as well as their reranking. The gain of reranking can be important as in the second example where the reference translation gets a final rank of 3, while the best rank of a native approach was 44.

6 Conclusion

We compared the distributional-based approaches of [1], [7], and [22] for the task of identifying the translation of (English) words in different Wikipedia collections (German, French, Spanish and Romanian).

Our experiments suggest that we are terribly in need of approaches that are able to manage rare words, a test case scenario we feel is more useful, since frequent translation pairs are likely to be listed in existing bilingual lexicons. Overall, we found that the approach of [7] is more stable than the other two we tested. To our satisfaction, the reranking approach we developed, delivers

uncrushable	infroissable				
Rapp	senente	imbroyables	pulvérix	attriteur	∅
Miko	nattées	paumage	**infroissable**	mouillettes	3
Faru	roninson	ospovat	talanov	mouraviova	∅
reranker	**infroissable**	incrustait	raquettistes	paludicroque	1

brotherliness	confraternité				
Rapp.	qoudous	tâche	attentifs	crainte	44
Miko	joie	volonté	envie	intensité	68
Faru	observant	cueuillies	concordent	moisson	∅
reranker	volonté	précepte	**fraternel**	désintéressé	3

Table 5. Top-4 translations produced by each native approach for two English-French term pairs (uncrushable/infroissable and brotherliness/confraternité), as well as their reranking. The last column indicates the rank of the reference translation in the top-100 candidates, or ∅ if it is absent from a list.

stable and sometimes drastic gains over native approaches. Reranking several candidate lists is overall preferable. A reranker can be trained very rapidly on a few hundred pairs of translations, exploiting a very narrow feature set.

This work leaves open a number of issues that deserve further investigations. First, it is natural to extend the list of features we considered here. The work of [15] provides a list of promising ones that we want to consider, although some are specific to news texts or Wikipedia. Also, we only combined 3 approaches with our reranker, while others could be attempted. We noticed that for an approach to work, we should better adjust its meta-parameters to the task. Investigating the reranking of different variants of a given approach would be interesting, and might be a solution for avoiding to adjust one approach to a specific task.

Acknowledgments

This work has been partly funded by the FRQNT. We are grateful to reviewers for their insightful comments.

References

1. Rapp, R.: Identifying Word Translations in Non-parallel Texts. In: Proceedings of the 33rd ACL. (1995) 320–322
2. Sharoff, S., Rapp, R., Zweigenbaum, P.: Overviewing Important Aspects of the Last Twenty Years of Research in Comparable Corpora. In: Building and Using Comparable Corpora. Springer Berlin Heidelberg (2013) 1–17
3. Yu, K., Tsujii, J.: Extracting bilingual dictionary from comparable corpora with dependency heterogeneity. In: Annual Conference of the NAACL, Companion Volume: Short Papers. (2009) 121–124

4. Otero, P.G.: Learning bilingual lexicons from comparable english and spanish corpora. Proceedings of MT Summit xI (2007) 191–198
5. Daille, B., Morin, E.: Effective Compositional Model for Lexical Alignment. In: Proceedings of the 3rd IJCNLP. (2008) 95–102
6. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. In: Proceedings of ICLR Workshop. (2013)
7. Mikolov, T., Le, Q.V., Sutskever, I.: Exploiting similarities among languages for machine translation. CoRR (2013)
8. Coulmance, J., Marty, J.M., Wenzek, G., Benhalloum, A.: Trans-gram, Fast Cross-lingual Word-embeddings. In: Proceedings of EMNLP. (2015) 1109–1113
9. Vulic, I., Moens, M.F.: Bilingual word embeddings from non-parallel document-aligned data applied to bilingual lexicon induction. In: Proceedings of the 53rd ACL. (2015)
10. Luong, T., Pham, H., Manning, C.D.: Bilingual word representations with monolingual quality in mind. In: Proceedings of the 1st Workshop on Vector Space Modeling for Natural Language Processing. (2015) 151–159
11. Gouws, S., Bengio, Y., Corrado, G.: BilBOWA: Fast Bilingual Distributed Representations without Word Alignments. In: Proceedings of the 32nd ICML. (2015) 748–756
12. Duong, L., Kanayama, H., Ma, T., Bird, S., Cohn, T.: Learning Crosslingual Word Embeddings without Bilingual Corpora. arXiv preprint arXiv:1606.09403 (2016)
13. Upadhyay, S., Faruqui, M., Dyer, C., Roth, D.: Cross-lingual Models of Word Embeddings: An Empirical Comparison. In: Proceedings of ACL. (2016)
14. Levy, O., Sgaard, A., Goldberg, Y.: Reconsidering cross-lingual word embeddings. arXiv preprint arXiv:1608.05426 (2016)
15. Irvine, A., Callison-Burch, C.: Supervised Bilingual Lexicon Induction with Multiple Monolingual Signals. In: Proceedings of NAACL-HLT. (2013) 518–523
16. Delpech, E., Daille, B., Morin, E., Lemaire, C.: Extraction of domain-specific bilingual lexicon from comparable corpora: compositional translation and ranking. Proceedings of COLING (2012)
17. Harastani, R., Daille, B., Morin, E.: Ranking Translation Candidates Acquired from Comparable Corpora. In: Proceedings of the 6th IJCNL. (2013) 401–409
18. Kontonatsios, G., Korkontzelos, I., Tsujii, J., Ananiadou, S.: Combining String and Context Similarity for Bilingual Term Alignment from Comparable Corpora. In: Proceedings of EMNLP. (2014) 1701–1712
19. Baldwin, T., Tanaka, T.: Translation by machine of complex nominals: Getting it right. In: Proceedings of the Workshop on Multiword Expressions: Integrating Processing. (2004) 24–31
20. Jakubina, L., Langlais, P.: Reranking translation candidates produced by several bilingual word similarity sources. In: 15th Conference of the European Chapter of the Association for Computational Linguitics. (April 2017)
21. Dinu, G., Baroni, M.: Improving zero-shot learning by mitigating the hubness problem. CoRR (2014)
22. Faruqui, M., Dyer, C.: Improving Vector Space Word Representations Using Multilingual Correlation. In: Proceedings of EACL. (2014)
23. Li, H., Kumaran, A., Pervouchine, V., Zhang, M.: Report of news 2009 machine transliteration shared task. In: Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration. NEWS '09 (2009) 1–18