Facing the most difficult case of Semantic Role Labeling: A collaboration of word embeddings and co-training

Quynh Ngoc Thi Do[1], **Steven Bethard**[2], **Marie-Francine Moens**[1]
[1]Katholieke Universiteit Leuven, Belgium
[2]University of Arizona, United States
quynhngocthi.do@cs.kuleuven.be
bethard@email.arizona.edu
sien.moens@cs.kuleuven.be

Abstract

We present a successful collaboration of word embeddings and co-training to tackle in the most difficult test case of semantic role labeling: predicting out-of-domain and unseen semantic frames. Despite the fact that co-training is a successful traditional semi-supervised method, its application in SRL is very limited. In this work, co-training is used together with word embeddings to improve the performance of a system trained on CoNLL 2009 training dataset. We also introduce a semantic role labeling system with a simple learning architecture and effective inference that is easily adaptable to semi-supervised settings with new training data and/or new features. On the out-of-domain testing set of the standard benchmark CoNLL 2009 data our simple approach achieves high performance and improves state-of-the-art results.

1 Introduction

Semantic role labeling (SRL) is an essential natural language processing (NLP) task that identifies the relations between a predicate and its arguments in a given sentence. Intuitively, it aims at answering the questions of "Who did What to Whom, and How, When and Where?" in text. For example, the processing of the sentence "He bought tons of roses yesterday" should result in the identification of a "buying" event corresponding to the predicate "bought" with three arguments including "he" as the *Agent (A0)*, "tons of roses" as the *Thing being bought (A1)*, and "yesterday" as the *Time (AM-TMP)* arguments. Traditional SRL systems have concentrated on supervised learning from several manually-built semantic corpora, (e.g., FrameNet (Baker et al., 1998) and PropBank (Palmer et al., 2005)). One important limitation of supervised approaches is that they depend heavily on the accuracy, coverage and labeling scheme of the labeled corpus. When the training and the testing data are in different domains, the linguistic patterns and their distributions in the testing domain are different from the ones observed in the training data, resulting in a considerable performance drop. Developing more manually-built semantic corpora is expensive and requires huge human efforts. Thus, exploiting large unlabeled datasets by semi-supervised or unsupervised approaches is a promising solution.

Our contribution in this paper is two-fold: First, we introduce a SRL system with a simple learning architecture and effective inference that is easily adaptable to new training data or new features in semi-supervised settings. Second, we present a semi-supervised approach that is a combination of using word embeddings as extra features and using a variant of a co-training algorithm to create new training data facing the most difficult cases of SRL: improving the SRL system trained on a relatively large training dataset (CoNLL 2009) when working with the "out-of-domain" and especially "unseen" semantic frames. Although co-training is a successful traditional semi-supervised method, its application to SRL is very limited. To our knowledge, there has been no successful case of co-training applied to a large amount of training data in the literature.

In this work, we first enrich a traditional feature set of SRL by the distributional word representations induced from a large unlabeled corpus. Then, we divide the feature set into two different sets with one referring to the semantic or meaning information of the argument candidate and one referring to its

Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, pages 1275–1284, Osaka, Japan, December 11-17 2016.

syntactic information based on the dependency structure. Two local classifiers are then trained, one on each of the two feature sets. Next, we label unannotated data from the same domain as the target data with these classifiers. Finally, a global classifier is trained with selected newly labeled instances and the joint feature set of the two local classifiers. Our experiments show that the combination of using distributional word representations and a co-training strategy effectively improves SRL in the challenging out-of-domain scenario. It outperforms using only word embeddings or co-training especially in unseen frames.

The rest of the paper is structured as follows. Section 2 discusses related works. We describe our SRL system and our methodology in Section 3 and Section 4 respectively. Our experiment is presented in Section 5 and Section 6 and finally we conclude in Section 7.

2 Related Work

In traditional supervised approaches, SRL is modeled as a pipeline of predicate identification, predicate disambiguation, argument identification, and argument classification steps. Hand-engineered linguistically-motivated feature templates represent the semantic structure employed to train classifiers for each step. It is common among the state-of-the-art systems to train a global reranker on top of the local classifiers to improve performance (Toutanova et al., 2005; Björkelund et al., 2010; Roth and Lapata, 2016). SRL models have also been trained using graphical models (Täckström et al., 2015) and neural networks (Collobert et al., 2011; FitzGerald et al., 2015). Some systems have applied a set of structural constraints to the argument classification sub-task, such as avoiding overlapping arguments and repeated core roles, and enforced these constraints with integer linear programming (ILP) (Punyakanok et al., 2008) or a dynamic program (Täckström et al., 2015).

Regarding leveraging unlabeled data, semi-supervised methods have been proposed to reduce human annotation efforts. He and Gildea (2006) investigate the possibility of a weakly supervised approach by using self-training and co-training for unseen frames of SRL. They separate the headword and path as the two views for co-training, but could not show a clear performance improvement. The sources of the problem appeared to be the big gap in performance between the headword and path feature sets and the complexity of the task. Some other works show slight improvements of using co-training for SRL when there is a limited number of labeled data (Lee et al., 2007; Samad Zadeh Kaljahi and Baba, 2011). Fürstenau and Lapata (2012) find novel instances for classifier training based on their similarity to manually labeled seed instances. This strategy is formalized via a graph alignment problem.

Recently, there has been interest in distributional word representations for natural language processing. Such representations are typically learned from a large corpus using neural networks (e.g., Weston et al. (2008)), probabilistic graphical models (e.g., Deschacht et al. (2012)) or term-cooccurrence statistics (e.g., Turney and Pantel (2010)) by capturing the contexts in which the words appear. Often words from the vocabulary or phrases are mapped to vectors of real numbers in a low dimensional continuous space resulting in so-called word embeddings. Deschacht et al. (2012) employ distributed representations for each argument candidate as extra features when training a supervised SRL. Roth and Woodsend (2014) propose to use the compositional representations such as interaction of predicate and argument, dependency path and the full argument span to improve a state-of-the-art SRL system.

3 A Semantic Role Labeling System for Semi-Supervised Approaches

In this section, we introduce a semantic role labeling system designed for semi-supervised settings. The system has a simple training strategy with local classifiers for different steps in SRL pipeline. Instead of training a global reranker on top of the local classifiers to improve performance as in other common pipeline-based state-of-the-art systems like (Toutanova et al., 2005; Björkelund et al., 2010; Roth and Lapata, 2016), we propose a novel joint inference technique that works across the argument identification and argument classification steps. That makes the SRL training simple (no reranker, only local classifiers) and therefore easily adaptable to new training examples or new features. We will show later in the experiment that the joint inference gives us comparable results to a reranker.

Following prior work (Toutanova et al., 2005; Björkelund et al., 2010; Roth and Lapata, 2016), our system consists of four modules: (1) A predicate identification (PI) module which detects whether a

word is a predicate. (2) A predicate disambiguation (PD) module which labels a predicate with a sense, where we train a local classifier for each predicate lemma. (3) An argument identification (AI) module that recognizes argument words, where given a predicate p, each word w_i in p's sentence is assigned a probability $P^{AI}(p, w_i)$ of being p's argument. (4) An argument classification (AC) module that assigns labels to argument words, where given a predicate p and a set of labels $\mathbf{L}$ – PropBank semantic role label set in this work, each word w_i is assigned probabilities $P^{AC}(p, w_i, L_j)$ to receive $L_j \in \mathbf{L}$ as semantic label.

We employ the features proposed by Björkelund et al. (2010) as the basic feature set. All of the local classifiers are trained using L2-regularized logistic regression. For multiclass problems, we use the one-vs-rest strategy.

At inference time, the local classifier predictions are merged using integer linear programming (ILP). In most of the prior work, ILP was only used for AC inference. However, this approach limits the interaction of AI and AC when making decisions. In another approach, Srikumar and Roth (2011) introduce a simple approach to joint inference over AI and AC allowing the two argument sub-tasks to support each other. Their local AC classifier has an empty label which indicates that the candidate is, in fact, not an argument. This forces AC module to learn also the argument identification and is in contrast with our approach in which the tasks of AI and AC classifiers are completely separated leading to a simpler AC learning. Their inference is formularized as an ILP problem that mazimizes the sum of local prediction scores over AI and AC. The authors then enforce consistency constraints between the identifier and the argument classifier predictions – the identifier should predict that a candidate is an argument if and only if the argument classifier does not predict the empty label. In this paper, we propose a novel ILP inference formulation in which the interaction of AI and AC is exploited and emphasized not only in the consistency constraint but also in the objective function.

Joint Argument Inference For each predicate, we perform joint inference over the AI and AC steps for all the words in the sentence. Given a predicate p and set of words in the sentence $w_1, w_2, ..., w_n$, each word is determined as either non-argument or as one of the semantic roles via an ILP formulation. Let $\mathbf{U}$ be the set of binary indicator variables corresponding to the decision whether w_i is a non-argument. Specifically, $u_i = 1$ if w_i is not an argument and $u_i = 0$ otherwise. Let $\mathbf{V}$ be the set of binary indicator variables corresponding to the decision whether w_i is a certain semantic role. Specifically, $v_{ij} = 1$ if w_i is assigned label L_j and $v_{ij} = 0$ otherwise.

A simple joint inference to maximize the sum of local prediction scores over AI and AC (see Srikumar and Roth (2011)) would be[1]:

$$\sum_{i=1}^{n} \left[(1 - P^{AI}(p, w_i)) * u_i + P^{AI}(p, w_i) * (1 - u_i) + \sum_{j=1}^{|\mathbf{L}|} \left(v_{ij} * P^{AC}(p, w_i, L_j) + (1 - v_{ij}) * (1 - P^{AC}(p, w_i, L_j)) \right) \right]$$

In this paper, to exploit more effectively the interaction between AI and AC, we propose to maximize the objective function:

$$\sum_{i=1}^{n} \left\{ (1 - P^{AI}(p, w_i)) * u_i * \lambda + P^{AI}(p, w_i) * \left[\sum_{j=1}^{|\mathbf{L}|} \left(v_{ij} * P^{AC}(p, w_i, L_j) + (1 - v_{ij}) * (1 - P^{AC}(p, w_i, L_j)) \right) \right] \right\}$$

Subject to:

$$\forall i : u_i + \sum_{j=1}^{|L|} v_{ij} = 1 \tag{1}$$

$$\forall j : if\ L_j\ is\ core\ role : \sum_{i=1}^{n} v_{ij} \leq 1 \tag{2}$$

[1]Note that we can not use exactly the same objective function as in Srikumar and Roth (2011) because our AC module does not produce the empty label.

λ is a parameter between 0 and 1 controlling the balance of recall and precision. If λ is small, the importance of predicting correct non-argument words is lowered, so more words are considered as argument candidates leading to the increase of recall and decrease of precision. In contrast, if λ is large, precision goes up and recall goes down.

Constraint 1 forces the system to assign either only one semantic role or a non-argument label to each word. Meanwhile, constraint 2 restricts the core roles (A0, A1, A2, A3, A4, A5, AA) to appear no more than once.

In the experiment, we will compare the performance of our proposed inference to the approaches using a reranker as in Björkelund et al. (2010) and the above simple joint inference of AI and AC.

4 Semi-supervised approach

4.1 Problem

In SRL, classifiers need linguistic clues to make a correct prediction. Two different types of clues are frequently distinguished: (1) Semantic or meaning clues. For example, when classifying arguments for the predicate "sleep", if the classifier sees the word "bed", assigning the role "AM-LOC" to the word seems reasonable. (2) Syntactic or word order clues. In the above example, if the classifier sees a candidate that is a child of the predicate in the dependency tree and has "SUBJECT" as deprel or "NNP" as part of speech (POS) tag, it is likely to be the role "A0". These clues are often helpful, but they often are not specific enough to predict the exact semantic role. For example, given the sentences "we cut the cake on Monday" and "we cut the cake on the table", the words "on" in both sentences have the same POS path to the predicate "cut", but they are two different roles ("AM-TMP" and "AM-LOC" respectively). Traditional approaches for SRL encode linguistic clues as indicator features such as the observed POS tag of a word and the syntactic path to its head. However, their occurrence is often sparse in the training data and in combination with being ambigous signals for semantic roles they do not generalize well across domains. When the target data is in a different domain, its vocabulary differs from the training data, and the classifier may fail to recognize semantic or meaning clues. If the predicate is observed in the training data, then the syntactic patterns may still be useful. However, in the worst case, if the predicate is unseen, both of the clues become weak leading to the most difficult case for SRL.

4.2 Method Description

Motivated by the successes of distributional word representations in capturing word similarity and of co-training in boosting the performance of classification by using two different views which naturally fit the two types of clues discussed above, we propose a combination of these approaches to tackle the problem. Our method, shown in Algorithm 1 and described in detail in Section 4.4, leverages unlabeled data to improve the performance of SRL. We first extend the feature set with distributional representations induced from a large unlabeled corpus. The two modules PI and PD are trained and used to label the predicate and predicate senses in the unlabeled texts. We then divide the feature sets of AI and AC into two intuitive sets, one for the semantic information of the argument candidate, and one for the syntactic information. A co-training strategy is applied twice, once to AI and once to AC. In each process, local classifiers are trained on the labeled data using the two divided feature sets. Then, we label a set of unlabeled data from the target domain using these classifiers. At the final step, selected newly labeled instances are used to train a global classifier with the joint feature set of the two local classifiers.

4.3 Word Embeddings

Instead of using the compositional representations proposed in (Roth and Woodsend, 2014), we follow a simple and natural approach to employ word embeddings: For each *word feature* (e.g., argument word, predicate word, right child word), we map its value to a distributed representation and concatenate the new representation to the original feature vector. For each *word set feature* (e.g., child word set), we sum up the distributional representations of the set elements, and concatenate the obtained vector to the original feature vector. Under this approach, word embeddings can be easily applied to any step of SRL which contains word or word set features. We will show later in the experiments that this simple approach

Table 1: The feature division for Co-training. -,N,V indicates that the feature is not used, used for nominal frames, and used for verbal frames respectively. Note that we omit distributional representation features and feature bigrams.

	Identification		Classification			Identification		Classification	
	Sem	Syn	Sem	Syn		Sem	Syn	Sem	Syn
DeprelPath	-	V,N	-	V	Deprel	V	-	V	-
POSPath	-	V,N	-	V	RightSiblingWord	-	V	-	-
Word	V,N	-	V,N	-	PredParentWord	-	V	-	V
PredParentPOS	-	V	-	V	Position	V,N	-	V,N	-
PredLemmaSense	V	V	V,N	V,N	PredLemma	N	N	V,N	V,N
ChildWordSet	N	-	-	-	RightChildPOS	N	-	V,N	-
PredPOS	N	N	V	V	RightChildWord	N	-	V,N	-
ChildDeprelSet	-	-	V	-	POS	-	-	V	-
LeftSiblingPOS	-	-		V,N	LeftChildPOS	-	-	V	-
PredWord	-	-	V,N	V,N	LeftSiblingWord	-	-		N
LeftChildWord	-	-	N	-	ChildPOSSet	-	-	N	-

gives us comparable results to other state-of-the-art systems. Word embeddings also help to improve the semantic or meaning clues, and can therefore boost the performance of the co-training strategy.

4.4 Co-training

The *co-training* semi-supervised learning paradigm was first proposed by Blum and Mitchell (1998), aiming at exploiting unlabeled data to improve performance given limited training data. The classical algorithm applies when the data can be represented by two or more separate, but redundant "views" such as two disjoint feature subsets. For example, web pages can be described by either the text on the web page or the text from hyperlinks pointing to the web page. The two classifiers trained on two "views" of the data can help each other, by adding one's most confident examples into the other one's training set.

In this work, we apply a variant of co-training to the two argument steps. We propose to divide the SRL feature sets into two views based on the dependency tree. The Sem_View, which emphasizes the semantic or meaning clues (and is related to the headword view of He and Gildea (2006)), consists of all features based on the argument candidate itself and all the words that are lower than the argument candidate on the dependency tree (e.g., left child word, right child word, children word set). The Syn_View, which emphasizes syntactic clues (and is related to the path view of He and Gildea (2006)), consists of all features based on the path from the argument to the predicate and all the words that are equal or higher than the argument candidate on the dependency tree (e.g., left sibling word, right sibling word, parent word). The features referring to the predicate itself are included in both views. More details of the feature division can be found in Table 1.

The details of the co-training method are shown in Algorithm 1. First of all, we extend our feature sets with word embeddings as proposed in Section 4.3. The two predicate modules PI and PD are trained on the original training data, then used to label the unannotated data. After that, we start the co-training process. For each of AI and AC[2], we loop for a number of iterations: the two views are trained on the labeled data using their corresponding feature sets with logistic regression. Both of the two views are used to label the unannotated data. We then select informative examples from the unlabeled dataset to be used as extra training data. Our selection strategy is to select the examples that cause a disagreement between the two views: one view is confident that the example belongs to class c (labeling score is higher than a certain threshold) while the other view is uncertain about this (labeling score is between a lower and upper threshold). This strategy differs from a classical co-training set up where we select the examples where the two local classifiers confidently agree (Blum and Mitchell, 1998). This is motivated by the

[2]Note that the co-training is applied to AI and AC separately. That means the two views of AI interact with each other, and the two views of AC interact with each other, but views from AI do not interact with views from AC.

Algorithm 1: The SRL Co-training Algorithm.

Input : A large collection of labeled sentences $\mathbf{L}$ and one of unlabeled sentences $\mathbf{U}$
Output : A full SRL

Enrich feature sets with word embeddings.
Train PI and PD on $\mathbf{L}$.
Label $\mathbf{U}$ by PI and PD.
Divide the feature sets of AI and AC into semantic/meaning clues (Sem_View) and syntactic/word-order clues (Syn_View).
for *each of the two argument sub-tasks (AI or AC)* **do**
    $\mathbf{V} = \mathbf{U}$
    $n = 0$
    while $\mathbf{V} \neq \emptyset \wedge n < |\mathbf{L}|$ **do**
        Build local classifier A on $\mathbf{L}$ using Sem_View features
        Build local classifier B on $\mathbf{L}$ using Syn_View features
        $n = |\mathbf{L}|$
        for *x in* $\mathbf{V}$ **do**
            Use A and B to label x.
            if *A or B is confident (labeling score* $> t_1$*) that* x *belongs to class* c *and the other is uncertain (*$t_3 <$ *labeling score* $< t_2$*)* **then**
                Add (x, c) to $\mathbf{L}$
            end
            Remove x from $\mathbf{V}$
            if $|\mathbf{L}| \geq n + k$ **then**
                Break the for loop
            end
        end
    end
    Build the global classifier for AI (or AC) on $\mathbf{L}$ using the joint feature set.
end

fact that the training data in SRL is sufficiently large to build a good system, so we only select the most informative new instances: the ones that make a disagreement between the two views. That means we are helping the classifiers to overcome the cases when one of the two clue types is weak which is common in out-of-domain prediction. Intuitively, when one clue type is strong and the other is not good, the strong one can help the other.

To ensure the balance between different classes, for AI, we added the same number of positive and negative examples to the labeled data. For main roles (e.g., A0, A1) in AC, we skip adding an instance to class c if the ratio of the number of instances in c to the rest is increased more than a certain number of times (2.0 in our experiments) since main roles are already the majority in the original training data. The maximum number of instances that could be added to the training data in each iteration is k[3]. The iterations will finish when there is no instance satisfying our selection criteria or there is no unlabeled data left. At the end of the process, a global classifier for each of the argument sub-task modules (AI or AC) is trained on the final labeled training set using the joint feature set of the two views.

Although co-training has reported success in many real-world applications (Blum and Mitchell, 1998; Kiritchenko and Matwin, 2001; Mihalcea, 2004; Javed et al., 2005), its application in SRL is still very limited. It is common in NLP that the two assumptions of co-training do not strictly hold. He and Gildea (2006) discovered that their two co-training views, headword and path, were not balanced. The headword view was more sensitive to new data, and led to a significant drop in precision. The path view was a more accurate and stable indicator for semantic role labeling. By adding distributional word representations to

[3] k is fixed to 50% of the size of the original training data in our experiment.

Table 2: Information about datasets.

	Section	Corpus	Type
Train	02-21	TreeBank	financial news
Dev	24	TreeBank	financial news
Ood	ck01-03	Brown	fiction
UB	ck04-29,cl,cm,cp	Brown	fiction
UW	01	TreeBank	financial news

the feature set, we expect to improve the performance of the headword view when faced with new data, and thereby produce a successful co-training strategy. Also, since it is difficult to make a good prediction with just one of the two different clue types, we build a global classifier for each of AI and AC with a joint feature set at the end of the process.

5 Experiments

We evaluate the performance of our method when training on a large training set and testing on an out-of-domain test set using a collection of unlabeled data. We expect that from the unlabeled data, which is mostly in the same domain as the target data, we can create a number of new training instances that improve system performance on the target domain.

5.1 Settings

We use the same training, development and out-of-domain test set as provided in the CoNLL 2009 shared task (Hajič et al., 2009). We also collect two sets of unlabeled data which are mostly texts in the same domain as the CoNLL 2009 out-of-domain test set. Table 2 shows some information about our datasets. We call the training set Train, development set Dev, out-of-domain test set Ood, unlabeled data from Brown corpus UB and unlabeled data from TreeBank UW. The training data has 39,279 financial sentences with 12,036 predicate words. In the out-of-domain test set, there are 788 different predicate words and 114 of them have never been observed in the training data. The collection of unlabeled data (UB and UW) contains 12,462 fictional and 1,828 financial news sentences.

The prepreprocessing modules including POS tagger, lemmatizer and dependency parser of (Björkelund et al., 2010) are used in our experiments. All the modules are retrained on the CoNLL 2009 training set. We use Word2Vec[4] (**?**) to learn 300-dimensional word representations from unlabeled corpora including Wikipedia[5], Reuters[6], TreeBank[7].

Following the standard setting of the CoNLL 2009 shared task, we skip the predicate identification step when evaluating our models. It is only used to annotate unlabeled data in the co-training experiments. All the methods are evaluated using the official CoNNL 2009 evaluation software.

We set λ, the parameter controlling the balance between precision and recall, to 0.3 in all of our experiments based on tuning that parameter on the Dev set.

We use as a baseline our system trained on the feature sets proposed by Björkelund et al. (2010) (Baseline).

To evaluate the effectiveness of our proposed inference, we replace the inference of the above Baseline model by a simple joint inference maximizing the local prediction scores over AI and AC (see Section 3). We call this model Simple. Furthermore, we also implement NoJoint model, which has the same settings as Baseline, but the joint inference is replaced by the classical predictions of the local modules.

Our three different proposed approaches are evaluated: using only word embeddings as in Section 4.3 (WE), using co-training strategy as in Section 4.4 (CO) but without word embeddings, and using the combination of word embeddings and co-training as in Section 4.4 (WECO). We also compare our

[4]https://code.google.com/archive/p/word2vec/
[5]http://corpus.byu.edu/wiki/
[6]http://about.reuters.com/researchandstandards/corpus/
[7]https://catalog.ldc.upenn.edu/ldc99t42

Table 3: Detailed CoNLL 2009 results on seen semantic frames (seen predicates) and unseen semantic frames (unseen predicates) on the out-of-domain test set. Significant differences (computed using a randomization test; cf. Yeh (2000)) from Baseline in terms of F1-score are marked by asterisks (* $p < 0.05$)

	Seen frames			Unseen frames		
Method	P	R	F1	P	R	F1
Baseline	79.7	72.7	76.1	77.0	67.1	71.7
WE	80.2	73.6	76.7*	78.8	67.6	72.8*
CO	80.3	72.3	76.1	80.8	66.8	73.1*
WECO	80.4	73.7	**76.9***	81.6	67.9	**74.2***

Table 4: CoNLL 2009 results on the out-of-domain test set

	P	R	F1
CoNLL-2009 1st place	-	-	74.6
Björkelund et al. (2010)	77.9	73.6	75.7
Roth and Woodsend (2014)	-	-	75.9
Lei et al. (2015)	-	-	75.6
Täckström et al. (2015)	-	-	75.5
FitzGerald et al. (2015)	-	-	75.9
Roth and Lapata (2016)	79.7	73.6	76.5
NoJoint	75.9	72.8	74.3
Simple	73.3	75.3	74.3
Baseline	79.5	72.3	75.7
WE	80.1	73.1	76.4
CO	80.3	71.9	75.9
WECO	80.5	73.2	**76.7**

results with the current state-of-the-art systems. We tune the thresholds t_1, t_2, and t_3 in Algorithm 1 by measuring performance on Dev set resulting in $(t_1, t_2, t_3) = (0.8, 0.5, 0.3)$.

5.2 Results

From Table 3, we can see that including word embeddings with the capacity of better capturing word similarity already improves the out-of-domain prediction. Using co-training seems to be more useful when working with unseen frames. It can be seen that CO gives high precision which has possibly benefited from the enforced agreement between the two views. Meanwhile, WE appears to be better in improving recall because of the good capacity to capture word similarity. Interestingly, the combination of these two approaches WECO brings us high scores in both precision and recall. It obtains the best results especially for unseen semantic frames which are the most difficult cases for SRL. WECO improves the prediction for unseen frames over the baseline, WE and CO 2.5, 1.4 and 1.1 F1 points respectively.

As can be seen from Table 4, with a simple learning architecture (local learning, no reranker) and an effective inference, our Baseline model already obtains results comparable to those of state-of-the-art systems. It reaches the F1 score of 75.7% which is the same as Björkelund et al. (2010) using a reranker on top of local classifiers with the same feature set as our baseline model. Our Baseline model surpasses both Simple and NoILP models by 1.4 F1 points proving the effectiveness of our ILP objective function. With our best model, WECO, the official CoNLL 2009 F1 score obtained on the out-of-domain test set outperforms a much more complicated system using reranker and neural network proposed in (Roth and Lapata, 2016) by 0.2 F1 points.

Selection strategy To evaluate our selection strategy in the co-training, we also perform an experiment with a classical selection strategy in which we select examples on which the two views agree with high confidence (i.e. larger than t_1 in both views), resulting in a reduction of 0.2 F1 points on the out-of-domain

test set.

6 Discussion

Semi-supervised setting The experiments show the promising application of semi-supervised methods in out-of-domain scenarios. We present the first successful case of using co-training for SRL when using all available training data (and not an artificially limited small subset). These successes encourage us to develop more advanced techniques digging into the collaboration of the two views in SRL. Semi-supervised techniques have the capacity to exploit a huge amount of unlabeled data, which is cheaper than manually-built annotated data. However, the success of these techniques depends on the quality of unlabeled data. If the unlabeled data itself does not contain informative knowledge, then semi-supervised methods will not help in improving the performance.

System architecture As shown before, even with a simple learning technique we still can obtain very strong results with an effective inference. This architecture is beneficial in semi-supervised settings which often require dealing with new training examples and new features. The inference might increase the computational complexity of the prediction, but given the current progress in efficient software and hardware solutions this will not pose serious problems.

7 Conclusion

We have presented a semi-supervised SRL system facing the most difficult case of SRL: predicting out-of-domain and unseen semantic frames. Our system, with a simple learning architecture and effective joint inference, obtains very strong results on the standard benchmark SRL data from the CoNLL 2009 shared task. We propose using a collaboration of word embeddings and a variant of co-training to exploit unlabeled data. Our method leverages the collaborative relationship of the two signal types in SRL (semantic and syntactic clues) to create informative new training examples that help out-of-domain prediction. An experiment exploiting unlabeled data which includes mostly fictional texts from the Brown corpus achieves better performance than state-of-the-art models on the out-of-domain test set of the CoNLL 2009 shared task.

Last but not least, our SRL system is made publicly available at http://liir.cs.kuleuven.be/software.php.

Acknowledgment

This work is funded by the EU ICT FP7 FET project "Machine Understanding for interactive StorytElling" (MUSE) `http://www.muse-project.eu/`.

References

Collin F. Baker, Charles J. Fillmore, and John B. Lowe. 1998. The berkeley framenet project. In *Proceedings of ACL 1998*, ACL '98, pages 86–90, Stroudsburg, PA, USA. Association for Computational Linguistics.

Anders Björkelund, Bernd Bohnet, Love Hafdell, and Pierre Nugues. 2010. A high-performance syntactic and semantic dependency parser. In *COLING 2010: Demonstrations*, pages 33–36, Beijing, China.

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *Proceedings of COLT 1998*, pages 92–100, New York, NY, USA. ACM.

R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12:2493–2537.

Koen Deschacht, Jan De Belder, and Marie-Francine Moens. 2012. The latent words language model. *Computer Speech and Language*, 26(5):384–409, October.

Nicholas FitzGerald, Oscar Täckström, Kuzman Ganchev, and Dipanjan Das. 2015. Semantic role labeling with neural network factors. In *Proceedings of EMNLP 2015*.

Hagen Fürstenau and Mirella Lapata. 2012. Semi-supervised semantic role labeling via structural alignment. *Computational Linguistics*, 38(1):135–171.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The conll-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of CoNLL 2009: Shared Task*, pages 1–18, Stroudsburg, PA, USA. ACL.

Shan He and Daniel Gildea. 2006. Self-training and co-training for semantic role labeling: Primary report. Technical report, Technical Report 891, University of Rochester.

Omar Javed, Saad Ali, and Mubarak Shah. 2005. Online detection and classification of moving objects using progressively improving detectors. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 696–701.

Svetlana Kiritchenko and Stan Matwin. 2001. Email classification with co-training. In *Proceedings of CASCON 2001*, pages 8–. IBM Press.

Joo-Young Lee, Young-In Song, and Hae-Chang Rim. 2007. Investigation of weakly supervised learning for semantic role labeling. In *Proceedings of ALPIT 2007*, pages 165–170, Washington, DC, USA. IEEE Computer Society.

Tao Lei, Yuan Zhang, Lluís Màrquez, Alessandro Moschitti, and Regina Barzilay. 2015. High-order low-rank tensors for semantic role labeling. In *Proceedings of NAACL 2015*, pages 1150–1160, Denver, Colorado, May–June. ACL.

Rada Mihalcea. 2004. Co-training and self-training for word sense disambiguation. In *In Proceedings of CoNLL 2004*.

Martha Palmer, Daniel Gildea, and Paul Kingsbury. 2005. The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106, March.

Vasin Punyakanok, Dan Roth, and Wen-tau Yih. 2008. The importance of syntactic parsing and inference in semantic role labeling. *Computation Linguistics*, 34(2):257–287, June.

Michael Roth and Mirella Lapata. 2016. Neural semantic role labeling with dependency path embeddings. In *Proceedings of ACL 2016*, Berlin, Germany.

Michael Roth and Kristian Woodsend. 2014. Composition of word representations improves semantic role labelling. In *EMNLP 2014*, pages 407–413.

Rasoul Samad Zadeh Kaljahi and Mohd Sapiyan Baba. 2011. Investigation of co-training views and variations for semantic role labeling. In *Proceedings of Workshop on Robust Unsupervised and Semisupervised Methods in Natural Language Processing*, pages 41–49, Hissar, Bulgaria, September.

Vivek Srikumar and Dan Roth. 2011. A joint model for extended semantic role labeling. In *Proceedings of EMNLP 2011*, pages 129–139, Stroudsburg, PA, USA. ACL.

Oscar Täckström, Kuzman Ganchev, and Dipanjan Das. 2015. Efficient inference and structured learning for semantic role labeling. *TACL*, 3:29–41.

Kristina Toutanova, Aria Haghighi, and Christopher D. Manning. 2005. Joint learning improves semantic role labeling. In *Proceedings of ACL 2005*, pages 589–596, Stroudsburg, PA, USA. ACL.

Peter D. Turney and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research*, 37(1):141–188, January.

Jason Weston, Frédéric Ratle, and Ronan Collobert. 2008. Deep learning via semi-supervised embedding. In *Proceedings of ICML 2008*, pages 1168–1175, New York, NY, USA. ACM.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Proceedings of COLING 2000*, COLING '00, pages 947–953, Stroudsburg, PA, USA. ACL.