Can One Language Bootstrap the Other: A Case Study on Event Extraction

Zheng Chen
The Graduate Center
The City University of New York
zchen1@gc.cuny.edu

Heng Ji
Queens College and The Graduate Center
The City University of New York
hengji@cs.qc.cuny.edu

Abstract

This paper proposes a new bootstrapping framework using cross-lingual information projection. We demonstrate that this framework is particularly effective for a challenging NLP task which is situated at the end of a pipeline and thus suffers from the errors propagated from upstream processing and has low-performance baseline. Using Chinese event extraction as a case study and bitexts as a new source of information, we present three bootstrapping techniques. We first conclude that the standard mono-lingual bootstrapping approach is not so effective. Then we exploit a second approach that potentially benefits from the extra information captured by an English event extraction system and projected into Chinese. Such a cross-lingual scheme produces significant performance gain. Finally we show that the combination of mono-lingual and cross-lingual information in bootstrapping can further enhance the performance. Ultimately this new framework obtained 10.1% relative improvement in trigger labeling (F-measure) and 9.5% relative improvement in argument-labeling.

1 Introduction

Bootstrapping methods can reduce the efforts needed to develop a training set and have shown promise in improving the performance of many tasks such as name tagging (Miller et al., 2004; Ji and Grishman, 2006), semantic class extraction (Lin et al., 2003), chunking (Ando and Zhang, 2005), coreference resolution (Bean and Riloff, 2004) and text classification (Blum and Mitchell, 1998). Most of these bootstrapping methods implicitly assume that:

- There exists a high-accuracy 'seed set' or 'seed model' as the baseline;
- There exists unlabeled data which is reliable and relevant to the test set in some aspects, e.g. from similar time frames and news sources; and therefore the unlabeled data supports the acquisition of new information, to provide new evidence to be incorporated to bootstrap the model and reduce the sparse data problem.
- The seeds and unlabeled data won't make the old estimates worse by adding too many incorrect instances.

However, for some more comprehensive and challenging tasks such as event extraction, the performance of the seed model suffers from the limited annotated training data and also from the errors propagated from upstream processing such as part-of-speech tagging and parsing. In addition, simply relying upon large unlabeled corpora cannot compensate for these limitations because more errors can be propagated from upstream processing such as entity extraction and temporal expression identification.

Inspired from the idea of co-training (Blum and Mitchell, 1998), in this paper we intend to bootstrap an event extraction system in one language (Chinese) by exploring new evidences from the event extraction system in another language (English) via cross-lingual projection. We conjecture that the *cross-lingual bootstrapping* for event extraction can naturally fit the co-training model: a same event is represented in two "views" (described in two languages). Furthermore, the cross-lingual bootstrapping can benefit from the different sources of training data. For example, the Chinese training corpus includes articles from Chinese new agencies in 2000 while most of English training data are from the US news agencies in 2003, thus

Proceedings of the NAACL HLT Workshop on Semi-supervised Learning for Natural Language Processing, pages 66–74,
Boulder, Colorado, June 2009. © 2009 Association for Computational Linguistics

English and Chinese event extraction systems have the nature of generating different results on parallel documents and may complement each other. In this paper, we explore approaches of exploiting the increasingly available bilingual parallel texts (***bitexts***).

We first investigate whether we can improve a Chinese event extraction system by simply using the Chinese side of bitexts in a regular monolingual bootstrapping framework. By gradually increasing the size of the corpus with unlabeled data, we did not get much improvement for trigger labeling and even observed performance deterioration for argument labeling. But then by aligning the texts at the word level, we found that the English event extraction results can be projected into Chinese for bootstrapping and lead to significant improvement. We also obtained clear further improvement by combining mono-lingual and cross-lingual bootstrapping.

The main contributions of this paper are two-fold. We formulate a new algorithm of cross-lingual bootstrapping, and demonstrate its effectiveness in a challenging task of event extraction; and we conclude that, for some applications besides machine translation, effective use of bitexts can be beneficial.

The remainder of the paper is organized as follows. Section 2 formalizes the event extraction task addressed in this paper. Section 3 discusses event extraction bootstrapping techniques. Section 4 reports our experimental results. Section 5 presents related work. Section 6 concludes this paper and points out future directions.

2 Event Extraction

2.1 Task Definition and Terminology

The event extraction that we address in this paper is specified in the Automatic Content Extraction (ACE)[1] program. The ACE 2005 Evaluation defines the following terminology for the event extraction task:

- **event trigger**: the word that most clearly expresses an event's occurrence
- **event argument**: an *entity*, a *temporal expression* or a *value* that plays a certain *role* in the event instance
- **event mention**: a phrase or sentence with a distinguished trigger and participant arguments

The event extraction task in our paper is to detect certain types of event mentions that are indicated by event triggers (*trigger labeling*), recognize the event participants e.g., *who*, *when*, *where*, *how* (*argument labeling*) and merge the co-referenced event mentions into a unified event (*post-processing*). In this paper, we focus on discussing trigger labeling and argument labeling.

In the following example,

Mike got married in 2008.

The event extraction system should identify "*married*" as the event trigger which indicates the event type of "Life" and subtype of "Marry". Furthermore, it should detect "*Mike*" and "*2008*" as arguments in which "*Mike*" has a role of "Person" and "*2008*" has a role of "Time-Within".

2.2 A Pipeline of Event Extraction

Our pipeline framework of event extraction includes trigger labeling, argument labeling and post-processing, similar to (Grishman et al., 2005), (Ahn, 2006) and (Chen and Ji, 2009). We depict the framework as Figure 1.

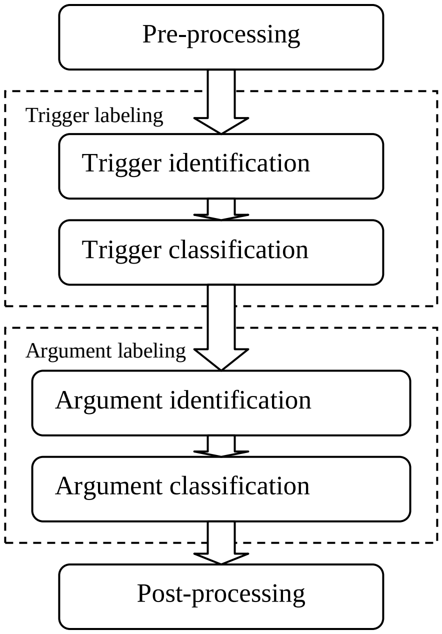

Figure 1. Pipeline of Event Extraction

The event extraction system takes raw documents as input and conducts some pre-processing steps. The texts are automatically annotated with

[1] http://www.nist.gov/speech/tests/ace/

word segmentation, Part-of-Speech tags, parsing structures, entities, time expressions, and relations. The annotated documents are then sent to the following four components. Each component is a classifier and produces confidence values;

- Trigger identification: the classifier recognizes a word or a phrase as the event trigger.
- Trigger classification: the classifier assigns an event type to an identified trigger.
- Argument identification: the classifier recognizes whether an entity, temporal expression or value is an argument associated with a particular trigger in the same sentence.
- Argument classification: the classifier assigns a role to the argument.

The post-processing merges co-referenced event mentions into a unified representation of event.

2.3 Two Monolingual Event Extraction Systems

We use two monolingual event extraction systems, one for English, and the other for Chinese. Both systems employ the above framework and use Maximum Entropy based classifiers. The corresponding classifiers in both systems also share some language-independent features, for example, in trigger identification, both classifiers use the “previous word” and “next word” as features, however, there are some language-dependent features that only work well for one monolingual system, for example, in argument identification, the next word of the candidate argument is a good feature for Chinese system but not for English system. To illustrate this, in the Chinese “*的*” (*of*) structure, the word “*的*” (*of*) strongly suggests that the entity on the left side of “*的*” is not an argument. For a specific example, in “*纽约市的市长*” (*The mayor of New York City*), “*纽约市*” (*New York City*) on the left side of “*的*” (*of*) cannot be considered as an argument because it is a modifier of the noun “*市长*”(*mayor*). Unlike Chinese, “*of*” (“*的*”) appears ahead of the entity in the English phrase.

Table 1 lists the overall Precision (P), Recall (R) and F-Measure (F) scores for trigger labeling and argument labeling in our two monolingual event extraction systems. For comparison, we also list the performance of an English human annotator and a Chinese human annotator.

Table 1 shows that event extraction is a difficult NLP task because even human annotators cannot achieve satisfying performance. Both monolingual systems relied on expensive human labeled data (much more expensive than other NLP tasks due to the extra tagging tasks of entities and temporal expressions), thus a natural question arises: can the monolingual system benefit from bootstrapping techniques with a relative small set of training data? The other question is: can a monolingual system benefit from the other monolingual system by cross-lingual bootstrapping?

Performance System/ Human	Trigger Labeling			Argument Labeling		
	P	R	F	P	R	F
English System	64.3	59.4	61.8	49.2	34.7	40.7
Chinese System	78.8	48.3	59.9	60.6	34.3	43.8
English Annotator	59.2	59.4	59.3	51.6	59.5	55.3
Chinese Annotator	75.2	74.6	74.9	58.6	60.9	59.7

Table 1.Performance of Two Monolingual Event Extraction Systems and Human Annotators

3 Bootstrapping Event Extraction

3.1 General Bootstrapping Algorithm

Bootstrapping algorithms have attracted much attention from researchers because a large number of unlabeled examples are available and can be utilized to boost the performance of a system trained on a small set of labeled examples. The general bootstrapping algorithm is depicted in Figure 2, similar to (Mihalcea, 2004).

Self-training and Co-training are two most commonly used bootstrapping methods.

A typical self-training process is described as follows: it starts with a set of training examples and builds a classifier with the full integrated feature set. The classifier is then used to label an additional portion of the unlabeled examples. Among the resulting labeled examples, put the most confident ones into the training set, and re-train the classifier. This iterates until a certain condition is satisfied (e.g., all the unlabeled examples have been labeled, or it reaches a certain number of iterations).

Co-training(Blum and Mitchell, 1998) differs from self-training in that it assumes that the data can be represented using two or more separate "views" (thus the whole feature set is split into disjoint feature subsets) and each classifier can be trained on one view of the data. For each iteration, both classifiers label an additional portion of the unlabeled examples and put the most confident ones to the training set. Then the two classifiers are retrained on the new training set and iterate until a certain condition is satisfied.

Both self-training and co-training can fit in the general bootstrapping process. If the number of classifiers is set to one, it is a self-training process, and it is a co-training process if there are two different classifiers that interact in the bootstrapping process.

```
Input:
    $L$ : a set of labeled examples,
    $U$ : a set of unlabeled examples
    { $C_i$ }: a set of classifiers
Initialization:
    Create a pool $U'$ of examples by choosing $P$ random examples from $U$
Loop until a condition is satisfied (e.g., $U = \varnothing$ , or iteration counter reaches a preset number $I$ )
    •  Train each classifier $C_i$ on $L$ , and label the examples in $U'$
    •  For each classifier $C_i$ ,select the most confidently labeled examples (e.g., the confidence score is above a preset threshold $\theta$ or the top $K$ ) and add them to $L$
    •  Refill $U'$ with examples from $U$ , and keep the size of $U'$ as constant $P$
```

Figure 2. General Bootstrapping Algorithm.

In the following sections, we adapt the bootstrapping techniques discussed in this section to a larger scale (system level). In other words, we aim to bootstrap the overall performance of the system which may include multiple classifiers, rather than just improve the performance of a single classifier in the system. It is worth noting that for the pipeline event extraction depicted in Section 2.2, there are two major steps that determine the overall system performance: trigger labeling and argument labeling. Furthermore, the performance of trigger labeling can directly affect the performance of argument labeling because the involving arguments are constructed according to the trigger. If a trigger is wrongly recognized, all the involving arguments will be considered as wrong arguments. If a trigger is missing, all the attached arguments will be considered as missing arguments.

3.2 Monolingual Self-training

It is rather smooth to adapt the idea of traditional self-training to monolingual self-training if we consider our monolingual event extraction system as a black box or even a single classifier that determines whether an event combining the result of trigger labeling and argument labeling is a reportable event.

Thus the monolingual self-training procedure for event extraction is quite similar with the one described in Section 3.1. The monolingual event extraction system is first trained on a starting set of labeled documents, and then tag on an additional portion of unlabeled documents. Note that in each labeled document, multiple events could be tagged and confidence score is assigned to each event. Then the labeled documents are added into the training set and the system is retrained based on the events with high confidence. This iterates until all the unlabeled documents have been tagged.

3.3 Cross-lingual Co-Training

We extend the idea of co-training to cross-lingual co-training. The intuition behind cross-lingual co-training is that the same event has different "views" described in different languages, because the lexical unit, the grammar and sentence construction differ from one language to the other. Thus one monolingual event extraction system probably utilizes the language dependent features that cannot work well for the other monolingual event extraction systems. Blum and Mitchell (1998) derived PAC-like guarantees on learning under two assumptions: 1) the two views are individually sufficient for classification and 2) the two views are conditionally independent given the class. Obviously, the first assumption can be satisfied in cross-lingual co-training for event extraction, since each monolingual event extraction system is sufficient for event extraction task. However, we reserve our opinion on the second assumption. Although the two monolingual event extraction systems may apply the same language-independent features such as the part-of-speech, the next word and the previous word, the features are exhibited in their own context of language, thus it is too subjective to conclude that the two feature sets are or are not conditionally independent. It is left to be an unsolved issue which needs further strict analysis and supporting experiments.

The cross-lingual co-training differs from traditional co-training in that the two systems in cross-lingual co-training are not initially trained from the same labeled data. Furthermore, in the bootstrapping phase, each system only labels half portion of the bitexts in its own language. In order to utilize the labeling result by the other system, we need to conduct an extra step named *cross-lingual projection* that transforms tagged events from one language to the other.

3.3.1 A Cross-lingual Co-training Algorithm

The algorithm for cross-lingual co-training is depicted in Figure 3.

```
Input:
    $L_1$ : a set of labeled examples in language A
    $L_2$ : a set of labeled examples in language B
    $U$ : a set of unlabeled bilingual examples (bitexts) with alignment information
    { $S_1, S_2$ }: two monolingual systems, one for language A and the other for language B.
Initialization:
    Create a pool $U'$ of examples by choosing $P$ random examples from $U$
Loop until a condition is satisfied (e.g., $U = \varnothing$, or iteration counter reaches a preset number $I$ )
●   Train $S_1$ on $L_1$ and $S_2$ on $L_2$
●   Use $S_1$ to label the examples in $U'$ (the portion in Language A) and use $S_2$ to label the examples in $U'$ (the portion in Language B)
●   For $S_1$, select the most confidently labeled examples (e.g., the confidence score is above a preset threshold $\theta$ or the top $K$ ), apply the operation of cross-lingual projection, transform the selected examples from Language A to Language B, and put them into $L_2$. The same procedure applies to $S_2$.
●   Refill $U'$ with examples from $U$, and keep the size of $U'$ as constant $P$
```

Figure 3. Cross-lingual Co-training Algorithm

3.3.2 Cross-lingual Semi-co-training

Cross-lingual semi-co-training is a variation of cross-lingual co-training, and it differs from cross-lingual co-training in that it tries to bootstrap only one system by the other fine-trained system. This technique is helpful when we have relatively large amount of training data in one language while we have scarce data in the other language.

Thus we only need to make a small modification in the cross-lingual co-training algorithm so that it can soon be adapted to cross-lingual semi-co-training, i.e., we retrain one system and do not retrain the other. In this paper, we will conduct experiments to investigate whether a fine-trained English event extraction system can bootstrap the Chinese event extraction system, starting from a small set of training data.

3.3.3 Cross-lingual Projection

Cross-lingual projection is a key operation in the cross-lingual co-training algorithm. In the case of event extraction, we need to project the triggers and the participant arguments from one language into the other language according to the alignment information provided by bitexts. Figure 4 shows an example of projecting an English event into the corresponding Chinese event.

Before projection:

```
<event ID="chtb_282-EV1" TYPE="Contact" SUBTYPE="Meet">
   <event_mention ID="chtb_282-EV1-1" p="1.000">
     <trigger>
        <charseq START="2259" END="2265">meeting</charseq>
     </trigger>
     <event_mention_argument ROLE="Entity" p="0.704"
        pRole="0.924"/>
       <extent>
         <charseq START="2238" END="2244">Gan Luo</charseq>
       </extent>
     </event_mention_argument>
   </event_mention>
</event>
```

After projection:

```
<event ID="chtb_282-EV1" TYPE="Contact" SUBTYPE="Meet">
   <event_mention ID="chtb_282-EV1-1" p="1.000">
     <trigger>
        <charseq START="454" END="455">会见</charseq>
     </trigger>
     <event_mention_argument ROLE="Entity" p="0.704"
        pRole="0.924"/>
       <extent>
         <charseq START="449" END="450">罗干</charseq>
       </extent>
    </event_mention_argument>
   </event_mention>
</event>
```

Figure 4. An Example of Cross-lingual Projection

4 Experiments and Results

4.1 Data and Scoring Metric

We used the ACE 2005 corpus to set up two monolingual event extraction systems, one for English, the other for Chinese.

The ACE 2005 corpus contains 560 English documents from 6 sources: newswire, broadcast news, broadcast conversations, weblogs, newsgroups and conversational telephone speech; meanwhile the corpus contains 633 Chinese documents from 3 sources: newswire, broadcast news and weblogs.

We then use 159 texts from the LDC Chinese Treebank English Parallel corpus with manual alignment for our cross-lingual bootstrapping experiments.

We define the following standards to determine the *correctness* of an event mention:

- *A trigger is correctly labeled* if its event type and offsets match a reference trigger.
- *An argument is correctly labeled* if its event type, offsets, and role match any of the reference argument mentions.

4.2 Monolingual Self-training on ACE 2005 Data

We first investigate whether our Chinese event extraction system can benefit from monolingual self-training on ACE data. We reserve 66 Chinese documents for testing purpose and set the size of seed training set to 100. For a single trial of the experiment, we *randomly* select 100 documents as training set and use the remaining documents as self-training data. For each iteration of the self-training, we keep the pool size as 50, in other words, we always pick another 50 ACE documents to self-train the system. The iteration continues until all the unlabeled ACE documents have been tagged and thus it completes one trial of the experiment. We conduct the same experiment for 100 trials and compute the average scores.

The most important motivation for us to conduct self-training experiments on ACE data is that the ACE data provide ground-truth entities and temporal expressions so that we do not have to take into account the effects of propagated errors from upstream processing such as entity extraction and temporal expression identification.

For one setting of the experiments, we set the confidence threshold to 0, in other words, we keep all the labeling results for retraining. The results are given in Figure 5 (trigger labeling) and Figure 6 (argument labeling). It shows that when the number of self-trained ACE documents reaches 450, we obtain a gain of 3.4% (F-Measure) above the baseline for trigger labeling and a gain of 1.4% for argument labeling.

For the other setting of the experiments, we set the confidence threshold to 0.8, and the results are presented in Figure 7 and Figure 8. Surprisingly, retraining on the high confidence examples does not lead to much improvement. We obtain a gain of 3.7% above the baseline for trigger labeling and 1.5% for argument labeling when the number of self-trained documents reaches 450.

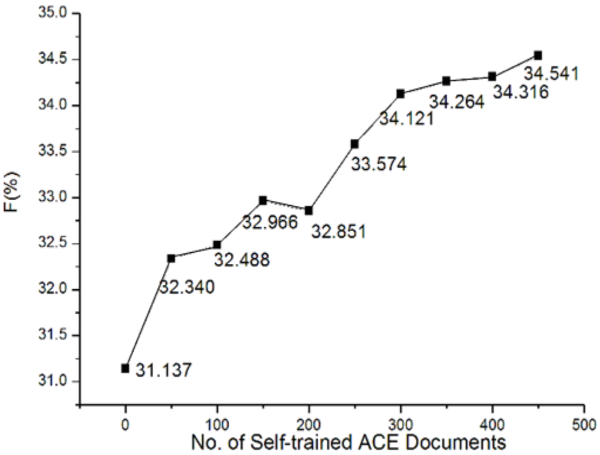

Figure 5. Self-training for trigger labeling (confidence threshold = 0)

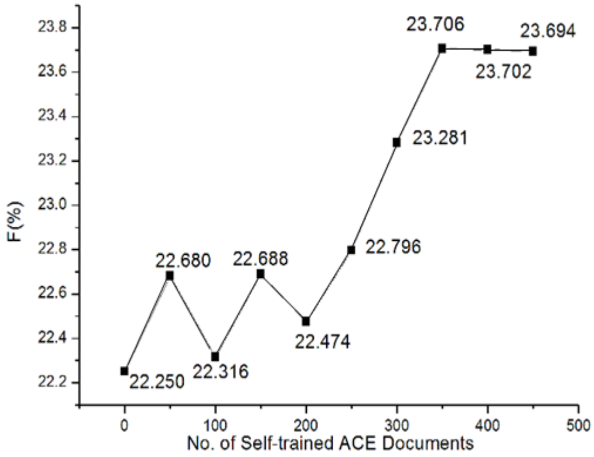

Figure 6. Self-training for argument labeling (confidence threshold= 0)

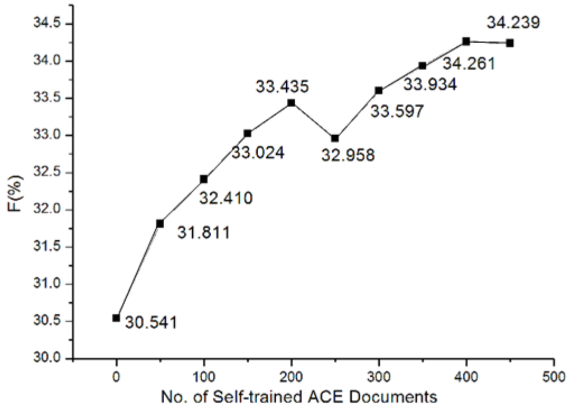

Figure 7. Self-training for trigger labeling (confidence threshold = 0.8)

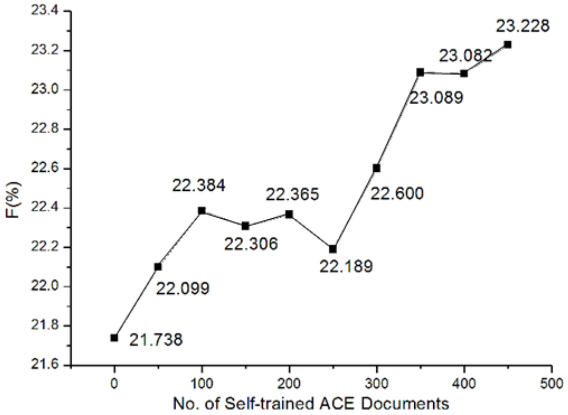

Figure 8. Self-training for argument labeling (confidence threshold = 0.8)

4.3 Cross-lingual Semi-co-training on Bitexts

The experiments in Section 4.2 show that we can obtain gain in performance by monolingual self-training on data with ground-truth entities and temporal expressions, but what if we do not have such ground-truth data, then how the errors propagated from entity extraction and temporal expression identification will affect the overall performance of our event extraction system? And if these errors are compounded in event extraction, can the cross-lingual semi-co-training alleviate the impact?

To investigate all these issues, we use 159 texts from LDC Chinese Treebank English Parallel corpus to conduct cross-lingual semi-co-training. The experimental results are summarized in Figure 9 and Figure 10.

For monolingual self-training on the bitexts, we conduct experiments exactly as section 4.2 except that the entities are tagged by the IE system and the labeling pool size is set to 20. When the number of bitexts reaches 159, we obtain a little gain of 0.4% above the baseline for trigger labeling and a loss of 0.1% below the baseline for argument labeling. The deterioration tendency of the self-training curve in Figure 10 indicates that entity extraction errors do have counteractive impacts on argument labeling.

We then conduct the cross-lingual semi-co-training experiments as follows: we set up an English event extraction system trained on a relative large training set (500 documents). For each trial of the experiment, we randomly select 100 ACE Chinese document as seed training set, and then it enters a cross-lingual semi-co-training process: for each iteration, the English system labels the English portions of the 20 bitexts and by cross-lingual projection, the labeled results are transformed into Chinese and put into the training set of Chinese system. From Figure 9 and Figure 10 we can see that when the number of bitexts reaches 159, we obtain a gain of 1.7% for trigger labeling and 0.7% for argument labeling.

We then apply a third approach to bootstrap our Chinese system: during each iteration, the Chinese system also labels the Chinese portions of the 20 bitexts. Then we combine the results from both monolingual systems using the following rules:

- If the event labeled by English system is not labeled by Chinese system, add the event to Chinese system
- If the event labeled by Chinese system is not labeled by English system, keep the event in the Chinese system
- If both systems label the same event but with different event types and arguments, select the one with higher confidence

From Figure 9 and Figure 10 we can see that this approach leads to even further improvement in performance, shown as the “Combined-labeled” curves. When the number of bitexts reaches 159, we obtain a gain of 3.1% for trigger labeling and 2.1% for argument labeling.

In order to check how robust our approach is, we conducted the Wilcoxon Matched-Pairs Signed-Ranks Test on F-measures for all these 100 trials. The results show that we can reject the hypotheses that the improvements using Cross-lingual Semi-co-training were random at a 99.99% confidence level, for both trigger labeling and argument labeling.

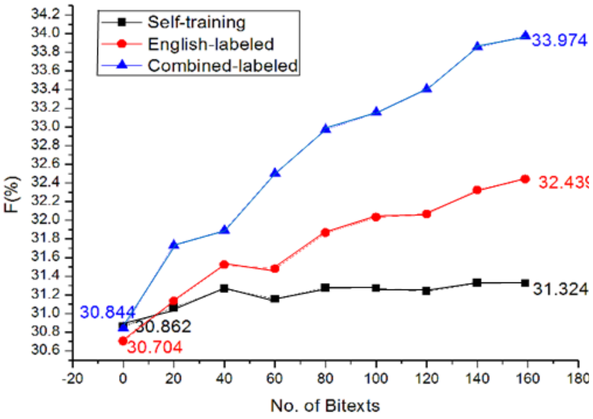

Figure 9. Self-training, and Semi-co-training (English- labeled & Combined-labeled) for Trigger Labeling

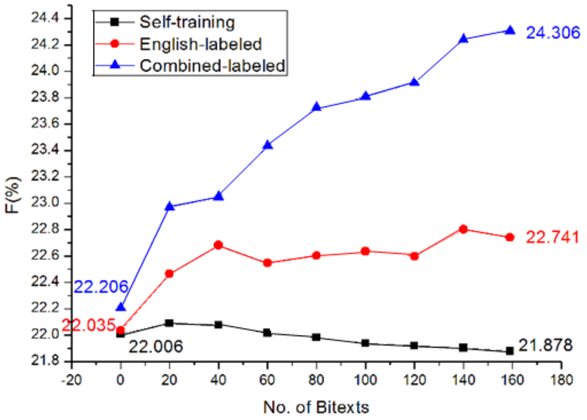

Figure 10. Self-training, and Semi-co-training (English- labeled & Combined-labeled) for Argument Labeling

5 Related Work

There is a huge literature on utilizing parallel corpus for monolingual improvement. To our knowledge, it can retrace to (Dagan et.al 1991). We apologize to those whose work is not cited due to space constraints. The work described here complements some recent research using bitexts or translation techniques as feedback to improve entity extraction. Huang and Vogel (2002) presented an effective integrated approach that can improve the extracted named entity translation dictionary and the entity annotation in a bilingual training corpus. Ji and Grishman (2007) expanded this idea of alignment consistency to the task of entity extraction in a monolingual test corpus without reference translations, and applied sophisticated inference rules to enhance both entity extraction and translation. Zitouni and Florian (2008) applied English mention detection on translated texts and added the results as additional features to improve mention detection in other languages.

In this paper we share the similar idea of importing evidences from English with richer resources to improve extraction in other languages. However, to the best of our knowledge this is the first work of incorporating cross-lingual feedback to improve the event extraction task. More importantly, it is the first attempt of combining cross-lingual projection with bootstrapping methods, which can avoid the efforts of designing sophisticated inference rules or features.

6 Conclusions and Future Work

Event extraction remains a difficult task not only because it is situated at the end of an IE pipeline and thus suffers from the errors propagated from upstream processing, but also because the labeled data are expensive and thus suffers from data scarcity. In this paper, we proposed a new co-training framework using cross-lingual information projection and demonstrate that the additional information from English system can be used to bootstrap a Chinese event extraction system.

To move a step forward, we would like to conduct experiments on cross-lingual co-training and investigate whether the two systems on both sides can benefit from each other. A main issue existing in cross-lingual co-training is that the cross-lingual projection may not be perfect due to the word alignment problem. In this paper, we used a corpus with manual alignment, but in the future we intend to investigate the effect of automatic alignment errors.

We believe that the proposed cross-lingual bootstrapping framework can also be applied to many other challenging NLP tasks such as relation extraction. However, we still need to provide a theoretical analysis of the framework.

Acknowledgments

This material is based upon work supported by the Defense Advanced Research Projects Agency under Contract No. HR0011-06-C-0023 via 27-001022, and the CUNY Research Enhancement Program and GRTI Program.

References

Avrim Blum and Tom Mitchell. 1998. Combining Labeled and Unlabeled Data with Co-training. *Proc. of the Workshop on Computational Learning Theory*. Morgan Kaufmann Publishers.

David Ahn. 2006. The stages of event extraction. Proc. COLING/ACL 2006 Workshop on Annotating and Reasoning about Time and Events. Sydney, Australia.

David Bean and Ellen Riloff. 2004. Unsupervised Learning of Contextual Role Knowledge for Coreference Resolution. *Proc. HLT-NAACL2004*. pp. 297-304. Boston, USA.

Fei Huang and Stephan Vogel. 2002. Improved Named Entity Translation and Bilingual Named Entity Extraction. *Proc. ICMI 2002*. Pittsburgh, PA, US.

Heng Ji and Ralph Grishman. 2006. Data Selection in Semi-supervised Learning for Name Tagging. *In ACL 2006 Workshop on Information Extraction Beyond the Document:48-55*. Sydney, Australia.

Heng Ji and Ralph Grishman. 2007. Collaborative Entity Extraction and Translation. *Proc. International Conference on Recent Advances in Natural Language Processing 2007*. Borovets, Bulgaria.

Ido Dagan, Alon Itai and Ulrike Schwall. 1991. Two languages are more informative than one. *Proc. ACL 1991*.

Imed Zitouni and Radu Florian. 2008. Mention Detection Crossing the Language Barrier. *Proc. EMNLP*. Honolulu, Hawaii.

Michael Collins and Yoram Singer. 1999. Unsupervised Models for Named Entity Classification. *Proc. of EMNLP/VLC-99.*

Rada Mihalcea. 2004. Co-training and self-training for word sense disambiguation. In Proceedings of the Conference on Computational Natural Language Learning (CoNLL-2004).

Ralph Grishman, David Westbrook and Adam Meyers. 2005. NYU's English ACE 2005 System Description. *Proc. ACE 2005 Evaluation Workshop*. Washington, US.

Rie Ando and Tong Zhang. 2005. A High-Performance Semi-Supervised Learning Methods for Text Chunking. *Proc. ACL2005*. pp. 1-8. Ann Arbor, USA

Scott Miller, Jethran Guinness and Alex Zamanian.2004. Name Tagging with Word Clusters and Discriminative Training. *Proc. HLT-NAACL2004*. pp. 337-342. Boston, USA

Winston Lin, Roman Yangarber and Ralph Grishman. 2003. Bootstrapping Learning of Semantic Classes from Positive and Negative Examples. *Proc. ICML-2003 Workshop on The Continuum from Labeled to Unlabeled Data*. Washington, D.C.

Zheng Chen and Heng Ji. 2009. Language Specific Issue and Feature Exploration in Chinese Event Extraction. *Proc. HLT-NAACL 2009*. Boulder, Co.