

# Together We Can: Bilingual Bootstrapping for WSD

Mitesh M. Khapra Salil Joshi Arindam Chatterjee Pushpak Bhattacharyya

Department Of Computer Science and Engineering,

IIT Bombay,

Powai,

Mumbai, 400076.

{miteshk, salilj, arindam, pb}@cse.iitb.ac.in

## Abstract

Recent work on bilingual Word Sense Disambiguation (WSD) has shown that a resource deprived language ( $L_1$ ) can benefit from the annotation work done in a resource rich language ( $L_2$ ) via parameter projection. However, this method assumes the presence of sufficient annotated data in one resource rich language which may not always be possible. Instead, we focus on the situation where there are two resource deprived languages, both having a very small amount of seed annotated data and a large amount of untagged data. We then use *bilingual bootstrapping*, wherein, a model trained using the seed annotated data of  $L_1$  is used to annotate the untagged data of  $L_2$  and vice versa using parameter projection. The untagged instances of  $L_1$  and  $L_2$  which get annotated with high confidence are then added to the seed data of the respective languages and the above process is repeated. Our experiments show that such a bilingual bootstrapping algorithm when evaluated on two different domains with *small seed sizes* using Hindi ( $L_1$ ) and Marathi ( $L_2$ ) as the language pair performs better than monolingual bootstrapping and significantly reduces annotation cost.

## 1 Introduction

The high cost of collecting sense annotated data for supervised approaches (Ng and Lee, 1996; Lee et al., 2004) has always remained a matter of concern for some of the resource deprived languages of the world. The problem is even more hard-hitting for multilingual regions (*e.g.*, India which has more than 20 constitutionally recognized languages). To circumvent this problem, unsupervised and knowledge based approaches (Lesk, 1986; Walker and Amsler, 1986; Agirre and Rigau, 1996; McCarthy et al.,

2004; Mihalcea, 2005) have been proposed as an alternative but they have failed to deliver good accuracies. Semi-supervised approaches (Yarowsky, 1995) which use a small amount of annotated data and a large amount of untagged data have shown promise albeit for a limited set of target words. The above situation highlights the need for high accuracy resource conscious approaches to all-words multilingual WSD.

Recent work by Khapra et al. (2010) in this direction has shown that it is possible to perform cost effective WSD in a target language ( $L_2$ ) without compromising much on accuracy by leveraging on the annotation work done in another language ( $L_1$ ). This is achieved with the help of a novel synset-aligned multilingual dictionary which facilitates the projection of parameters learned from the Wordnet and annotated corpus of  $L_1$  to  $L_2$ . This approach thus obviates the need for collecting large amounts of annotated corpora in multiple languages by relying on sufficient annotated corpus in one resource rich language. However, in many situations such a pivot resource rich language itself may not be available. Instead, we might have two or more languages having a small amount of annotated corpus and a large amount of untagged corpus. Addressing such situations is the main focus of this work. Specifically, we address the following question:

*In the absence of a pivot resource rich language is it possible for two resource deprived languages to mutually benefit from each other's annotated data?*

While addressing the above question we assume that even though it is hard to obtain large amounts of annotated data in multiple languages, it should be fairly easy to obtain a large amount of untagged data in these languages. We leverage on such untagged

data by employing a bootstrapping strategy. The idea is to train an initial model using a small amount of annotated data in both the languages and iteratively expand this seed data by including untagged instances which get tagged with a high confidence in successive iterations. Instead of using monolingual bootstrapping, we use bilingual bootstrapping via parameter projection. In other words, the parameters learned from the annotated data of  $L_1$  (and  $L_2$  respectively) are projected to  $L_2$  (and  $L_1$  respectively) and the projected model is used to tag the untagged instances of  $L_2$  (and  $L_1$  respectively).

Such a bilingual bootstrapping strategy when tested on two domains, *viz.*, Tourism and Health using Hindi ( $L_1$ ) and Marathi ( $L_2$ ) as the language pair, consistently does better than a baseline strategy which uses only seed data for training without performing any bootstrapping. Further, it consistently performs better than monolingual bootstrapping. A simple and intuitive explanation for this is as follows. In monolingual bootstrapping a language can benefit only from its own seed data and hence can tag only those instances with high confidence which it has already seen. On the other hand, in bilingual bootstrapping a language can benefit from the seed data available in the other language which was not previously seen in its self corpus. This is very similar to the process of co-training (Blum and Mitchell, 1998) wherein the annotated data in the two languages can be seen as two different views of the same data. Hence, the classifier trained on one view can be improved by adding those untagged instances which are tagged with a high confidence by the classifier trained on the other view.

The remainder of this paper is organized as follows. In section 2 we present related work. Section 3 describes the Synset aligned multilingual dictionary which facilitates parameter projection. Section 4 discusses the work of Khapra et al. (2009) on parameter projection. In section 5 we discuss bilingual bootstrapping which is the main focus of our work followed by a brief discussion on monolingual bootstrapping. Section 6 describes the experimental setup. In section 7 we present the results followed by discussion in section 8. Section 9 concludes the paper.

## 2 Related Work

Bootstrapping for Word Sense Disambiguation was first discussed in (Yarowsky, 1995). Starting with a very small number of seed collocations an initial decision list is created. This decision list is then applied to untagged data and the instances which get tagged with a high confidence are added to the seed data. This algorithm thus proceeds iteratively increasing the seed size in successive iterations. This monolingual bootstrapping method showed promise when tested on a limited set of target words but was not tried for all-words WSD.

The failure of monolingual approaches (Ng and Lee, 1996; Lee et al., 2004; Lesk, 1986; Walker and Amsler, 1986; Agirre and Rigau, 1996; McCarthy et al., 2004; Mihalcea, 2005) to deliver high accuracies for all-words WSD at low costs created interest in bilingual approaches which aim at reducing the annotation effort. Recent work in this direction by Khapra et al. (2009) aims at reducing the annotation effort in multiple languages by leveraging on existing resources in a pivot language. They showed that it is possible to project the parameters learned from the annotation work of one language to another language provided aligned Wordnets for the two languages are available. However, they do not address situations where two resource deprived languages have aligned Wordnets but neither has sufficient annotated data. In such cases bilingual bootstrapping can be used so that the two languages can mutually benefit from each other's small annotated data.

Li and Li (2004) proposed a bilingual bootstrapping approach for the more specific task of Word Translation Disambiguation (WTD) as opposed to the more general task of WSD. This approach does not need parallel corpora (just like our approach) and relies only on in-domain corpora from two languages. However, their work was evaluated only on a handful of target words (9 nouns) for WTD as opposed to the broader task of WSD. Our work instead focuses on improving the performance of all words WSD for two resource deprived languages using bilingual bootstrapping. At the heart of our work lies *parameter projection* facilitated by a synset aligned multilingual dictionary described in the next section.

### 3 Synset Aligned Multilingual Dictionary

A novel and effective method of storage and use of dictionary in a multilingual setting was proposed by Mohanty et al. (2008). For the purpose of current discussion, we will refer to this multilingual dictionary framework as *MultiDict*. One important departure in this framework from the traditional dictionary is that **synsets are linked, and after that the words inside the synsets are linked**. The basic mapping is thus between synsets and thereafter between the words.

Concepts	L1 (English)	L2 (Hindi)	L3 (Marathi)
04321: a youth- ful male person	{male child, boy}	{लड़का ( <i>ladkaa</i> ), बालक ( <i>baalak</i> ), बच्चा ( <i>bachchaa</i> )}	{मुलगा ( <i>mulgaa</i> ), पोरगा ( <i>porgaa</i> ), पोर ( <i>por</i> )}

Table 1: Multilingual Dictionary Framework

Table 1 shows the structure of MultiDict, with one example row standing for the concept of *boy*. The first column is the pivot describing a concept with a unique ID. The subsequent columns show the words expressing the concept in respective languages (in the example table, *English, Hindi and Marathi*). After the synsets are linked, cross linkages are set up manually from the words of a synset to the words of a linked synset of the pivot language. For example, for the Marathi word मुलगा (*mulgaa*), “a youthful male person”, the correct lexical substitute from the corresponding Hindi synset is लड़का (*ladkaa*). The average number of such links per synset per language pair is approximately 3. However, since our work takes place in a semi-supervised setting, we do not assume the presence of these manual cross linkages between synset members. Instead, in the above example, we assume that all the words in the Hindi synset are equally probable translations of every word in the corresponding Marathi synset. Such cross-linkages between synset members facilitate parameter projection as explained in the next section.

### 4 Parameter Projection

Khapra et al. (2009) proposed that the various parameters essential for domain-specific Word Sense Disambiguation can be broadly classified into two categories:

#### Wordnet-dependent parameters:

- belongingness-to-dominant-concept
- conceptual distance
- semantic distance

#### Corpus-dependent parameters:

- sense distributions
- corpus co-occurrence

They proposed a scoring function (Equation (1)) which combines these parameters to identify the correct sense of a word in a context:

$$S^* = \arg \max_i (\theta_i V_i + \sum_{j \in J} W_{ij} * V_i * V_j) \quad (1)$$

where,

$i \in \text{Candidate Synsets}$

$J = \text{Set of disambiguated words}$

$\theta_i = \text{BelongingnessToDominantConcept}(S_i)$

$V_i = P(S_i | \text{word})$

$W_{ij} = \text{CorpusCooccurrence}(S_i, S_j)$

\*  $1/WN\text{ConceptualDistance}(S_i, S_j)$

\*  $1/WN\text{SemanticGraphDistance}(S_i, S_j)$

The first component  $\theta_i V_i$  of Equation (1) captures influence of the corpus specific sense of a word in a domain. The other component  $W_{ij} * V_i * V_j$  captures the influence of interaction of the candidate sense with the senses of context words weighted by factors of co-occurrence, conceptual distance and semantic distance.

**Wordnet-dependent parameters** depend on the structure of the Wordnet whereas the **Corpus-dependent parameters** depend on various statistics learned from a sense marked corpora. Both the tasks of (a) constructing a Wordnet from scratch and (b) collecting sense marked corpora for multiple languages are tedious and expensive. Khapra et

al. (2009) observed that by *projecting relations* from the Wordnet of a language and by *projecting corpus statistics* from the sense marked corpora of the language to those of the target language, *the effort required in constructing semantic graphs for multiple Wordnets and collecting sense marked corpora for multiple languages can be avoided or reduced*. At the heart of their work lies the *MultiDict* described in previous section which facilitates parameter projection in the following manner:

1. By linking with the synsets of a pivot resource rich language (Hindi, in our case), the cost of building Wordnets of other languages is partly reduced (semantic relations are inherited). The Wordnet parameters of Hindi Wordnet now become projectable to other languages.
2. For calculating corpus specific sense distributions,  $P(\text{Sense } S_i | \text{Word } W)$ , we need the counts,  $\#(S_i, W)$ . By using cross linked words in the synsets, these counts become projectable to the target language (Marathi, in our case) as they can be approximated by the counts of the cross linked Hindi words calculated from the Hindi sense marked corpus as follows:

$$P(S_i | W) = \frac{\#(S_i, \text{marathi\_word})}{\sum_j \#(S_j, \text{marathi\_word})}$$

$$P(S_i | W) \approx \frac{\#(S_i, \text{cross\_linked\_hindi\_word})}{\sum_j \#(S_j, \text{cross\_linked\_hindi\_word})}$$

The rationale behind the above approximation is the observation that within a domain the counts of cross-linked words will remain the same across languages.

This parameter projection strategy as explained above lies at the heart of our work and allows us to perform bilingual bootstrapping by projecting the models learned from one language to another.

## 5 Bilingual Bootstrapping

We now come to the main contribution of our work, *i.e.*, bilingual bootstrapping. As shown in Algorithm 1, we start with a small amount of seed data ( $LD_1$  and  $LD_2$ ) in the two languages. Using this data we learn the parameters described in the previous section. We collectively refer to the parameters learned

---

### Algorithm 1 Bilingual Bootstrapping

---

$LD_1 :=$  Seed Labeled Data from  $L_1$   
 $LD_2 :=$  Seed Labeled Data from  $L_2$   
 $UD_1 :=$  Unlabeled Data from  $L_1$   
 $UD_2 :=$  Unlabeled Data from  $L_2$

**repeat**

$\theta_1 :=$  model trained using  $LD_1$   
 $\theta_2 :=$  model trained using  $LD_2$

{Project models from  $L_1/L_2$  to  $L_2/L_1$ }

$\hat{\theta}_2 := \text{project}(\theta_1, L_2)$

$\hat{\theta}_1 := \text{project}(\theta_2, L_1)$

**for all**  $u_1 \in UD_1$  **do**

$s :=$  sense assigned by  $\hat{\theta}_1$  to  $u_1$

**if**  $\text{confidence}(s) > \epsilon$  **then**

$LD_1 := LD_1 + u_1$

$UD_1 := UD_1 - u_1$

**end if**

**end for**

**for all**  $u_2 \in UD_2$  **do**

$s :=$  sense assigned by  $\hat{\theta}_2$  to  $u_2$

**if**  $\text{confidence}(s) > \epsilon$  **then**

$LD_2 := LD_2 + u_2$

$UD_2 := UD_2 - u_2$

**end if**

**end for**

**until** *convergence*

---

from the seed data as models  $\theta_1$  and  $\theta_2$  for  $L_1$  and  $L_2$  respectively. The parameter projection strategy described in the previous section is then applied to  $\theta_1$  and  $\theta_2$  to obtain the projected models  $\hat{\theta}_2$  and  $\hat{\theta}_1$  respectively. These projected models are then applied to the untagged data of  $L_1$  and  $L_2$  and the instances which get labeled with a high confidence are added to the labeled data of the respective languages. This process is repeated till we reach convergence, *i.e.*, till it is no longer possible to move any data from  $UD_1$  (and  $UD_2$ ) to  $LD_1$  (and  $LD_2$  respectively).

We compare our algorithm with monolingual bootstrapping where the self models  $\theta_1$  and  $\theta_2$  are directly used to annotate the unlabeled instances in  $L_1$  and  $L_2$  respectively instead of using the projected models  $\hat{\theta}_1$  and  $\hat{\theta}_2$ . The process of monolingual boot-

---

**Algorithm 2 Monolingual Bootstrapping**

---

 $LD_1 :=$  Seed Labeled Data from  $L_1$  $LD_2 :=$  Seed Labeled Data from  $L_2$  $UD_1 :=$  Unlabeled Data from  $L_1$  $UD_2 :=$  Unlabeled Data from  $L_2$ **repeat** $\theta_1 :=$  model trained using  $LD_1$  $\theta_2 :=$  model trained using  $LD_2$ **for all**  $u_1 \in UD_1$  **do** $s :=$  sense assigned by  $\theta_1$  to  $u_1$ **if**  $\text{confidence}(s) > \epsilon$  **then** $LD_1 := LD_1 + u_1$  $UD_1 := UD_1 - u_1$ **end if****end for****for all**  $u_2 \in UD_2$  **do** $s :=$  sense assigned by  $\theta_2$  to  $u_2$ **if**  $\text{confidence}(s) > \epsilon$  **then** $LD_2 := LD_2 + u_2$  $UD_2 := UD_2 - u_2$ **end if****end for****until** *convergence*

---

strapping is shown in Algorithm 2.

## 6 Experimental Setup

We used the publicly available dataset<sup>1</sup> described in Khapra et al. (2010) for all our experiments. The data was collected from two domains, *viz.*, Tourism and Health. The data for Tourism domain was collected by manually translating English documents downloaded from Indian Tourism websites into Hindi and Marathi. Similarly, English documents for Health domain were obtained from two doctors and were manually translated into Hindi and Marathi. The entire data was then manually annotated by three lexicographers adept in Hindi and Marathi. The various statistics pertaining to the total number of words, number of words per POS category and average degree of polysemy are described in Tables 2 to 5.

Although Tables 2 and 3 also report the num-

---

<sup>1</sup>[http://www.cfilt.iitb.ac.in/wsd/annotated\\_corpus](http://www.cfilt.iitb.ac.in/wsd/annotated_corpus)

Category	Polysemous words		Monosemous words	
	Tourism	Health	Tourism	Health
<b>Noun</b>	62336	24089	35811	18923
<b>Verb</b>	6386	1401	3667	5109
<b>Adjective</b>	18949	8773	28998	12138
<b>Adverb</b>	4860	2527	13699	7152
<b>All</b>	92531	36790	82175	43322

Table 2: Polysemous and Monosemous words per category in each domain for Hindi

Category	Polysemous words		Monosemous words	
	Tourism	Health	Tourism	Health
<b>Noun</b>	45589	17482	27386	11383
<b>Verb</b>	7879	3120	2672	1500
<b>Adjective</b>	13107	4788	16725	6032
<b>Adverb</b>	4036	1727	5023	1874
<b>All</b>	70611	27117	51806	20789

Table 3: Polysemous and Monosemous words per category in each domain for Marathi

Category	Avg. degree of Wordnet polysemy for polysemous words	
	Tourism	Health
<b>Noun</b>	3.02	3.17
<b>Verb</b>	5.05	6.58
<b>Adjective</b>	2.66	2.75
<b>Adverb</b>	2.52	2.57
<b>All</b>	3.09	3.23

Table 4: Average degree of Wordnet polysemy per category in the 2 domains for Hindi

Category	Avg. degree of Wordnet polysemy for polysemous words	
	Tourism	Health
<b>Noun</b>	3.06	3.18
<b>Verb</b>	4.96	5.18
<b>Adjective</b>	2.60	2.72
<b>Adverb</b>	2.44	2.45
<b>All</b>	3.14	3.29

Table 5: Average degree of Wordnet polysemy per category in the 2 domains for Marathi

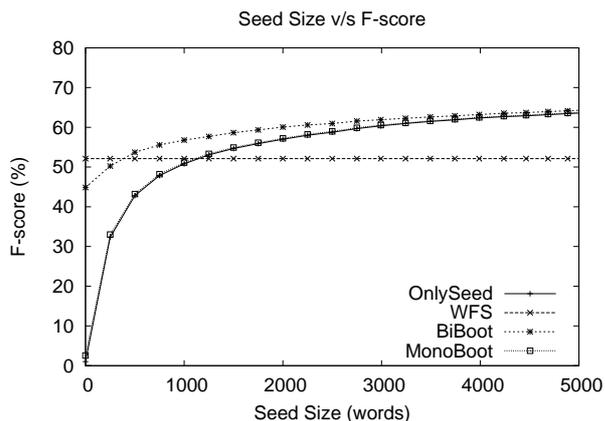


Figure 1: Comparison of *BiBoot*, *MonoBoot*, *OnlySeed* and *WFS* on Hindi Health data

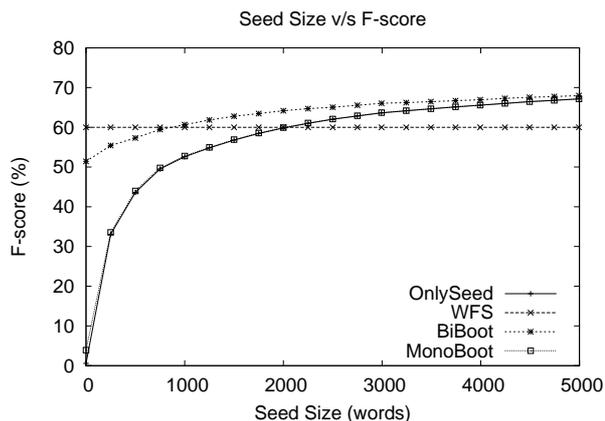


Figure 2: Comparison of *BiBoot*, *MonoBoot*, *OnlySeed* and *WFS* on Hindi Tourism data

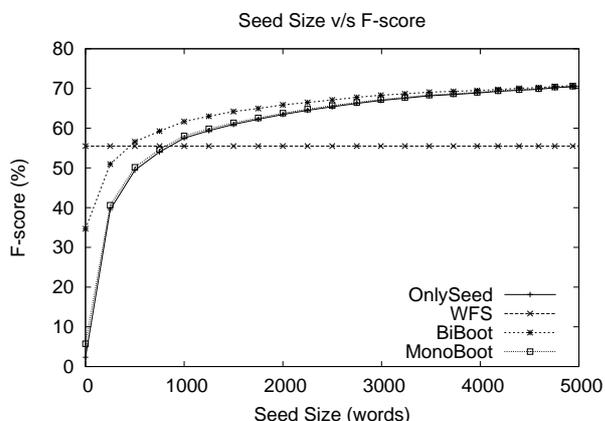


Figure 3: Comparison of *BiBoot*, *MonoBoot*, *OnlySeed* and *WFS* on Marathi Health data

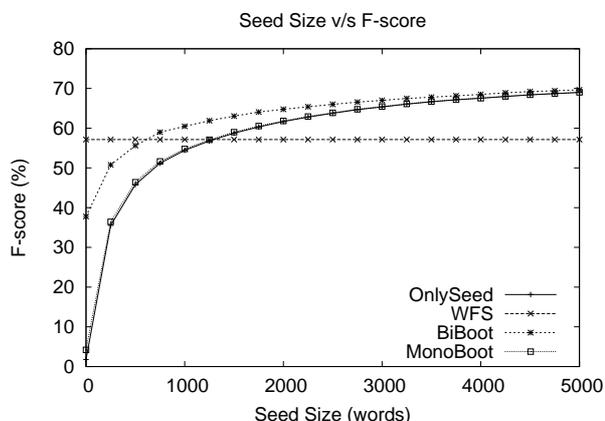


Figure 4: Comparison of *BiBoot*, *MonoBoot*, *OnlySeed* and *WFS* on Marathi Tourism data

ber of monosemous words, we would like to clearly state that we do not consider monosemous words while evaluating the performance of our algorithms (as monosemous words do not need any disambiguation).

We did a 4-fold cross validation of our algorithm using the above described corpora. Note that even though the corpora were parallel we did not use this property in any way in our experiments or algorithm. In fact, the documents in the two languages were randomly split into 4 folds without ensuring that the parallel documents remain in the same folds for the two languages. We experimented with different seed sizes varying from 0 to 5000 in steps of 250. The seed annotated data and untagged instances for bootstrapping are extracted from 3 folds of the data and

the final evaluation is done on the held-out data in the 4th fold.

We ran both the bootstrapping algorithms (*i.e.*, *monolingual bootstrapping* and *bilingual bootstrapping*) for 10 iterations but, we observed that after 1-2 iterations the algorithms converge. In each iteration only those words for which  $P(\text{assigned\_sense}|\text{word}) > 0.6$  get moved to the labeled data. Ideally, this threshold (0.6) should have been selected using a development set. However, since our work focuses on resource scarce languages we did not want to incur the additional cost of using a development set. Hence, we used a fixed threshold of 0.6 so that in each iteration only those words get moved to the labeled data for which the assigned sense is clearly a majority sense ( $P > 0.6$ ).

Language-Domain	Algorithm	F-score(%)	No. of tagged words needed to achieve this F-score	% Reduction in annotation cost
Hindi-Health	<i>BiBoot</i>	57.70	1250	$\frac{(2250+2250)-(1250+1750)}{(2250+2250)} * 100 = 33.33\%$
	<i>OnlySeed</i>	57.99	2250	
Marathi-Health	<i>BiBoot</i>	64.97	1750	
	<i>OnlySeed</i>	64.51	2250	
Hindi-Tourism	<i>BiBoot</i>	60.67	1000	$\frac{(2000+2000)-(1000+1250)}{(2000+2000)} * 100 = 43.75\%$
	<i>OnlySeed</i>	59.83	2000	
Marathi-Tourism	<i>BiBoot</i>	61.90	1250	
	<i>OnlySeed</i>	61.68	2000	

Table 6: Reduction in annotation cost achieved using Bilingual Bootstrapping

## 7 Results

The results of our experiments are summarized in Figures 1 to 4. The  $x$ -axis represents the amount of seed data used and the  $y$ -axis represents the F-scores obtained. The different curves in each graph are as follows:

- BiBoot***: This curve represents the F-score obtained after 10 iterations by using bilingual bootstrapping with different amounts of seed data.
- MonoBoot***: This curve represents the F-score obtained after 10 iterations by using monolingual bootstrapping with different amounts of seed data.
- OnlySeed***: This curve represents the F-score obtained by training on the seed data alone without using any bootstrapping.
- WFS***: This curve represents the F-score obtained by simply selecting the first sense from Wordnet, a typically reported baseline.

## 8 Discussions

In this section we discuss the important observations made from Figures 1 to 4.

### 8.1 Performance of Bilingual bootstrapping

For small seed sizes, the F-score of bilingual bootstrapping is consistently better than the F-score obtained by training only on the seed data without using any bootstrapping. This is true for both the languages in both the domains. Further, bilingual bootstrapping also does better than monolingual bootstrapping for small seed sizes. As explained earlier,

this better performance can be attributed to the fact that in monolingual bootstrapping the algorithm can tag only those instances with high confidence which it has already seen in the training data. Hence, in successive iterations, very little new information becomes available to the algorithm. This is clearly evident from the fact that the curve of monolingual bootstrapping (*MonoBoot*) is always close to the curve of *OnlySeed*.

### 8.2 Effect of seed size

The benefit of bilingual bootstrapping is clearly felt for small seed sizes. However, as the seed size increases the performance of the 3 algorithms, *viz.*, *MonoBoot*, *BiBoot* and *OnlySeed* is more or less the same. This is intuitive, because, as the seed size increases the algorithm is able to see more and more tagged instances in its self corpora and hence does not need any assistance from the other language. In other words, the annotated data in  $L_1$  is not able to add any new information to the training process of  $L_2$  and vice versa.

### 8.3 Bilingual bootstrapping reduces annotation cost

The performance boost obtained at small seed sizes suggests that bilingual bootstrapping helps to reduce the overall annotation costs for both the languages. To further illustrate this, we take some sample points from the graph and compare the number of tagged words needed by *BiBoot* and *OnlySeed* to reach the same (or nearly the same) F-score. We present this comparison in Table 6.

The rows for *Hindi-Health* and *Marathi-Health* in Table 6 show that when *BiBoot* is employed we need 1250 tagged words in Hindi and 1750 tagged words in Marathi to attain F-scores of 57.70% and 64.97% respectively. On the other hand, in the absence of bilingual bootstrapping, (*i.e.*, using *OnlySeed*) we need 2250 tagged words each in Hindi and Marathi to achieve similar F-scores. *BiBoot* thus gives a reduction of 33.33% in the overall annotation cost (  $\{1250 + 1750\}$  v/s  $\{2250 + 2250\}$ ) while achieving similar F-scores. Similarly, the results for *Hindi-Tourism* and *Marathi-Tourism* show that *BiBoot* gives a reduction of 43.75% in the overall annotation cost while achieving similar F-scores. Further, since the results of *MonoBoot* are almost the same as *OnlySeed*, the above numbers indicate that *BiBoot* provides a reduction in cost when compared to *MonoBoot* also.

#### 8.4 Contribution of monosemous words in the performance of *BiBoot*

As mentioned earlier, monosemous words in the test set are not considered while evaluating the performance of our algorithm but, we add monosemous words to the seed data. However, we do not count monosemous words while calculating the seed size as there is no manual annotation cost associated with monosemous words (they can be tagged automatically by fetching their singleton sense id from the wordnet). We observed that the monosemous words of  $L_1$  help in boosting the performance of  $L_2$  and vice versa. This is because for a given monosemous word in  $L_2$  (or  $L_1$  respectively) the corresponding cross-linked word in  $L_1$  (or  $L_2$  respectively) need not necessarily be monosemous. In such cases, the cross-linked polysemous word in  $L_2$  (or  $L_1$  respectively) benefits from the projected statistics of a monosemous word in  $L_1$  (or  $L_2$  respectively). This explains why *BiBoot* gives an F-score of 35-52% even at zero seed size even though the F-score of *OnlySeed* is only 2-5% (see Figures 1 to 4).

## 9 Conclusion

We presented a bilingual bootstrapping algorithm for Word Sense Disambiguation which allows two resource deprived languages to mutually benefit

from each other's data via parameter projection. The algorithm consistently performs better than monolingual bootstrapping. It also performs better than using only monolingual seed data without using any bootstrapping. The benefit of bilingual bootstrapping is felt prominently when the seed size in the two languages is very small thus highlighting the usefulness of this algorithm in highly resource constrained scenarios.

## References

- Eneko Agirre and German Rigau. 1996. Word sense disambiguation using conceptual density. In *In Proceedings of the 16th International Conference on Computational Linguistics (COLING)*.
- Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. pages 92–100. Morgan Kaufmann Publishers.
- Mitesh M. Khapra, Sapan Shah, Piyush Kedia, and Pushpak Bhattacharyya. 2009. Projecting parameters for multilingual word sense disambiguation. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 459–467, Singapore, August. Association for Computational Linguistics.
- Mitesh Khapra, Saurabh Sohoney, Anup Kulkarni, and Pushpak Bhattacharyya. 2010. Value for money: Balancing annotation effort, lexicon building and accuracy for multilingual wsd. In *Proceedings of the 23rd International Conference on Computational Linguistics*.
- Yong Keok Lee, Hwee Tou Ng, and Tee Kiah Chia. 2004. Supervised word sense disambiguation with support vector machines and multiple knowledge sources. In *Proceedings of Senseval-3: Third International Workshop on the Evaluation of Systems for the Semantic Analysis of Text*, pages 137–140.
- Michael Lesk. 1986. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *In Proceedings of the 5th annual international conference on Systems documentation*.
- Hang Li and Cong Li. 2004. Word translation disambiguation using bilingual bootstrapping. *Comput. Linguist.*, 30:1–22, March.
- Diana McCarthy, Rob Koeling, Julie Weeds, and John Carroll. 2004. Finding predominant word senses in untagged text. In *ACL '04: Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, page 279, Morristown, NJ, USA. Association for Computational Linguistics.

- Rada Mihalcea. 2005. Large vocabulary unsupervised word sense disambiguation with graph-based algorithms for sequence data labeling. In *In Proceedings of the Joint Human Language Technology and Empirical Methods in Natural Language Processing Conference (HLT/EMNLP)*, pages 411–418.
- Rajat Mohanty, Pushpak Bhattacharyya, Prabhakar Pande, Shraddha Kalele, Mitesh Khapra, and Aditya Sharma. 2008. Synset based multilingual dictionary: Insights, applications and challenges. In *Global Wordnet Conference*.
- Hwee Tou Ng and Hian Beng Lee. 1996. Integrating multiple knowledge sources to disambiguate word senses: An exemplar-based approach. In *In Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 40–47.
- D. Walker and R. Amsler. 1986. The use of machine readable dictionaries in sublanguage analysis. In *In Analyzing Language in Restricted Domains, Grishman and Kittredge (eds)*, LEA Press, pages 69–83.
- David Yarowsky. 1995. Unsupervised word sense disambiguation rivaling supervised methods. In *Proceedings of the 33rd annual meeting on Association for Computational Linguistics*, pages 189–196, Morristown, NJ, USA. Association for Computational Linguistics.