Semi-Supervised Semantic Role Labeling with Cross-View Training

Rui Cai and **Mirella Lapata**
Institute for Language, Cognition and Computation
School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh EH8 9AB
Rui.Cai@ed.ac.uk mlap@inf.ed.ac.uk

Abstract

The successful application of neural networks to a variety of NLP tasks has provided strong impetus to develop end-to-end models for semantic role labeling which forego the need for extensive feature engineering. Recent approaches rely on high-quality annotations which are costly to obtain, and mostly unavailable in low resource scenarios (e.g., rare languages or domains). Our work aims to reduce the annotation effort involved via semi-supervised learning. We propose an end-to-end SRL model and demonstrate it can effectively leverage unlabeled data under the cross-view training modeling paradigm. Our LSTM-based semantic role labeler is jointly trained with a sentence learner, which performs POS tagging, dependency parsing, and predicate identification which we argue are critical to learning directly from unlabeled data without recourse to external pre-processing tools. Experimental results on the CoNLL-2009 benchmark dataset show that our model outperforms the state of the art in English, and consistently improves performance in other languages, including Chinese, German, and Spanish.

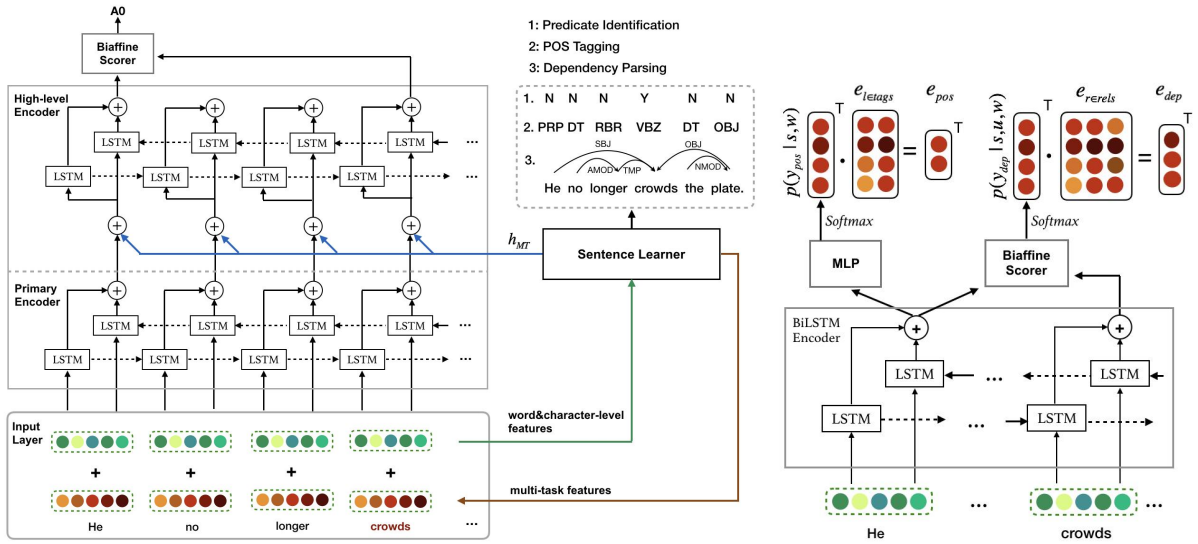
1 Introduction

Semantic role labeling — the task of automatically identifying and labeling the semantic roles conveyed by sentential constituents — has enjoyed renewed interest in the last few years thanks to the popularity of neural network models and their ability to learn continuous representations which forego the need for extensive feature engineering. Recent modeling developments aside, semantic role labeling (SRL) has been generally recognized as a core task in NLP with relevance for applications ranging from machine translation (Aziz et al., 2011; Marcheggiani et al., 2018), to information extraction (Christensen et al., 2011), and summarization (Khan et al., 2015).

State-of-the art semantic role labelers (He et al., 2018b; Cai et al., 2018) rely on high-quality annotations (of semantic predicates and their arguments) for use in training. These annotations are costly to obtain, and mostly unavailable in low resource scenarios (e.g., rare languages or domains) motivating the need for effective semi-supervised methods that leverage unlabeled examples. Cross-View Training (CVT; Clark et al. 2018) is a recently proposed semi-supervised learning algorithm that improves representation learning using a mix of labeled and unlabeled data. The main idea behind CVT is to train a model to produce consistent predictions across different restricted views of the input with the aid of auxiliary prediction tasks. Clark et al. (2018) demonstrate performance gains when applying CVT to sequence tagging tasks, machine translation, and dependency parsing.

Unfortunately, application of CVT to semantic role labeling is fraught with difficulty. This is partly due to the nature of the task which relies on various syntactic features, even when conceptualized as a sequence labeling task (Marcheggiani et al., 2017; He et al., 2018b; Cai et al., 2018). The reliance on syntactic features is problematic for semi-supervised training, since these would have to be extracted from large amounts of unlabeled data. In addition, any semantic role labeler would need to identify (and disambiguate) predicates prior to labeling their augments which might be given during training (e.g., as in the CoNLL 2009 dataset), but would still have to be detected on unlabeled data. Resorting to various pre-processing tools for semi-supervised training almost defeats the purpose of using unlabeled data which would have to be annotated, albeit automatically, with pre-trained models which require labeled data on their own, and might not be portable across languages and domains. Moreover, the usage of external tools often leads to pipeline-style

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 1018–1027, Hong Kong, China, November 3–7, 2019.



(a) Semantic role labeler (left) and sequence learner (right)

(b) Sentence learner.

Figure 1: Overview of end-to-end SRL model.

architectures where errors propagate to later processing stages, affecting model performance.

In this paper we aim to render semi-supervised learning for semantic role labeling as simple as possible, by eliminating the reliance on multiple external pre-processing tools. We develop a sentence learner which is able to perform all tasks subsidiary to semantic role labeling (i.e., POS tagging, dependency parsing, and predicate detection) on labeled and unlabeled data. The sentence learner is jointly trained with a semantic role labeler, and its outputs are fed to the semantic role labeler during supervised and semi-supervised learning. Aside from building a self-sufficient semantic role labeler which can be directly applied on plain text, an added benefit of the proposed approach is that the sentence learner naturally provides multiple hidden layers (for the various subtasks) from which "multi-task hidden features" (Peters et al., 2018) can be extracted.

In addition to overcoming the difficulty of utilizing plain text for semi-supervised SRL, we show that application of CVT to SRL requires special attention over and above the sequence tagging and dependency parsing tasks discussed in Clark et al. (2018). We investigate how to best formulate different views for CVT focusing on semantic predicates which have been proven to be very useful in recent SRL models (Marcheggiani et al., 2017; Marcheggiani and Titov, 2017; Cai et al., 2018). Experimental results on the CONLL-2009 benchmark datasets show that our model is able to outperform the state of the art in English, and to improve SRL performance in other languages, including Chinese, German, and Spanish.

2 Model Description

Figure 1 provides a schematic overview of our model which has two main components, namely a sentence learner and a semantic role labeler. The sentence learner consists of:

- look-ups of word embeddings and character embeddings;
- a bidirectional LSTM (BiLSTM) encoder which takes as input the representation of each word in a sentence and produces context-dependent representations;
- a multi-task prediction module to perform POS tagging, dependency parsing, and predicate identification.

While the semantic role labeler consists of:

- an input layer which combines multi-source representations of the input sentence;
- a primary bidirectional LSTM (BiLSTM) encoder which takes as input the representation of each word in a sentence and produces context-dependent embeddings;
- a high-level BiLSTM encoder which takes as input the hidden states of the primary BiLSTM and multi-task hidden features;
- a biaffine classifier for calculating the score of each semantic role for each word.

2.1 Sentence Learner

The sentence learner (see Figure 1b) operates over sentences to perform the intermediate tasks of POS tagging, dependency parsing, and predicate identification, which are subsequently used to inform the decisions of the semantic role labeler.

Sentence Encoder As we expect the sentence encoder to be applied directly to plain text, we represent words as the concatenation of character- and word-level features. We learn character-level representations x_{cr} by feeding character embeddings to a convolutional neural network module. We represent words with randomly initialized word embeddings $x_{re} \in R^{d_w}$, pre-trained word embeddings $x_{pe} \in R^{d_w}$ estimated on an external text collection, and pre-trained ELMo embeddings x_{elmo} (Peters et al., 2018). The final word representation is given by $x = x_{re} \circ x_{pe} \circ x_{cr} \circ x_{elmo}$, where $\circ$ represents concatenation.

Following Marcheggiani et al. (2017), sentences are represented using a two-layer bi-directional LSTM (Hochreiter and Schmidhuber, 1997); the BiLSTM receives at time step t a representation x for each word and recursively computes two hidden states, one for the forward pass ($\overrightarrow{h}^t$), and another one for the backward pass ($\overleftarrow{h}^t$). Each word is the concatenation of its forward and backward LSTM state vectors $h^t = \overrightarrow{h}^t \circ \overleftarrow{h}^t$.

Multi-task Learning After obtaining word representations with the sentence encoder, the input is POS tagged, dependency parsed, and predicates are identified. Given sentence s, the probability distribution of POS tags for word w is obtained using a one hidden-layer neural network applied to the corresponding encoder output h_2^w:

$$p(y_{pos}|s,w) = \text{softmax}(U \cdot \text{ReLU}(W h_2^w) + b) \quad (1)$$

For dependency parsing, words in a sentence are treated as nodes in a graph. In particular, each word w in sentence s receives exactly one in-going edge (u, w, r) from head word u to its dependent w with relation r. We use a graph-based dependency parser similar to the one presented in Clark et al. (2018), which treats dependency parsing as a classification task and its goal is to predict which in-going edge (u, w, r) connects to each word w. Mathematically, the probability of an edge is:

$$p((u,w,r)|s) = \propto \exp(\text{score}(h_2^u, h_2^w, r)) \quad (2)$$

where "score" is the scoring function:

$$\begin{aligned} \text{score}(z_1, z_2, r) = &\text{ReLU}(W_{head} z_1 + b_{head}) \\ &(W_r + W)\,\text{ReLU}(W_{dep} z_2 + b_{dep}) \end{aligned} \quad (3)$$

The bilinear classifier uses a weight matrix W_r specific to the candidate relation and a weight matrix W shared across all dependency relations.

With regard to predicate identification, we introduce a virtual root following Cai et al. (2018) who model the entire SRL task as word pair classification. Similarly to the dependency parsing module described above, representations produced by the encoder for the virtual root and words are passed through separate hidden layers and a biaffine classifier is applied to produce a score for each word.

2.2 Semantic Role Labeler

Word Representation For our SRL model, words are represented by a vector x which is the concatenation of four types of features: predicate-specific, character-level, word-level, and multi-task features. Following previous work (Marcheggiani et al., 2017), we leverage a predicate specific indicator embedding x_{ie} rather than directly using a binary flag. Character- and word-level features are shared with the sentence learner.

With regard to multi-task features, for each word w in input sentence s, we employ a probability-weighted POS tag embedding e_{pos} and dependency relation embedding e_{dep}:

$$\begin{aligned} e_{pos} &= \sum_{l \in tags} p(y_{pos} = l|s,w)[l] * e_l \\ e_{dep} &= \sum_{r \in rels} p(y_{dep} = r|s,u,w)[r] * e_r \end{aligned} \quad (4)$$

where e_l and e_r and the embeddings of POS tags and dependency relations respectively, and $p(y_{dep} = r|s,u)$ is the probability of relation r given its predicted dependency head u.

In order to incorporate more syntactic information, we adopt as an additional feature the probability of linking a word to candidate predicate x_{pr} which we obtain from the sentence learner. x_{pr} is made of two scalar values, p_{head} and p_{dep}, which represent the probability of a word being the syntactic head or dependent of the current predicate, respectively.

Multi-task Hidden Features Drawing inspiration from ELMo (Peters et al., 2018), a recently

proposed model for generating word representations based on bidirectional LSTMs trained with a coupled language modeling objective, we extract various hidden features via multi-task learning. ELMo representations are deep, essentially a linear combination of representations learnt at all layers of an LSTM instead of just the final layer. Compared with unsupervised ELMo representations, our sentence learner takes advantage of labeled data — it attempts to learn representations towards multiple SRL-related tasks rather than generally effective ones.

In order to utilize all hidden layers in the sentence learner, we collapse them into a single vector. Although we could simply concatenate these or select the top layer, we compute multi-task hidden features h_{MT} as a weighting of the BiLSTM layers, followed by a non-linear projection:

$$h_{MT} = \mathrm{ReLU}(W_{hidden}(\gamma \sum_{j=1}^{j=L} \beta_j h_j)) \quad (5)$$

where L is the depth of sentence learner, β are softmax-normalized weights for h_j, and the scalar parameter γ is of practical importance to aid optimization (Peters et al., 2018).

Biaffine Role Scorer After the high-level BiLSTM encoder produces representations h for each word, we perform two distinct non-linear transformations for the currently considered predicate and its candidate arguments, respectively:

$$\begin{aligned} h_{pred} &= \mathrm{ReLU}(W_{pred} h + b_{pred}) \\ h_{arg} &= \mathrm{ReLU}(W_{arg} h + b_{arg}) \end{aligned} \quad (6)$$

where h_{pred} and h_{arg} are hidden representations for the predicate and candidate arguments. The score s_{role} of a semantic role between a predicate and its arguments is calculated as:

$$\begin{aligned} s_{role} = & h_{arg}^{\top} W_{role} h_{pred} \\ & + U_{role}(h_{arg} \circ h_{pred}) \\ & + b_{role} \end{aligned} \quad (7)$$

where W_{role}, U_{role}, and b_{role} are parameters updated during training.

2.3 Cross-view Training for SRL

CVT works by improving representation learning for a model. Let $D_{ul} = \{x_1, x_2, .., x_N\}$ represent an unlabeled dataset. We use $p_\theta(y|x_i)$ to denote the output distribution over classes produced by the model with parameters θ. CVT adds multiple different auxiliary prediction modules to a model, which are used when learning on unlabeled examples. Each prediction module takes as input an intermediate representation $h^j(x_i)$ produced by a primary BiLSTM encoder and outputs a distribution over all possible classes $p_\theta^j(y|x_i)$. Each h^j is chosen such that it can only see parts of the input.

Given an unlabeled example, the model first produces soft targets $p_\theta(y|x_i)$ by performing inference. CVT then trains auxiliary prediction modules to match the teacher prediction module on the unlabeled data by minimizing:

$$\mathcal{L}_{\mathrm{CVT}}(\theta) = \frac{1}{D_{ul}} \sum_{x_i \in D_{ul}} \sum_{j=1}^{k} D(p_\theta(y|x_i), p_\theta^j(y|x_i)) \quad (8)$$

where D is a distance function between probability distributions (we use KL divergence). During training, we keep predictions $p_\theta(y|x_i)$ from the teacher module fixed so that the auxiliary modules learn to imitate the teacher, but not vice versa. As auxiliary modules train, the representations they use as input improve so they are useful for making predictions even when some of the models' inputs are not available. This in turn improves the primary prediction module, which is built on top of the same shared representations.

We applied CVT on the primary Bi-LSTM encoder of our semantic role labeler, while utilizing the output of the sentence learner on the unlabeled data. Given unlabeled sentence $s = w^1, ..., w^T$, the primary Bi-LSTM encoder produces hidden representations h_{pri} for each word, while the semantic role labeler produces the teacher prediction. The sentence learner may recognize more than one words as predicates in sentence s, and we just randomly choose one as the target predicate w^p.

The auxiliary prediction modules take $\overrightarrow{h}^t_{pri}$ and $\overleftarrow{h}^t_{pri}$ as input. Specifically, we add the following four auxiliary prediction modules to the model:

$$\begin{aligned} p_\theta^{\mathrm{fwd}}(r^t|w^t, w^p, s) &= \mathrm{NN}^{\mathrm{fwd}}(\overrightarrow{h}^t_{pri}(s)) \\ p_\theta^{\mathrm{bwd}}(r^t|w^t, w^p, s) &= \mathrm{NN}^{\mathrm{bwd}}(\overleftarrow{h}^t_{pri}(s)) \\ p_\theta^{\mathrm{future}}(r^t|w^t, w^p, s) &= \mathrm{NN}^{\mathrm{future}}(\overrightarrow{h}^{t-1}_{pri}(s)) \\ p_\theta^{\mathrm{past}}(r^t|w^t, w^p, s) &= \mathrm{NN}^{\mathrm{past}}(\overleftarrow{h}^{t+1}_{pri}(s)) \end{aligned} \quad (9)$$

The "forward" module makes predictions without seeing the right context of the current word.

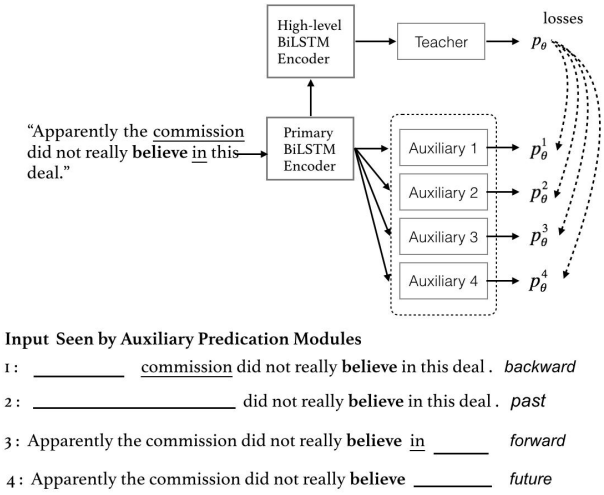


Figure 2: CVT on a sentence from the CoNLL 2009 training set. The current predicate is "*believe*", while "*commission*" and "*in*" are two candidate arguments.

The "future" module makes predictions without seeing the right context *and* the current word itself. The "backward" and "past" modules are defined analogously for left contexts. Figure 2 illustrates the auxiliary modules and the types of context they see. Unlike the biaffine role scorer, the auxiliary modules do not explicitly use the hidden state of the target predicate, so as to encourage the primary Bi-LSTM encoder to capture long-distance relations between the predicate and its context.

We empirically observed (see Section 3) that applying CVT on representations which do not "see" the target predicate is not ideal for SRL. We therefore devised a strategy which applies different auxiliary modules to each word depending on its relative position to the target predicate. We only apply "backward" and "past" modules to words preceding the predicate, while "forward" and "future" modules apply to words following the predicate. This way, we ensure that each word is aware of the current predicate when performing CVT. In the example in Figure 2, "backward" and "past" views would be applied to *commission* and "forward" and "future" views to *in*.

We also apply CVT on the first hidden layer of the sentence learner to further improve the performances of auxiliary tasks, utilizing the views introduced in Clark et al. (2018) for sequence tagging and dependency parsing.

2.4 Training Objectives

For both supervised learning and cross-view training our model makes predictions on labeled and unlabeled examples across all tasks (SRL and auxiliary tasks). During supervised learning, the model is trained on labeled data and its objective is the sum of cross-entropy losses for all tasks. With respect to multi-task CVT, the model takes unlabeled examples as input and calculates the CVT loss given in Equation (8). The semi-supervised objective is the sum of the CVT loss for all auxiliary modules across all tasks.

Hyperparameter	value
d_w (English word embeddings)	100
d_w (other languages word embeddings)	300
d_{cr} (character-level representations)	100
d_{pos} (POS embeddings)	32
d_{de} (dependency label embeddings)	32
d_{ie} (predicate indicator embeddings)	16
Bi-LSTM hidden states size	400
hidden-layer size in biaffine scorers	300
Multi-task hidden features size	200
primary BiLSTM depth	1
high-level BiLSTM depth	2
batch size	30
learning rate	0.001

Table 1: Hyperparameter values.

3 Experiments

We implemented our model[1] in PyTorch and evaluated it on the English CoNLL-2009 benchmark following the standard training, testing, and development set splits. To evaluate whether our model generalizes to other languages, we also report experiments on Chinese, German, and Spanish, again using standard CoNLL-2009 splits. This subset of languages has been commonly used in previous work (Björkelund et al., 2010; Roth and Lapata, 2016; Lei et al., 2015) and allows us to compare our model against a wide range of alternative approaches. The benchmarks contain gold-standard dependency annotations, and also gold lemmas, part-of-speech tags, and morphological features. With regard to unlabeled datasets, we used the 1 Billion Word Language Model Benchmark (Chelba et al., 2013) for English, the Sougou News Data[2] for Chinese, the NEGRA corpus [3] for

[1] Our code is publicly available at https://github.com/RuiCaiNLP/SemiSRL.

[2] www.sogou.com/labs/resource/ca.php

[3] www.coli.uni-sb.de/sfb378/negra-corpus/

Single Models	P	R	F_1
Björkelund et al. (2010)	87.1	84.5	85.8
Lei et al. (2015)	—	—	86.6
FitzGerald et al. (2015)	—	—	86.7
Roth and Lapata (2016)	88.1	85.3	86.7
Marcheggiani and Titov (2017)	89.1	86.8	88.0
Marcheggiani et al. (2017)	88.7	86.8	87.7
He et al. (2018b)	89.7	89.3	89.5
Cai et al. (2018)	89.9	89.2	89.6
Li et al. (2018)	90.3	89.3	89.8
Ours (supervised training)	91.1	90.4	**90.7**
Ours (with CVT)	91.7	90.8	**91.2**
Ensemble Models	P	R	F
FitzGerald et al. (2015)	—	—	87.7
Roth and Lapata (2016)	90.3	85.7	87.9
Marcheggiani and Titov (2017)	90.5	87.7	89.1

Table 2: English results on CoNLL-2009 in-domain (WSJ) test set. Differences in F_1 between our models and previous systems are statistically significant ($p < 0.05$) using stratified shuffling (Noreen, 1989).

Single Models	P	R	F_1
Björkelund et al. (2010)	75.7	72.2	73.9
Lei et al. (2015)	-	-	75.6
FitzGerald et al. (2015)	-	-	75.2
Roth and Lapata (2016)	76.9	73.8	75.3
Marcheggiani and Titov (2017)	78.5	75.9	77.2
Marcheggiani et al. (2017)	79.4	76.2	77.7
He et al. (2018b)	81.9	76.9	79.3
Cai et al. (2018)	79.8	78.3	79.0
Li et al. (2018)	80.6	79.0	79.8
Ours (supervised training)	82.1	81.3	**81.6**
Ours (with CVT)	83.2	81.9	**82.5**
Ensemble Models	P	R	F
FitzGerald et al. (2015)	-	-	75.5
Roth and Lapata (2016)	79.7	73.6	76.5
Marcheggiani and Titov (2017)	80.8	77.1	78.9

Table 3: CoNLL-2009 out-of domain results (English; Brown test set). Differences in F_1 between our models and previous systems are statistically significant ($p < 0.05$) using stratified shuffling (Noreen, 1989).

German, and the Spanish Language News Corpus[4] for Spanish.

For experiments on English, we used the embeddings of Dyer et al. (2015) which were learned using the structured skip n-gram approach of Ling et al. (2015). We also used a convolutional neural network (Chiu and Nichols, 2016; Ma and Hovy, 2016) to learn character-level representations. For Chinese, Spanish, and German word embeddings were pre-trained on Wikipedia using `fastText` (Bojanowski et al., 2017).

The Bi-LSTM encoders in our model, used recurrent dropout (Gal and Ghahramani, 2016) with an 80% keep probability between time-steps and layers during supervised training; keep probability was set to 90% when applying the model to unlabeled data. We used the Adam optimizer (Kingma and Ba, 2014) and performed hyperparameter tuning and model selection on the English development set; optimal hyperparameter values (for all languages) are shown in Table 1.

3.1 Results

Our results on the English (in-domain) test set are summarized in Table 2. We compared our system against previous models which employ external tools to obtain required features. We also report the results of various ensemble SRL models (second block). Most comparisons involve neural systems which are based on BiLSTMs (Marcheggiani et al., 2017; He et al., 2018b; Marcheggiani and Titov, 2017; Cai et al., 2018) or use neural networks for learning SLR-specific embeddings (FitzGerald et al., 2015; Roth and Lapata, 2016). We also report the results of two strong symbolic models based on tensor factorization (Lei et al., 2015) and a pipeline of modules that carry out tokenization, lemmatization, part-of-speech tagging, dependency parsing, and semantic role labeling (Björkelund et al., 2010). As can be seen in Table 2, our supervised model outperforms previously published single and ensemble models. With cross-view training, our model achieves 91.2% F_1 (the difference over the supervised model is statistically significant at $p < 0.05$), which is an absolute improvement of 1.4% over the state of the art (Li et al., 2018).

Results on the out-of-domain English test set are presented in Table 3. We include comparisons with the same models as in the in-domain case. Again, our end-to-end model significantly outperforms previously published single and ensemble models, even without taking unlabeled data into account. We achieve a relatively higher improvement with CVT on out-of-domain data (F_1 increases from 81.6% to 82.5%, and the difference is significant at $p < 0.05$). This suggests that semi-

[4] `catalog.ldc.upenn.edu/LDC99T41`

Chinese	P	R	F_1
Björkelund et al. (2010)	82.4	75.1	78.6
Roth and Lapata (2016)	83.2	75.9	79.4
Marcheggiani and Titov (2017)	84.6	80.4	82.5
He et al. (2018b)	84.2	81.5	82.8
Cai et al. (2018)	84.7	84.0	84.3
Li et al. (2018)	84.8	81.2	83.0
Ours (supervised training)	84.9	84.3	84.6
Ours (with CVT)	85.4	84.6	**85.0**
German	P	R	F_1
Björkelund et al. (2010)	81.2	78.3	79.7
Roth and Lapata (2016)	81.8	78.5	80.1
Ours (supervised training)	84.5	82.1	83.3
Ours (with CVT)	84.9	82.7	**83.8**
Spanish	P	R	F
Björkelund et al. (2010)	78.9	74.3	76.5
Roth and Lapata (2016)	83.2	77.4	80.2
Marcheggiani et al. (2017)	81.4	79.3	80.3
Ours (supervised training)	83.0	81.3	82.1
Ours(with CVT)	83.6	82.2	**82.9**

Table 4: CoNLL-2009 results on Chinese, German, and Spanish (test sets). Differences in F_1 between our models and previous systems are statistically significant ($p < 0.05$) using stratified shuffling (Noreen, 1989).

supervised training indeed increases the robustness of our model, leading to more accurate predictions for both SRL and auxiliary tasks.

Table 4 presents the results of our experiments (without ELMo) on Chinese, German, and Spanish. Although we have not performed detailed parameter selection in these languages (i.e., we used the same parameters as in English), our model achieves state-of-the-art performance across all three languages.

3.2 Ablation Studies

To investigate the contribution of the sentence learner and cross-view training, we conducted a series of ablation studies on the English development set without predicate disambiguation.

Our experiments are summarized in Table 5. The first block shows the performance of the full model. In the second block, we assess the effect of different kinds of representations used in our model. Interestingly, the impact of ELMo (about 0.6 in F_1) is slightly less compared to multi-task hidden features (about 0.7 in F_1). This suggests that multi-task hidden features provide as useful information for SRL as pre-trained representations. We next eliminate the sentence learner model and have the semantic role labeler use the predicted POS tags and dependency labels provided in CoNLL-2009 dataset. As can be seen, this leads to a substantial drop in performance over the full model (1.4% in F_1).

Model	P	R	F_1
Ours	88.6	86.8	87.7
w/o ELMo	87.9	86.4	87.1
w/o multi-task hidden features	88.0	86.1	87.0
w/o sentence learner	87.2	85.5	86.3
w/o cross-view training	88.0	85.8	86.9
w/o splitting sentence	87.4	85.7	86.6

Table 5: Ablation results on the CoNLL-2009 English development set.

Preceding	Following	F_1	ΔF_1
Backward	*Forward*	87.3	0.4
Past	*Future*	87.4	0.5

Table 6: CVT with different auxiliary modules for SRL (CoNLL-2009 English development set). Δ denotes difference from model trained without CVT.

In the third block, we remove cross-view training from our model, and observe a 0.8% drop in F_1 over the full model. Finally, we apply the auxiliary modules on the full sentence instead of treating the words preceding and following the target predicate differently, and observe a 0.3% drop in F_1 over the supervised model. This is not surprising as the predicate indicator plays an important role in improving the performance of semantic role labeler.

3.3 CVT Analysis

In Table 6, we briefly explore which auxiliary prediction modules are more important for CVT when applied to SRL. We apply two types of auxiliary modules both of which take care not to "see" the target predicate directly. The "forward/backward" module does not see the right/left context of the current word, while the "future/past" module does not see the right/left context and the current token itself. Both kinds of modules improve performance (over a supervised model without CVT, see second row in Table 5); future and past modules are slightly better corroborating the results of Clark et al. (2018) on sequence tagging. Overall, the results in Table 6 suggest that more restricted views of the input are beneficial.

Strategy	F_1	ΔF_1
Randomly chosen word	87.0	0.1
Randomly chosen predicate	87.7	0.8
Most confident predicate	86.4	-0.5

Table 7: CVT with different predicate selection strategies for SRL (CoNLL-2009 English development set). Δ denotes difference from model trained without CVT.

We further explore how the strategy of selecting the target predicate (in sentences containing multiple candidates) influences performance. For each unlabeled sentence, we adopt the strategy of randomly selecting a predicate amongst those words identified as predicate candidates by the sentence learner. We could also select the predicate with the highest predicted score or a random word from the sentence. The experiments in Table 7 confirm that the adopted strategy works best delivering a 0.8 improvement in F_1 over a supervised model without CVT (second row in Table 5). Selecting the most confident predicate is the worst possible strategy, decreasing F_1 performance by 0.6 over a supervised model without CVT (see Table 5); the model concentrates on a few predicates with very high scores (these tend to be common verbs such as *say*, *is*, and *have*), while ignoring nominal predicates and less frequent verbs. The strategy of randomly selecting a word from the sentence performs better, precisely because it pays attention to a wider range of predicates.

4 Related Work

Our model resonates the recent trend of developing neural network models for semantic role labeling using relatively simple architectures based on bidirectional LSTMs (Marcheggiani et al., 2017; Cai et al., 2018; Strubell et al., 2018). It also agrees with previous work in adopting multi-task learning as a means to improve a main task by jointly learning one or more related auxiliary tasks (Collobert et al., 2011; Søgaard and Goldberg, 2016; Swayamdipta et al., 2017; Peng et al., 2017; Strubell et al., 2018).

The idea of resorting to semi-supervised learning as a means of reducing the annotation effort involved in creating labeled data for SRL is by no means new. Fürstenau and Lapata (2012) propose to augment a labeled dataset with unlabeled examples whose roles are inferred automatically via annotation projection. Other work uses a language model to learn word similarities from unlabeled texts (Croce et al., 2010; Deschacht and Moens, 2009) or constructs an informed prior from labeled data in order to learn a generative model from unlabeled data (Titov and Klementiev, 2012).

More recently, Mehta et al. (2018) have proposed a semi-supervised method for constituency-based SRL. Their work builds upon a state-of-the-art neural model (He et al., 2018b; Peters et al., 2018) whose training objective they augment with a syntactic inconsistency loss component. Their hypothesis is that by leveraging syntactic structure during training, the SRL model may become more robust in low resource scenarios. This method is very much geared towards improving constituent-based SRL, where syntactic constraints are widely used during decoding (He et al., 2017, 2018a). And requires a robust syntactic parser to analyze (out-of-domain) unlabeled sentences for consistency. Our model does not rely on external tools, and is generally applicable across semantic role representations based on dependencies or constituents (i.e., phrases or spans). However, we leave experiments on the latter for future work.

Although the focus of this work has been on semi-supervised learning, we have developed a competitive SRL system which could be used on its own, after being trained on labeled data. Following previous work (Strubell et al., 2018; Cai et al., 2018) we proposed an end-to-end model, which is able to distinguish predicates and label their arguments while learning a POS-tagger and a dependency parser. Importantly, our model can be simultaneously used for supervised and semi-supervised learning without modification. Cai et al. (2018) do not use any syntactic information which is critical when dealing with unlabeled data. Strubell et al. (2018) directly predict POS tags and predicates on top of the lower layers of their model; while this information is fed to the final SRL classifier, it is not propagated through the network, and is not shared with their multi-head self-attention layers.

5 Conclusions

In this paper we developed an end-to-end SRL model and demonstrated it can effectively leverage unlabeled data under the crossview training modeling paradigm. Experiments on the CONLL-2009 benchmark datasets show that our model delivers state of the art performance in English, Chi-

nese, German, and Spanish. Directions for future work are many and varied. We would like to apply the proposed model in low-resource settings, e.g., to transfer roles from English to another language via annotation projection or to learn an SRL model from weak supervision where only annotations for dependency labels are available.

Acknowledgments

We thank the anonymous reviewers for their helpful feedback and suggestions. We gratefully acknowledge the support of the European Research Council (award number 681760, "Translating Multiple Modalities into Text").

References

Wilker Aziz, Miguel Rios, and Lucia Specia. 2011. Shallow semantic trees for SMT. In *Proceedings of the Sixth Workshop on Statistical Machine Translation*, pages 316–322, Edinburgh, Scotland.

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.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Jiaxun Cai, Shexia He, Zuchao Li, and Hai Zhao. 2018. A full end-to-end semantic role labeler, syntactic-agnostic over syntactic-aware? In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2753–2765, Santa Fe, New Mexico, USA.

Ciprian Chelba, Tomas Mikolov, Mike Schuster, Qi Ge, Thorsten Brants, Phillipp Koehn, and Tony Robinson. 2013. One billion word benchmark for measuring progress in statistical language modeling. *arXiv preprint arXiv:1312.3005*.

Jason Chiu and Eric Nichols. 2016. Named entity recognition with bidirectional LSTM-CNNs. *Transactions of the Association for Computational Linguistics*, 4:357–370.

Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2011. An analysis of open information extraction based on semantic role labeling. In *Proceedings of the 6th International Conference on Konwledge Capture*, pages 113–119, Banff, Canada.

Kevin Clark, Minh-Thang Luong, Christopher D. Manning, and Quoc Le. 2018. Semi-supervised sequence modeling with cross-view training. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1914–1925, Brussels, Belgium.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Danilo Croce, Cristina Giannone, Paolo Annesi, and Roberto Basili. 2010. Towards open-domain semantic role labeling. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 237–246, Uppsala, Sweden.

Koen Deschacht and Marie-Francine Moens. 2009. Semi-supervised semantic role labeling using the Latent Words Language Model. In *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, pages 21–29, Singapore.

Chris Dyer, Miguel Ballesteros, Wang Ling, Austin Matthews, and Noah A. Smith. 2015. Transition-based dependency parsing with stack long short-term memory. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 334–343, Beijing, China.

Nicholas FitzGerald, Oscar Täckström, Kuzman Ganchev, and Dipanjan Das. 2015. Semantic role labeling with neural network factors. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 960–970, Lisbon, Portugal.

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

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA.

Luheng He, Kenton Lee, Omer Levy, and Luke Zettlemoyer. 2018a. Jointly predicting predicates and arguments in neural semantic role labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 364–369, Melbourne, Australia.

Luheng He, Kenton Lee, Mike Lewis, and Luke Zettlemoyer. 2017. Deep semantic role labeling: What works and what's next. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 473–483, Vancouver, Canada.

Shexia He, Zuchao Li, Hai Zhao, and Hongxiao Bai. 2018b. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*

(Volume 1: Long Papers), pages 2061–2071, Melbourne, Australia.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9:1735–1780.

Atif Khan, Naomie Salim, and Yogan Jaya Kumar. 2015. A framework for multi-document abstractive summarization based on semantic role labelling. *Applied Soft Computing*, 30:737–747.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Tao Lei, Yuan Zhang, Lluís Màrquez, Alessandro Moschitti, and Regina Barzilay. 2015. High-order low-rank tensors for semantic role labeling. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1150–1160, Denver, Colorado.

Zuchao Li, Shexia He, Jiaxun Cai, Zhuosheng Zhang, Hai Zhao, Gongshen Liu, Linlin Li, and Luo Si. 2018. A unified syntax-aware framework for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2401–2411, Brussels, Belgium.

Wang Ling, Chris Dyer, Alan W Black, and Isabel Trancoso. 2015. Two/too simple adaptations of word2vec for syntax problems. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1299–1304, Denver, Colorado.

Xuezhe Ma and Eduard Hovy. 2016. End-to-end sequence labeling via bi-directional LSTM-CNNs-CRF. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1064–1074, Berlin, Germany.

Diego Marcheggiani, Joost Bastings, and Ivan Titov. 2018. Exploiting semantics in neural machine translation with graph convolutional networks. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 486–492, New Orleans, Louisiana.

Diego Marcheggiani, Anton Frolov, and Ivan Titov. 2017. A simple and accurate syntax-agnostic neural model for dependency-based semantic role labeling. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 411–420, Vancouver, Canada.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1506–1515, Copenhagen, Denmark.

Sanket Vaibhav Mehta, Jay Yoon Lee, and Jaime Carbonell. 2018. Towards semi-supervised learning for deep semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4958–4963, Brussels, Belgium.

Eric W Noreen. 1989. *Computer-intensive methods for testing hypotheses*. Wiley New York.

Hao Peng, Sam Thomson, and Noah A. Smith. 2017. Deep multitask learning for semantic dependency parsing. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2037–2048, Vancouver, Canada.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana.

Michael Roth and Mirella Lapata. 2016. Neural semantic role labeling with dependency path embeddings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1192–1202, Berlin, Germany.

Anders Søgaard and Yoav Goldberg. 2016. Deep multi-task learning with low level tasks supervised at lower layers. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 231–235, Berlin, Germany.

Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. 2018. Linguistically-informed self-attention for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5027–5038, Brussels, Belgium.

Swabha Swayamdipta, Sam Thomson, Chris Dyer, and Noah A Smith. 2017. Frame-semantic parsing with softmax-margin segmental RNNs and a syntactic scaffold. *arXiv preprint arXiv:1706.09528*.

Ivan Titov and Alexandre Klementiev. 2012. Semi-supervised semantic role labeling: Approaching from an unsupervised perspective. In *Proceedings of COLING 2012*, pages 2635–2652, Mumbai, India.