Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: (NAACL HLT), Montreal, Canada, June 2012, pp. 720--730.

Translation-Based Projection for Multilingual Coreference Resolution

Altaf Rahman and **Vincent Ng**
Human Language Technology Research Institute
University of Texas at Dallas
Richardson, TX 75083-0688
{altaf,vince}@hlt.utdallas.edu

Abstract

To build a coreference resolver for a new language, the typical approach is to first coreference-annotate documents from this target language and then train a resolver on these annotated documents using supervised learning techniques. However, the high cost associated with manually coreference-annotating documents needed by a supervised approach makes it difficult to deploy coreference technologies across a large number of natural languages. To alleviate this corpus annotation bottleneck, we examine a translation-based projection approach to multilingual coreference resolution. Experimental results on two target languages demonstrate the promise of our approach.

1 Introduction

Noun phrase (NP) coreference resolution is the task of determining which NPs (or *mentions*) refer to each real-world entity in a document. Recent years have witnessed a surge of interest in multilingual coreference resolution. For instance, the ACE 2004/2005 evaluations and SemEval-2010 Shared Task 1 have both involved coreference resolution in multiple languages. As evidenced by the participants in these evaluations, the most common approach to building a resolver for a new language is *supervised*, which involves training a resolver on coreference-annotated documents from the target language. Although supervised approaches work reasonably well, they present a challenge to deploying coreference technologies across a large number of natural languages. Specifically, for each new language of interest, one has to hire native speakers of the language to go through the labor-intensive, time-consuming process of hand-annotating a potentially large number of documents with coreference annotation before a supervised resolver can be trained.

One may argue that a potential solution to this *corpus annotation bottleneck* is to employ an *unsupervised* or *heuristic* approach to coreference resolution, especially in light of the fact that they have recently started to rival their supervised counterparts. However, by adopting these approaches, we are simply replacing the corpus annotation bottleneck by another, possibly equally serious, bottleneck, the *knowledge acquisition bottleneck*. Specifically, in these approaches, one has to employ knowledge of the target language to design coreference rules (e.g., Mitkov (1999), Poon and Domingos (2008), Raghunathan et al. (2010)) or sophisticated generative models (e.g., Haghighi and Klein (2007,2010), Ng (2008)) to combine the available knowledge sources.

One could argue that designing coreference rules and generative models may not be as time-consuming as annotating a large coreference corpus. This may be true for a well-studied language like English, where we can easily compose a rule that disallows coreference between two mentions if they disagree in number and gender, for instance. However, computing these features may not be as simple as we hope for a language like Chinese: the lack of morphology complicates the determination of number information, and the fact that most Chinese first names are used by both genders makes gender determination difficult. The difficulty in accurately computing features translates to difficulties in composing coreference rules: for example, the aforementioned rule involving gender and number agreement, as well as rules that implement traditional linguistic constraints on coreference, may no longer be accurate and desirable to have if the features involved cannot be accurately computed. Consequently, we believe that research in multilingual coreference resolution will continue to be dominated by supervised approaches.

Given the high cost of annotating data with coreference chains, it is crucial to explore methods for obtaining annotated data in a cost-effective manner. Motivated in part by this observation, we examine one such method that has recently shown promise for a variety of NLP tasks, translation-based projection, which is composed of three steps. To coreference annotate a text in the target language, we (1) machine-translate it to a resource-rich language (henceforth the *source* language); (2) automatically produce the desired linguistic annotations (which in our case are coreference annotations) on the translated text using the linguistic tool developed for the source language (which in our case is a coreference resolver) ; and (3) project the annotations from the source language to the target language.

Unlike supervised approaches, this projection approach does not require any coreference-annotated data from the target language. Equally importantly, unlike its unsupervised counterparts, this approach does not require that we have any linguistic knowledge of the target language. In fact, we have no knowledge of the target languages we employ in our evaluation. One of our goals is to examine the feasibility of building a coreference resolver for a language for which we have no coreference-annotated data *and* no linguistic knowledge of the language.

Recall that we view projection as an approach for alleviating the corpus annotation bottleneck, not as a solution to the multilingual coreference resolution problem. In fact, though rarely emphasized in previous work on applying projection, we note that projection alone cannot be used to solve multilingual NLP problems, including coreference resolution. The reason is that every language has its own idiosyncrasies with respect to linguistic properties, and projection simply cannot produce annotations capturing those properties that are specific to the target language. Our goal in this paper is to explore the extent to which projection, which does not require that we have any knowledge of the target language, can push the limits of multilingual coreference resolution. If our results indicate that projection is a promising approach, then the automatic coreference annotations it produces can be used to augment the manual annotations that capture the properties specific to the target language, thus alleviating the corpus annotation bottleneck.

2 Related Work on Projection

The idea of projecting annotations from a resource-rich language to a resource-scarce language was originally proposed by Yarowsky and Ngai (2001) and subsequently developed by others (e.g., Resnik (2004), Hwa et al. (2005)). These projection algorithms assume as input a parallel corpus for the source language and the target language. Given the recent availability of machine translation (MT) services on the Web, researchers have focused more on translated-based projection rather than acquiring a parallel corpus themselves. MT-based projection has been applied to various NLP tasks, such as part-of-speech tagging (e.g., Das and Petrov (2011)), mention detection (e.g., Zitouni and Florian (2008)), and sentiment analysis (e.g., Mihalcea et al. (2007)).

There have been two initial attempts to apply projection to create coreference-annotated data for a resource-poor language, both of which involve projecting hand-annotated coreference data from English to Romanian via a parallel corpus. Specifically, Harabagiu and Maiorano (2000) create an English-Romanian corpus by manually translating the MUC-6 corpus into Romanian and manually project the English annotations to Romanian. On the other hand, Postolache et al. (2006) apply a word alignment algorithm to project the hand-annotated English coreference chains and then manually fix the projection errors on the Romanian side. Hence, their goal is different from ours in at least two respects. First, while they employ significant knowledge of the target language to create a *clean* coreference corpus, we examine the quality of coreference-annotated data created via an entirely automatic process, determining quality by the performance of the resolver trained on the data. Second, unlike ours, neither of these attempts is at the level of defining a technology for projection annotations that can potentially be deployed across a large number of languages without coreference-annotated data.

3 Translation-Based Projection

Recall that our MT-based projection approach to coreference resolution is composed of three steps. Given a text in the target language, we (1) machine-translate the text to the source language; (2) automatically produce coreference annotations on the translated text using a coreference resolver developed for the source language; and (3) project the annotations from the source language to the target language. In this section, we employ our approach in three settings, which differ in terms of the extent to which linguistic taggers (e.g., chunkers and named entity (NE) recognizers) for the target language are available. The goal is to examine whether these linguistic taggers can be profitably exploited to improve the performance of the projection approach. Below we assume that English and French are our source and target languages, respectively.

3.1 Setting 1: No French Taggers Available

In this setting, we assume that we do not have access to any French tagger that we can exploit to improve projection. Hence, all we can do is to employ the three steps involved in the projection approach as described at the beginning of this section to create coreference-annotated data for French. Specifically, we translate a French text to an English text using GoogleTranslate[1], and create coreference chains for the translated English text using Reconcile[2] (Stoyanov et al., 2010). To project mentions from English to French, we first align the English and French words in each pair of parallel sentences, and then project the English mentions onto the French text using the alignment. However, since the alignment is noisy, the French words to which the words in the English mention are aligned may not form a contiguous text span. To fix this problem, we follow Yarowsky and Ngai (2001) and use the smallest text span that covers all the aligned French words to create the French mention.[3] We process the English mentions in the text in a left-to-right manner, as processing the mentions sequentially enables us to ensure that an English mention is not mapped to a French text span that has already been mapped to by a previously-processed English mention.[4]

To align English and French words, we trained a word alignment model using GIZA++[5] (Och and Ney, 2000) on a parallel corpus comprising the English-French section of Europarl[6] (Koehn, 2005) as well as all the French texts (and their translated English counterparts) for which we want to automatically create coreference chains. Following common practice, we stemmed the parallel corpus using the Porter stemmer (Porter, 1980) in order to reduce data sparseness. However, even with stemming, we found that many English words were not aligned to any French words by the resulting alignment model. This would prevent many English mentions from being projected to the French side, potentially harming the recall of the French coreference annotations. To improve alignment coverage, we retrained the alignment model by supplying GIZA++ with an English-French bilingual dictionary that we assembled using three online dictionary databases: OmegaWiki, Wiktionary, and Universal Dictionary. Furthermore, if a word w appears in both the English side and the French side in a pair of parallel sentences, we assume that it has the same orthographic form in both languages and hence we augment the bilingual dictionary with the entry (w, w).

Note that the use of a supervised resolver like Reconcile does *not* render our approach supervised, since we can replace it with any resolver, be it supervised, heuristic, or unsupervised. In other words, we treat the resolver built for the source language as a black box that can produce coreference annotations.

3.2 Setting 2: Mention Extractor Available

Next, we consider a comparatively less resource-scarce setting where a French mention extractor is available for identifying mentions in a French text[7], and describe how we can modify the projection approach to exploit this French mention extractor.

Given a French text we want to coreference-

[1] See `http://translate.google.com`.

[2] See `http://www.cs.utah.edu/nlp/reconcile`. We use the resolver pre-trained on the Wolverhampton corpus.

[3] Other methods for projecting mentions can be found in Postolache et al. (2006), for example.

[4] While we chose to process the mentions in a left-to-right manner, any order of processing the mentions would work.

[5] See `http://code.google.com/p/giza-pp/`.

[6] See `http://www.statmt.org/europarl/`.

[7] Mention extraction is a term used in Automatic Content Evaluation to refer to the task of determining the NPs that a coreference system should consider in the resolution process.

annotate, we first translate it to English using GoogleTranslate and align the French and English words using a French-to-English word alignment algorithm. Next, we identify the mentions in the French text using the given mention extractor, and project them onto the English text using the NP projection algorithm described in Setting 1. Finally, we run Reconcile on the resulting English mentions to generate coreference chains for the translated text, and project these chains back to the French text.

As explained before, the performance of this method is sensitive to the accuracy of the NP projection algorithm in recovering the English mentions, which in turn depends on the accuracy of the word alignment algorithm. To make this method more robust to noisy word alignment, we make a modification to it. Rather than running Reconcile on the mentions produced by the NP projection algorithm, we use Reconcile to identify the mentions directly from the translated text. After that, we create a mapping between the English mentions produced by the NP projection algorithm and those produced by Reconcile using a small set of heuristics.

Specifically, let M_P be the set of mentions identified by the NP projection algorithm and M_R be the set of mentions identified by Reconcile. For each mention m_P in M_P, we map it to a mention in M_R that shares the same right boundary. If this fails, we map it to a mention that covers its entire text span. If this fails again, we map it to a mention that has a partial overlap with it. If this still fails, we assume that m_P is not found by Reconcile and simply add m_P to M_R. As before, we process the mentions in M_P in a left-to-right manner in order to ensure that no two mentions in M_P are mapped to the same Reconcile mention. Finally, we discard all mentions in M_R that are not mapped by any mention in M_P, and present M_R to Reconcile for coreference resolution. Since we now have a 1-to-1 mapping between the Reconcile mentions and the French mentions, projecting the coreference results back to French is trivial.

It may not be immediately clear why the exploitation of the mention extractor in this setting may yield better coreference annotations than those produced in Setting 1. To see the reason, recall that one source of errors inherent in a projection approach is word alignment errors. In Setting 1, when we tried to project English mentions to the French text, word alignment errors would adversely affect the ability of the NP projection algorithm to correctly define the boundaries of the French mentions. Since coreference performance depends crucially on the ability to correctly identify mentions (Stoyanov et al., 2009), the presence of word alignment errors implies that the resulting French coreference annotations could score poorly even if the English coreference annotations produced by Reconcile were of high quality. In the current setting, on the other hand, we reduce the sensitivity of coreference performance to word alignment errors via the use of the French mention extractor to produce more accurate French mention boundaries.

3.3 Setting 3: Additional Taggers Available

Finally, we consider a setting that is the least resource-scarce of the three. We assume that in addition to a French mention extractor, we have access to other French linguistic taggers (e.g., syntactic and semantic parsers) that will allow us to generate the linguistic features needed to train a French resolver on the projected coreference annotations.

Specifically, assume that *Test* is a set of French texts we want to coreference-annotate, and *Training* is a set of French texts that is disjoint from *Test* but is drawn from the same domain as *Test*.[8] To annotate the *Test* texts, we perform the following steps. First, we employ the French mention extractor in combination with the method described in Setting 2 to automatically coreference-annotate the *Training* texts. Next, motivated by Kobdani et al. (2011), we train a French coreference resolver on the automatically coreference-annotated training texts, using the features provided by the available linguistic taggers. Finally, we apply the resolver to generate coreference chains for each *Test* text.

Two questions arise. First, is this method necessarily better than the one described in Setting 2? We hypothesize that the answer is affirmative: not only can this method exploit the knowledge about the target language provided by the additional linguistic taggers, but the resulting coreference resolver may allow us to generalize from the (noisily labeled) data and make this method more robust to the noise in-

[8] We assume that it is easy to assemble the *Training* set, since unlabeled texts are typically easy to collect in practice.

herent in the projected coreference annotations than the previously-described methods. Second, is this method necessarily better than projection via a parallel corpus? Like the first question, this is also an empirical question. Nevertheless, one reason why this method is intuitively better is that it ensures that the training and test documents are drawn from the same domain. On the other hand, when projecting annotations via a parallel corpus, we may encounter a domain mismatch problem if the parallel corpus and the test documents come from different domains, and the coreference resolver may not work well if it is trained and tested on different domains.

4 Coreference Resolution System

To train the coreference resolver employed in Setting 3 in the previous section, we need to derive linguistic features from the documents in the target language. In our experiments, we employ the coreference data sets produced as part of the SemEval-2010 shared task on Coreference Resolution in Multiple Languages. The shared task organizers have made publicly available six data sets that correspond to six European languages. Each data set comprises not only *training* and *test* documents that are coreference-annotated, but also a number of word-based linguistic features from which we derive mention-based linguistic features for training a resolver. In this section, we will describe how this resolver is trained and then applied to generate coreference chains for unseen documents.

Training the coreference classifier. As our coreference model, we train a *mention-pair* model, which is a classifier that determines whether two mentions are co-referring or not (e.g., Soon et al. (2001), Ng and Cardie (2002)).[9] Each instance $\boldsymbol{i}(m_j, m_k)$ corresponds to m_j (a candidate antecedent) and m_k (the mention to be resolved), and is represented by a set of 23 features shown in Table 1. As we can see, each feature is either *relational*, capturing the relation between m_j and m_k, or *non-relational*, capturing the linguistic property of m_k. The possible values of a relational feature (except LEXICAL) are **C** (compatible), **I** (incompatible), and **NA** (the comparison cannot be made due to missing data). For a non-relational feature, we refer the reader to the data sets for the list of possible values.[10]

We follow Soon et al.'s (2001) method for creating training instances. Specifically, we create (1) a positive instance for each anaphoric mention m_k and its closest antecedent m_j; and (2) a negative instance for m_k paired with each of the intervening mentions, $m_{j+1}, m_{j+2}, \ldots, m_{k-1}$. The classification associated with a training instance is either positive or negative, depending on whether the two mentions are coreferent in the associated text. To train the classifier, we use SVM^{light} (Joachims, 1999).

Applying the classifier to a test text. After training, the classifier is used to identify an antecedent for a mention in a test text. Specifically, each mention, m_k, is compared to each preceding mention, m_j, from right to left, and m_j is selected as the antecedent of m_k if the pair is classified as coreferent. The process terminates as soon as an antecedent is found for m_k or the beginning of the text is reached.

5 Evaluation

We evaluate our MT-based projection approach for each of the three settings described in Section 3.

5.1 Experimental Setup

Data sets. We use the Spanish and Italian data sets from the SemEval-2010 shared task on Coreference Resolution in Multiple Languages.[11] Each data set is composed of a training set and a test set. Statistics of these data sets are shown in Table 2.

	Spanish	Italian
Training Set Statistics		
number of mentions	78779	24853
number of non-singleton clusters	48681	18376
number of singleton clusters	37336	15984
Test Set Statistics		
number of mentions	14133	13394
number of non-singleton clusters	8789	9520
number singleton clusters	6737	8288

Table 2: Statistics of the data sets.

[9]Note that any supervised coreference model can be used, such as an entity-mention model (e.g., Luo et al. (2004), Yang et al. (2008)) or a ranking model (e.g., Denis and Baldridge (2008), Rahman and Ng (2009)).

[10]The data sets can be downloaded from `http://stel.ub.edu/semeval2010-coref/datasets`.

[11]Note, however, that our approach is equally applicable to other languages evaluated in the shared task.

Features describing m_k, the mention to be resolved		
1	NUM_WORDS	the number of words in m_k
2	COARSE_POS	the coarse POS of m_k (see "PoS" in Recasens et al. (2010))
3	FINE_POS	the fine-grained POS of m_k (see "PoS type" in Recasens et al. (2010))
4	NE	the named entity tag of m_k if m_k is a named entity; else NA
5	SR	the semantic role of m_k
6	GRAMROLE	the grammatical role of m_k
7	NUMBER	the number of m_k
8	GENDER	the gender of m_k
9	PERSON	the person of m_k (e.g., first, second, third) if it is pronominal; else NA
Features describing the relationship between m_j, a candidate antecedent and m_k, the mention to be resolved		
10	CS_STR_MATCH	determines whether the mentions are the same string
11	CI_STR_MATCH	same as feature 10, except that case differences are ignored
12	CS_SUBSTR_MATCH	determines whether one mention is a substring of the other
13	CI_SUBSTR_MATCH	same as feature 12, except that case differences are ignored
14	NUMBER_MATCH	determines whether the mentions agree in number
15	GENDER_MATCH	determines whether the mentions agree in gender
16	COARSE_POS_MATCH	determines whether the mentions have the same coarse POS tag
17	FINE_POS_MATCH	determines whether the mentions have the same fine-grained POS tag
18	ROLE_MATCH	determines whether the mentions have the same grammatical role
19	NE_MATCH	determines whether both are NEs and have the same NE type
20	SR_MATCH	determines whether the mentions have the same semantic role
21	ALIAS	determines whether one mention is an abbreviation or an acronym of the other
22	PERSON_MATCH	determines whether both mentions are pronominal and have the same person
23	LEXICAL	the concatenation of the heads of the two mentions

Table 1: Feature set for coreference resolution.

Scoring programs. To score the output of a coreference resolver, we employ four scoring programs, MUC (Vilain et al., 1995), B^3 (Bagga and Baldwin, 1998), ϕ_3-CEAF (Luo, 2005), and BLANC (Recasens and Hovy, 2011), which were downloaded from the shared task website (see Footnote 10).

Gold-standard versus regular settings. The format of each data set follows that of a typical CoNLL shared task data set. In other words, each row corresponds to a word in a document; moreover, all but the last column contain the linguistic features computed for the words, and the last column stores the coreference information. Some of the features were computed via automatic means, but some were extracted from human annotations. Given this distinction, the shared task organizers defined two evaluation settings: in the *regular* setting, only the columns that were computed automatically can be used to derive coreference features for classifier training, and results should be reported on system mentions; on the other hand, in the *gold-standard* setting, only the columns that were extracted from human annotations can be used to derive coreference features, and results should be reported on true mentions. We will present results corresponding to both settings. Note that these two settings should not be confused with the three settings described in Section 3.

Mention extraction. Recall that Settings 2 and 3 both assume the availability of a mention extractor for extracting mentions in the target language. In our experiments, we extract mentions using two methods. First, we assume the availability of an oracle mention extractor that will enable us to extract *true mentions* (i.e., gold-standard mentions) directly from the test texts. Second, we employ simple heuristics to automatically extract *system mentions*.

Since coreference performance is sensitive to the accuracy of mention extraction (Stoyanov et al., 2009), we experiment with several heuristic methods for extracting system mentions for both Spanish and Italian. According to our cross-validation experiments on the training data, the best heuristic for extracting Spanish mentions is different from that for extracting Italian mentions. Specifically, for

Spanish, the best heuristic method operates as follows. First, it extracts all the syntactic heads (i.e., the word tokens whose gold dependency labels are SUBJ, PRED, or GMOD). Second, for each syntactic head, it identifies the smallest text span containing the head and all of its dependents, and creates a mention from this text span. For Italian, on the other hand, the best heuristic simply involves creating one mention for each gold NE. The reason why this simple heuristic works well is that most of the Italian mentions are NEs, owing to the fact that abstract NPs and pronouns are also annotated as NEs in the Italian data set. When evaluated on the test set, the heuristic-based mention extractor achieves F-scores of 80.2 (78.4 recall, 82.1 precision) for Spanish and 92.3 (85.9 recall, 99.6 precision) for Italian.

5.2 Results and Discussion

5.2.1 Supervised Results

Our supervised systems. While our MT-based projection approach is unsupervised (i.e., it does not rely on any coreference annotations from the target language), it would be informative to see the performance of the *supervised* resolvers, since their performance can be viewed as a crude upper bound on the performance of our unsupervised systems. Specifically, we train a mention-pair model on the training set using the 23 features shown in Table 1 and SVM^{light} as the underlying learning algorithm[12], and apply the resulting model in combination with Soon et al.'s clustering algorithm (see Section 4) to generate coreference chains for the test texts.

Results on the test sets, reported in terms of recall (R), precision (P), and F-score (F) computed by the four coreference scorers, are shown in the first two rows of Table 3 (Spanish) and Table 4 (Italian). For convenience, we summarize a system's performance using a single number, which is shown in the last column (Average) and is obtained by taking a simple average of the F-scores of the four scorers. More specifically, row 1, which is marked with a 'G', and row 2, which is marked with a 'R', show the results obtained under the *gold-standard* setting and the *regular* setting, respectively.

As we can see, under the gold-standard setting, the supervised resolver achieves an average F-score of 66.1 (Spanish) and 65.9 (Italian). Not surprisingly, under the regular setting, its average F-score drops statistically significantly[13] to 54.6 (Spanish) and 63.4 (Italian).[14]

Best systems in the shared task. To determine whether the upper bounds established by our supervised systems are reasonable, we show the results of the best-performing resolvers participating in the shared task for both languages under the gold-standard and regular settings in rows 3 and 4 of Tables 3 and 4. Since none of the participating systems achieved the best score over all four scorers, we report the performance of the system that has the highest average F-score. According to the shared task website, TANL-1 (Attardi et al., 2010) achieved the best average F-score in the regular setting for Spanish, whereas SUCRE (Kobdani and Schütze, 2010) outperformed others in the remaining settings.

Comparing these best shared task results with our supervised results in rows 1 and 2, we see that our average F-score for Spanish/Gold is worse than its shared task counterpart by 0.7 points, but otherwise our system outperforms in other settings w.r.t. average F-score, specifically by 5.0 points for Spanish/Regular (due to a better MUC F-score), by 3.4–4.7 points for Italian (due to better CEAF, B^3, and BLANC scores). Overall, these results suggest that the scores achieved by our systems are at least as competitive as the best shared task scores.

5.2.2 Unsupervised Results

Next, we evaluate our projection algorithm.

Setting 1. Results of our approach, when applied in Setting 1, are shown in row 5 of Tables 3 and 4. Given that it has to operate under the severe condition where no linguistic taggers are available for the target language, it is perhaps not surprising to see that its performance is significantly worse than that of its supervised counterparts.

Setting 2. Recall that this setting is less resource-scarce than Setting 1 in that a mention extractor for

[12] All SVM learning parameters in this and other experiments in this paper are set to their default values.

[13] All significance test results in this paper are obtained using one-way ANOVA, with p set to 0.05.

[14] Separately, we determined whether the performance drop in the regular setting is due to the use of automatically computed features or the use of system mentions, and found that the latter was almost entirely responsible for the drop.

	Approach	CEAF R	CEAF P	CEAF F	MUC R	MUC P	MUC F	B^3 R	B^3 P	B^3 F	BLANC R	BLANC P	BLANC F	Average F
1	Supervised (G)	68.8	68.8	68.8	58.2	52.6	55.3	76.5	75.1	75.8	62.9	66.1	64.3	66.1
2	Supervised (R)	57.4	60.1	58.8	41.0	46.3	43.5	57.6	64.8	61.0	53.9	65.0	55.2	54.6
3	Shared task best (G)	69.8	69.8	69.8	52.7	58.3	55.3	75.8	79.0	77.4	67.3	62.5	64.5	66.8
4	Shared task best (R)	58.6	60.0	59.3	14.0	48.4	21.7	56.6	79.0	66.0	51.4	74.7	51.4	49.6
5	Setting 1	35.9	52.9	42.8	10.8	48.7	17.7	30.5	63.9	41.3	51.2	72.6	48.7	37.6
6	Setting 2 (True)	65.6	65.6	65.6	16.8	64.7	26.7	64.3	96.9	77.3	52.8	78.8	54.6	56.1
7	Setting 2 (System)	53.2	55.7	54.4	13.4	58.5	21.8	49.8	79.7	61.3	50.7	75.5	49.5	46.8
8	Setting 3 (G)	65.9	65.9	65.9	48.1	45.2	46.6	72.3	72.6	72.5	60.1	61.4	60.7	61.4
9	Setting 3 (R)	55.3	55.3	55.3	34.1	41.6	37.5	55.1	63.6	59.0	53.8	62.1	54.9	51.7

Table 3: Results for Spanish

	Approach	CEAF R	CEAF P	CEAF F	MUC R	MUC P	MUC F	B^3 R	B^3 P	B^3 F	BLANC R	BLANC P	BLANC F	Average F
1	Supervised (G)	74.5	74.5	74.5	31.8	67.4	43.2	74.4	93.6	82.9	58.4	79.6	62.9	65.9
2	Supervised (R)	73.7	74.3	74.0	31.9	68.0	43.4	60.8	92.5	73.3	58.4	79.6	62.9	63.4
3	Shared task best (G)	66.0	66.0	66.0	48.1	42.3	45.0	76.7	76.9	76.8	54.8	63.5	56.9	61.2
4	Shared task best (R)	57.1	66.2	61.3	50.1	50.7	50.4	63.6	79.2	70.6	55.2	68.3	57.7	60.0
5	Setting 1	17.0	26.0	20.6	8.1	28.5	12.6	14.1	30.5	19.3	50.1	62.9	32.9	21.4
6	Setting 2 (True)	73.3	73.3	73.3	14.2	60.6	23.0	72.9	96.8	83.2	51.9	77.9	53.2	58.2
7	Setting 2 (System)	60.4	70.1	64.9	17.2	68.2	27.5	59.3	97.1	73.6	52.0	82.9	53.4	54.9
8	Setting 3 (G)	64.3	64.3	64.3	28.3	63.3	39.1	65.3	87.4	74.8	55.1	74.7	57.5	58.9
9	Setting 3 (R)	61.1	62.9	61.9	29.5	63.2	40.2	60.3	84.1	70.2	55.3	72.9	58.3	57.7

Table 4: Results for Italian

the target language is available. Results of our algorithm, when operating under Setting 2 using true mentions and system mentions, are shown in rows 6 and 7 of Tables 3 and 4, respectively. In comparison to the results for Setting 1, we see that the F-scores obtained under Setting 2 increase significantly, regardless of (1) the scoring programs and (2) whether true mentions or system mentions are used. These results provide evidence for our earlier hypothesis that our projection algorithm can profitably exploit the linguistic knowledge about the target language that is available to it. In particular, the mention extractor helps make our approach less sensitive to word alignment and NP projection errors.

In comparison to our supervised results in rows 1 and 2, our algorithm still lags behind by about 8–10 points in average F-score. However, this should not be surprising, since our algorithm is unsupervised. Looking closer at the results, we can see that the performance lag by our approach can be attributed to its lower recall: in general, the lag in MUC recall appears to be more acute than that in B^3 and CEAF recall. Since MUC only scores non-singleton clusters wheres B^3 and CEAF score both singleton and non-singleton clusters, these results suggest that our approach is better at identifying singleton clusters than recovering coreference links.

Setting 3. Finally, we evaluate our approach in a setting where it has access to all the information available to our supervised resolvers, except for the gold-standard coreference annotations on the training sets. Specifically, our approach uses projected coreference annotations to train a resolver on the training texts, whereas the supervised resolvers do so using gold-standard annotations.

Comparing Settings 2 and 3 with respect to true mentions (rows 6 and 8 of Tables 3 and 4), we see mixed results. According to MUC and BLANC, the resolvers in Setting 3 are significantly better than those in Setting 2 for both languages. According to B^3, the resolvers in Setting 2 are significantly better than those in Setting 3 for both languages. According to CEAF, the Spanish resolvers in Setting 3 are significantly better than their counterparts in Setting 2, but the opposite is true for the Italian resolvers.

To understand these somewhat contradictory performance trends, let us first note that the dramatic increase in the MUC F-score can be attributed to large gains in MUC recall. This suggests that the classifiers being trained in Setting 3 have enabled the discovery of additional coreference links. In other words, there are benefits to be obtained just by learning over noisy coreference annotations, a result that we believe is quite interesting. However, not all of these newly discovered coreference links are correct. The fact that some scoring programs (e.g., B^3) are more sensitive to spurious coreference links than the others (e.g., MUC) explains these mixed results.

Nevertheless, according to average F-score, the resolvers in Setting 3 perform significantly better than those in Setting 2 for both languages: F-score increases by 5.3 points for Spanish and 0.7 points for Italian. Similar trends can be observed when comparing the two settings w.r.t. system mentions (rows 7 and 9 of Tables 3 and 4): F-score increases by 4.9 points for Spanish and 2.8 points for Italian.

While our Setting 3 results still underperform the supervised results in rows 1 and 2, we can see that they achieve 93–94% of the average F-scores of the supervised Spanish resolvers and 89–91% of the average F-scores of the supervised Italian resolvers. Importantly, recall that our approach achieves this level of performance without relying on any gold-standard coreference annotations in Spanish and Italian, and we believe that these results demonstrate the promise of our MT-based projection approach.

Since these results suggest that our approach cannot be successfully applied without MT services, a parallel corpus for learning a word alignment model, and a mention extractor for the target language, a natural question is: to what extent do these requirements limit the applicability of our approach? While it is the case that our approach cannot be applied to a truly resource-scarce language, it can be applied to the numerous Indian and East European languages for which the aforementioned requirements are satisfied but coreference-annotated data is not readily available.

6 Conclusions and Future Work

We explored the under-investigated yet challenging task of performing coreference resolution for a language for which we have no coreference-annotated data and no linguistic knowledge of the language. Our translation-based projection approach has the flexibility to exploit any available knowledge about the target language. In experiments with Spanish and Italian, we obtained promising results: our approach achieved around 90% of the performance of a supervised resolver when only a mention extractor for the target language was available. We believe that this approach has the potential to allow coreference technologies to be deployed across a larger number of languages than is currently possible, and that this is just the beginning of a new line of work.

To gain additional insights into our approach, we plan to pursue several directions. First, we will isolate the impact of each factor that adversely affects its performance, including errors in projection, translation, and coreference resolution in the resource-rich language. Second, we will perform an empirical comparison of two approaches to projecting coreference annotations, our translation-based approach and Camargo de Souza and Orasan's (2011) approach, where annotations are projected via a parallel corpus. Third, rather than translate from the target to the source language, we will examine whether it is better to translate all the coreference-annotated data available in the source language to the target language, and train a coreference model for the target language on the translated data. Fourth, since the success of our projection approach depends heavily on the accuracies of machine translation as well as coreference resolution in the source language, we will determine whether their accuracies can be improved via an ensemble approach, where we employ multiple MT engines and multiple coreference resolvers. Finally, we plan to employ our approach to alleviate the corpus-annotation bottleneck, specifically by using the annotated data it produces to augment the manual coreference annotations that capture the specific properties of the target language.

Acknowledgments

We thank the three anonymous reviewers for their detailed and insightful comments on an earlier draft of the paper. This work was supported in part by NSF Grants IIS-0812261 and IIS-1147644. Any opinions, findings, or conclusions expressed in this paper are those of the authors and do not necessarily reflect the views or official policies of NSF.

References

Giuseppe Attardi, Maria Simi, and Stefano Dei Rossi. 2010. TANL-1: Coreference resolution by parse analysis and similarity clustering. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 108–111.

Amit Bagga and Breck Baldwin. 1998. Entity-based cross-document coreferencing using the vector space model. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics*, pages 79–85.

Jennifer Camargo de Souza and Constantine Orasan. 2011. Can projected chains in parallel corpora help coreference resolution? In *Anaphora Processing and Applications*, pages 59–69. Springer.

Dipanjan Das and Slav Petrov. 2011. Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 600–609.

Pascal Denis and Jason Baldridge. 2008. Specialized models and ranking for coreference resolution. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 660–669.

Aria Haghighi and Dan Klein. 2007. Unsupervised coreference resolution in a nonparametric bayesian model. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 848–855.

Aria Haghighi and Dan Klein. 2010. Coreference resolution in a modular, entity-centered model. In *Proceedings of Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 385–393.

Sanda Harabagiu and Steven Maiorano. 2000. Multilingual coreference resolution. In *Proceedings of the Sixth Applied Natural Language Processing Conference*, pages 142–149.

Rebecca Hwa, Philip Resnik, Amy Weinberg, Clara Cabezas, and Okan Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural Language Engineering*, 11(3):311–325.

Thorsten Joachims. 1999. Making large-scale SVM learning practical. In Bernhard Schölkopf, Christopher Burges, and Alexander Smola, editors, *Advances in Kernel Methods – Support Vector Learning*, pages 44–56. MIT Press, Cambridge, MA.

Hamidreza Kobdani and Hinrich Schütze. 2010. SUCRE: A modular system for coreference resolution. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 92–95.

Hamidreza Kobdani, Hinrich Schütze, Michael Schiehlen, and Hans Kamp. 2011. Bootstrapping coreference resolution using word associations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 783–792.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *Proceedings of MT Summit X*.

Xiaoqiang Luo, Abe Ittycheriah, Hongyan Jing, Nanda Kambhatla, and Salim Roukos. 2004. A mention-synchronous coreference resolution algorithm based on the Bell tree. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics*, pages 135–142.

Xiaoqiang Luo. 2005. On coreference resolution performance metrics. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 25–32.

Rada Mihalcea, Carmen Banea, and Janyce Wiebe. 2007. Learning multilingual subjective language via cross-lingual projections. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 976–983..

Ruslan Mitkov. 1999. Multilingual anaphora resolution. *Machine Translation*, 14(3–4):281–299.

Vincent Ng and Claire Cardie. 2002. Improving machine learning approaches to coreference resolution. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 104–111.

Vincent Ng. 2008. Unsupervised models for coreference resolution. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 640–649.

Franz Josef Och and Hermann Ney. 2000. Improved statistical alignment models. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*.

Hoifung Poon and Pedro Domingos. 2008. Joint unsupervised coreference resolution with Markov Logic. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 650–659.

Martin F. Porter. 1980. An algorithm for suffix stripping. *Program*, 14(3):130–137.

Oana Postolache, Dan Cristea, and Constantin Orasan. 2006. Transferring coreference chains through word alignment. In *Proceedings of the Fifth International Conference on Language Resources and Evaluation*, pages 889–892.

Karthik Raghunathan, Heeyoung Lee, Sudarshan Rangarajan, Nate Chambers, Mihai Surdeanu, Dan Jurafsky, and Christopher Manning. 2010. A multi-pass sieve for coreference resolution. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 492–501.

Altaf Rahman and Vincent Ng. 2009. Supervised models for coreference resolution. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 968–977.

Marta Recasens and Eduard Hovy. 2011. BLANC: Implementing the Rand Index for coreference evaluation. *Natural Language Engineering*, 17(4):485–510.

Marta Recasens, Lluís Màrquez, Emili Sapena, M. Antònia Martí, Mariona Taulé, Véronique Hoste, Massimo Poesio, and Yannick Versley. 2010. Semeval-2010 task 1: Coreference resolution in multiple languages. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 1–8.

Philip Resnik. 2004. Exploiting hidden meanings: Using bilingual text for monolingual annotation. In *Proceedings of the 5th International Conference on Computational Linguistics and Intelligent Text Processing*, pages 283–299.

Wee Meng Soon, Hwee Tou Ng, and Daniel Chung Yong Lim. 2001. A machine learning approach to coreference resolution of noun phrases. *Computational Linguistics*, 27(4):521–544.

Veselin Stoyanov, Nathan Gilbert, Claire Cardie, and Ellen Riloff. 2009. Conundrums in noun phrase coreference resolution: Making sense of the state-of-the-art. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 656–664.

Veselin Stoyanov, Claire Cardie, Nathan Gilbert, Ellen Riloff, David Buttler, and David Hysom. 2010. Coreference resolution with reconcile. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 156–161.

Marc Vilain, John Burger, John Aberdeen, Dennis Connolly, and Lynette Hirschman. 1995. A model-theoretic coreference scoring scheme. In *Proceedings of the Sixth Message Understanding Conference*, pages 45–52.

Xiaofeng Yang, Jian Su, Jun Lang, Chew Lim Tan, and Sheng Li. 2008. An entity-mention model for coreference resolution with inductive logic programming. In *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 843–851.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 200–207.

Imed Zitouni and Radu Florian. 2008. Mention detection crossing the language barrier. In *Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing*, pages 600–609.