Multiplicative Representations for Unsupervised Semantic Role Induction

Yi Luan[†♣] **Yangfeng Ji**[◇] **Hannaneh Hajishirzi**[†] **Boyang Li**[♣]
[†]Department of Electrical Engineering, University of Washington
[◇]School of Interactive Computing, Georgia Institute of Technology
[♣]Disney Research
{luanyi,hannaneh}@uw.edu, jiyfeng@gatech.edu, boyang.li@disney.com

Abstract

In unsupervised semantic role labeling, identifying the role of an argument is usually informed by its dependency relation with the predicate. In this work, we propose a neural model to learn argument embeddings from the context by explicitly incorporating dependency relations as multiplicative factors, which bias argument embeddings according to their dependency roles. Our model outperforms existing state-of-the-art embeddings in unsupervised semantic role induction on the CoNLL 2008 dataset and the SimLex999 word similarity task. Qualitative results demonstrate our model can effectively bias argument embeddings based on their dependency role.

1 Introduction

Semantic role labeling (SRL) aims to identify predicate-argument structures of a sentence. The following example shows the arguments labeled with the roles A0 (typically the agent of an action) and A1 (typically the patient of an action), as well as the predicate in bold.

[Little Willy $_{A0}$] **broke** [a window $_{A1}$].

As manual annotations are expensive and time-consuming, supervised approaches (Gildea and Jurafsky, 2002; Xue and Palmer, 2004; Pradhan et al., 2005; Punyakanok et al., 2008; Das et al., 2010; Das et al., 2014) to this problem are held back by limited coverage of available gold annotations (Palmer and Sporleder, 2010). SRL performance decreases remarkably when applied to out-of-domain data (Pradhan et al., 2008).

Unsupervised SRL offer a promising alternative (Lang and Lapata, 2011; Titov and Klementiev, 2012; Garg and Henderson, 2012; Lang and Lapata, 2014; Titov and Khoddam, 2015). It is commonly formalized as a clustering problem, where each cluster represents an induced semantic role. Such clustering is usually performed through manually defined semantic and syntactic features defined over argument instances. However, the representation based on these features are usually sparse and difficult to generalize.

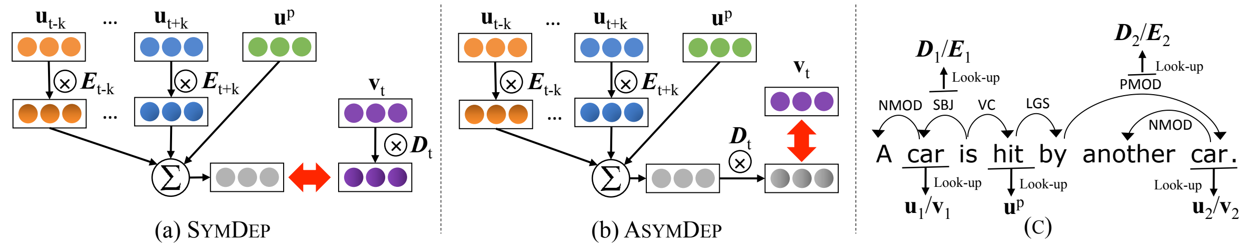
Inspired by the recent success of distributed word representations (Mikolov et al., 2013; Levy and Goldberg, 2014; Pennington et al., 2014), we introduce two unsupervised models that learn embeddings of arguments, predicates, and syntactic dependency relations between them. The embeddings are learned by predicting each argument from its context, which includes the predicate and other arguments in the same sentence. Driven by the importance of syntactic dependency relations in SRL, we explicitly model dependencies as multiplicative factors in neural networks, yielding more succinct models than existing representation learning methods employing dependencies (Levy and Goldberg, 2014; Woodsend and Lapata, 2015). The learned argument embeddings are then clustered and are evaluated by the clusters' agreement with ground truth labels.

On unsupervised SRL, our models outperform the state of the art by Woodsend and Lapata (2015) on gold parses and Titov and Khoddam (2015) on automatic parses. Qualitative results suggest our model is effective in biasing argument embeddings toward a specific dependency relation.

2 Related Work

There has been growing interest in using neural networks and representation learning for supervised and unsupervised SRL (Collobert et al., 2011; Hermann et al., 2014; Zhou and Xu, 2015;

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 118–123,
Berlin, Germany, August 7-12, 2016. ©2016 Association for Computational Linguistics



(a) SYMDEP

(b) ASYMDEP

(c)

Figure 1: (a): The SYMDEP model. (b): The ASYMDEP model. (c): An example of how embeddings relate to the parse tree. In SYMDEP, the biasing of dependency is uniformly applied to all argument embeddings. In ASYMDEP, they are concentrated on one side of the dot product.

FitzGerald et al., 2015). Closely related to our work, Woodsend and Lapata (2015) concatenate one hot features of dependency, POS-tag and a distributed representation for head word and project the concatenation onto a dense feature vector space. Instead of using dependency relations as one-hot vectors, we explicitly model the multiplicative compositionality between arguments and dependencies, and investigate two different compositionality configurations.

Our model is related to Levy and Goldberg (2014) who use dependency relations in learning word embeddings. In comparison, our models separate the representation of dependency relations and arguments, thereby allow the same word in different relations to share weights in order to reduce model parameters and data sparsity.

3 Approach

Most unsupervised approaches to SRL perform the following two steps: (1) identifying the arguments of the predicate and (2) assigning arguments to unlabeled roles, such as argument clusters. Step (1) can be usually tackled with heuristic rules (Lang and Lapata, 2014). In this paper, we focus on tackling step (2) by creating clusters of arguments that belongs to the same semantic role. As we assume PropBank-style roles (Kingsbury and Palmer, 2002), our models allocate a separate set of role clusters for each predicate and assign its arguments to the clusters. We evaluate the results by the overlapping between the induced clusters and PropBank-style gold labels.

The example below suggests that SRL requires more than just lexical embeddings.

[A car $_{A1}$] is **hit** by [another car $_{A0}$].

The A0 and A1 roles are very similar lexically, but their dependency relations to the predicate differ. To allow the same lexical embedding to shift according to different relations to the predicate, we propose the following models.

3.1 Models

Following the framework of CBOW (Mikolov et al., 2013), our models predict an argument by its context, which includes surrounding arguments and the predicate.

Let $\boldsymbol{v}_t$ be the embedding of the t^{th} argument in a sentence, and $\boldsymbol{u}_t$ the embedding of the argument when it is part of the context. Let $\boldsymbol{u}^p$ be the embedding of the predicate. $\boldsymbol{u}^c = \{\boldsymbol{u}_{t-k}, \ldots, \boldsymbol{u}_{t-1}, \boldsymbol{u}_{t+1}, \ldots, \boldsymbol{u}_{t+k}\}$ are the vectors surrounding the t^{th} argument with a window of size k.[1] The prediction of the t^{th} argument is:

$$p(\boldsymbol{v}_t | \boldsymbol{u}^p, \boldsymbol{u}^c) \propto \exp(\boldsymbol{f}(\boldsymbol{v}_t)^\intercal \boldsymbol{g}(\boldsymbol{u}^p, \boldsymbol{u}^c)) \qquad (1)$$

where $\boldsymbol{f}(\cdot)$ and $\boldsymbol{g}(\cdot)$ are two transformation functions of the target argument embedding and context vectors respectively.

We further associate a dependency relation with each argument (explained in more details in §4.1). Let matrix $\boldsymbol{D}_t$ encode the biasing effect of the dependency relation between the t^{th} argument and its predicate, and $\boldsymbol{E}_t$ be the corresponding dependency matrix for the t^{th} argument if it is used as a context. We define a $\otimes$ operator:

$$\begin{aligned} \boldsymbol{v}_t \otimes \boldsymbol{D}_t &\triangleq \tanh\left(\boldsymbol{D}_t \boldsymbol{v}_t\right) \\ \boldsymbol{u}_t \otimes \boldsymbol{E}_t &\triangleq \tanh\left(\boldsymbol{E}_t \boldsymbol{u}_t\right), \end{aligned} \qquad (2)$$

where $\tanh(\cdot)$ is the element-wise tanh function. Eq. 2 composes an argument and its dependency with a multiplicative nonlinear operation. The multiplicative formulation encourages the decoupling of dependencies and arguments, which is

[1] To be precise, the embeddings are indexed by the arguments, which are then indexed by their positions, like $\boldsymbol{u}_{w(t)}$. Here we omit w. The same convention applies to dependency matrices, which are indexed by the dependency label first.

useful in learning representations tightly focused on lexical and relational semantics, respectively.

Symmetric-Dependency. In our first model, we apply the dependency multiplication to all arguments. We have

$$\boldsymbol{f}_1(\boldsymbol{v}_t) = \boldsymbol{v}_t \otimes \boldsymbol{D}_t \tag{3}$$

$$\boldsymbol{g}_1(\boldsymbol{u}^p, \boldsymbol{u}_c) = \boldsymbol{u}^p \otimes \boldsymbol{E}^p + \sum_{\boldsymbol{u}_i \in \boldsymbol{u}^c} \boldsymbol{u}_i \otimes \boldsymbol{E}_i \tag{4}$$

This model is named Symmetric-Dependency (SymDep) for the symmetric use of $\otimes$. Since the predicate does not have an dependency with itself, we let $\boldsymbol{E}^p = \boldsymbol{I}$. Generally, $\forall i, \boldsymbol{E}_i \neq \boldsymbol{I}$.

Asymmetric-Dependency. An alternative model is to concentrate the dependency relations' effects by shifting the dependency of the predicted argument from $\boldsymbol{f}(\cdot)$ to $\boldsymbol{g}(\cdot)$, thereby move all $\otimes$ operations to construct context vector:

$$\boldsymbol{g}_2(\boldsymbol{u}^p, \boldsymbol{u}^c) = (\boldsymbol{u}^p \otimes \boldsymbol{E}^p + \sum_{\boldsymbol{u}_i \in \boldsymbol{u}^c} \boldsymbol{u}_i \otimes \boldsymbol{E}_i) \otimes \boldsymbol{D}_t \tag{5}$$

$$\boldsymbol{f}_2(\boldsymbol{v}_t) = \boldsymbol{v}_t \tag{6}$$

This model is named Asymmetric-Dependency or AsymDep. Figure 1 shows the two models side by side. Note that Eq. 5 actually defines a feed-forward neural network structure $\boldsymbol{g}_2(\boldsymbol{u}^p, \boldsymbol{u}^c)$ for predicting arguments. Consider the prediction function defined in Eq. 1, these two models will be equivalent if we eliminate all nonlinearities introduced by $\tanh(\cdot)$.

3.2 Clustering Arguments

In the final step of semantic role induction, we perform agglomerative clustering on the learned embeddings of arguments. We first create a number of seed clusters based on syntactic positions (Lang and Lapata, 2014), which are hierarchically merged. Similar to Lang and Lapata (2011), we define the similarity between clusters as the cosine similarity ($CosSim$) between the centroids with a penalty for clustering two arguments from the same sentence into the same role. Consider two clusters C and C' with the centroids $\boldsymbol{x}$ and $\boldsymbol{y}$ respectively, their similarity is:

$$S(C, C') = CosSim(\boldsymbol{x}, \boldsymbol{y}) - \alpha \cdot pen(C, C') \tag{7}$$

where α is heuristically set to 1.

To compute the penalty, let $V(C, C')$ be the set of arguments $a_i \in C$ such that a_i appears in the same sentence with another argument $a_j \in C'$. We have

$$pen(C, C') = \frac{|V(C, C')| + |V(C', C)|}{|C| + |C'|} \tag{8}$$

where $|\cdot|$ is set cardinality. When this penalty is large, the clusters C and C' will appear dissimilar, so it becomes difficult to merge them into the same cluster, preventing a_i and a_j from appearing in the same cluster.

4 Experiments

We evaluate our models in unsupervised SRL and compare the effectiveness our approach in modeling dependency relations with the previous work.

4.1 Setup

Our models are trained on 24 million tokens and 1 million sentences from the North American News Text corpus (Graff, 1995). We use MATE (Björkelund et al., 2009) to parse the dependency tree and identify predicates and arguments. Embeddings of head words are the only feature we use in clustering. Dependency matrices are restricted to contain only diagonal terms. The vocabulary sizes for arguments and predicates are 10K and 5K respectively. We hand-picked the dimension of embeddings to be 50 for all models.

We take the first dependency relation on the path from an argument's head word to the predicate as its dependency label, considering the dependency's direction. For example, the label for the first *car* in Figure 1(c) is SBJ^{-1}. We use negative sampling (Mikolov et al., 2013) to approximate `softmax` in the objective function. For SymDep, we sample both the predicted argument and dependency. For AsymDep, we sample only the argument. Models are trained using AdaGrad (Duchi et al., 2011) with L2 regularization. All embeddings are randomly initialized.[2]

Baselines. We compare against several baselines using representation learning: CBOW and Skip-Gram (Mikolov et al., 2013), GloVe (Pennington et al., 2014), L&G (Levy and Goldberg, 2014) and Arg2vec (Woodsend and Lapata, 2015). Similar to ours, L&G and Arg2vec both encode dependency relations in the embeddings. We train all models on the same dataset as ours using publicly avail-

[2] Resulted embeddings can be downloaded from `https://bitbucket.org/luanyi/unsupervised-srl`.

able code[3], and then apply the same clustering algorithm. Introduced by Lang and Lapata (2014), SyntF is a strong baseline that clusters arguments based on purely syntactic cues: voice of the verb, relative position to the predicate, syntactic relations, and realizing prepositions. The window size for Arg2vec and our models are set to 1, while all other embeddings are set to 2. We also employ two state-of-the-art methods from Titov and Klementiev (2012) (T&K12) and Titov and Khoddam (2015) (T&K15).

4.2 SRL Results

Following common practices (Lang and Lapata, 2014), we measure the overlap of induced semantic roles and their gold labels on the CoNLL 2008 training data (Surdeanu et al., 2008). We report purity (PU), collocation (CO), and their harmonic mean (F1) evaluated on gold arguments in two settings of gold parses and automatic parses from the MaltParser (Nivre et al., 2007). Table 1 shows the results.[4]

	Gold parses			Automatic parses		
Model	PU	CO	F1	PU	CO	F1
SyntF	81.6	78.1	79.8	77.0	71.5	74.1
Skip-Gram	86.6	74.7	80.2	84.3	72.4	77.9
CBOW	84.6	74.9	79.4	84.0	71.5	77.2
GloVe	84.9	74.1	79.2	83.0	70.8	76.5
L&G	87.0	75.6	80.9	86.6	71.3	78.2
Arg2vec	84.0	77.7	80.7	86.9	71.4	78.4
SymDep	85.3	77.9	81.4	81.9	76.6	**79.2**
AsymDep	85.6	78.3	**81.8**	82.9	75.2	78.9
T&K12	88.7	78.1	**83.0**	86.2	72.7	78.8
T&K15	79.7	86.2	82.8	-	-	-
Sym𝟙Dep	83.8	77.4	80.5	82.3	74.8	78.4

Table 1: Purity, collocation and F1 measures for the CoNLL-2008 data set.

SymDep and AsymDep outperform all representation learning baselines for SRL. T&K12 outperforms our models on gold parsing because they use a strong generative clustering method, which shared parameters across verbs in the clustering step. In addition, T&K15 incorporates feature-rich latent structure learning. Nevertheless, our models perform better with automatic parses, indicating the robustness of our models under noise in automatic parsing. Future work involves more sophisticated clutering techniques (Titov and Klementiev, 2012) as well as incorporating feature-rich models (Titov and Khoddam, 2015) to improve performance further.

Table 1 shows that including dependency relations (L&G, Arg2vec, SymDep, and AsymDep) improves performance. Additionally, our models achieve the best performance among those, showing the strength of modeling dependencies as multiplicative factors. Arg2vec learns word embedings from the context features which are concatenation of syntactic features (dependency reations and POS tags) and word embedings. L&G treats each word-dependency pair as a separate token, leading to a large vocabulary (142k in our dataset) and potentially data scarcity. In comparison, SymDep and AsymDep formulate the dependency as the weight matrix of the second nonlinear layer, leading to a deeper structure with less parameters compared to previous work.

Qualitative results. Table 2 demonstrates the effectiveness of our models qualitatively. For example, we identify that *car* is usually the subject of *crash* and *unload*, and the object of *sell* and *purchase*. In comparison, CBOW embeddings do not reflect argument-predicate relations.

Ablation Study. To further understand the effects of the multiplicative representation on unsupervised SRL, we create an ablated model Sym𝟙Dep, where we force all dependencies in SymDep to use the same matrix. The network has the same structure as SymDep, but the dependency information is removed. Its performance on SRL is shown at the bottom of Table 1. Sym𝟙Dep performs slightly worse than Arg2vec. This suggests that the performance gain in SymDep can be attributed to the use of dependency information instead of the way of constructing context.

4.3 Word Similarity Results

As a further evaluation of the learned embeddings, we test if similarities between word embeddings agree with human annotation from SimLex999 (Hill et al., 2015). Table 3 shows that SymDep outperforms Arg2vec on both nouns and verbs, suggesting multiplicative dependency relations are indeed effective. However, AsymDep performs better than SymDep on noun similarity but much worse on verb similarity. We explore this further in an ablation study.

[3] Except that Arg2vec is reimplemented since there is no public code online.

[4] The numbers reported for Arg2vec with gold parsing (80.7) is different from Woodsend and Lapata (2015) (80.9) since we use a different clustering method and different training data.

Argument	SymDep (SBJ)	SymDep (OBJ)	CBOW
car	*crash*, roar, capsize, land, lug, *unload*, bounce, ship	*sell*, *purchase*, buy, retrieve, board, haul, lease, unload	train, splash, mail, shelter, jet, ferry, drill, ticket
victim	injure, die, protest, complain, weep, hospitalize, shout, suffer	insult, assault, stalk, avenge, harass, interview, housing, apprehend	void, murder, kidnap, widow, massacre, surge, sentence, defect
teacher	teach, mentor, educate, note, reminisce, say, learn, lecture	hire, bar, recruit, practice, assault, enlist, segregate, encourage	coach, mentor, degree, master, guide, pilot, partner, captain
student	learn, resurface, object, enroll, note, protest, deem, teach	teach, encourage, educate, assault, segregate, enroll, attend, administer	graduate, degree, mortgage, engineer, mentor, pilot, partner, pioneer

Table 2: The 8 most similar predicates to a given argument in a given dependency role.

Model	Nouns	Verbs
L&G	31.4	27.2
Arg2vec	38.2	31.4
SymDep	39.2	**36.5**
AsymDep	**39.7**	15.3
Asym$\mathbb{1}$Dep	33.2	24.2

Table 3: A POS-based analysis of the various embeddings. Numbers are the Spearman's ρ scores of each model on nouns and verbs of SimLex999.

Ablation Study. We create an ablated model to explore the reason for AsymDep's performance on verb similarity. Asym$\mathbb{1}$Dep is based on AsymDep where we force all dependency relations for the predicted argument $\boldsymbol{v}_t$ to use the same matrix $\boldsymbol{D}_i$. The aim of this experiment is to check the negative influence of asymmetric dependency matrix to verb embedding. The results are shown at the bottom of Table 3. By keeping $\boldsymbol{D}_i$ dependency independent, performance on verbs is significantly improved with the cost of noun performance.

5 Conclusions

We present a new unsupervised semantic role labeling approach that learns embeddings of arguments by predicting each argument from its context and considering dependency relation as a multiplicative factor. Two proposed neural networks outperform current state-of-the-art embeddings on unsupervised SRL and the SimLex999 word similarity task. As an effective model for dependency relations, our multiplicative argument-dependency factor models encourage the decoupling of argument and dependency representations. Disentangling linguistic factors in similar manners may be worth investigating in similar tasks such as frame semantic parsing and event detection.

References

Anders Björkelund, Love Hafdell, and Pierre Nugues. 2009. Multilingual semantic role labeling. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task*, pages 43–48.

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.

Dipanjan Das, Nathan Schneider, Desai Chen, and Noah A. Smith. 2010. Probabilistic frame-semantic parsing. In *Proceedings of the Human Language Technologies Conference of the North American Chapter of the Associattion for Computational Linguistics*, pages 948–956.

Dipanjan Das, Desai Chen, André FT Martins, Nathan Schneider, and Noah A. Smith. 2014. Frame-semantic parsing. *Computational Linguistics*, 40(1):9–56.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *The Journal of Machine Learning Research*, 12:2121–2159.

Nicholas FitzGerald, Oscar Tckstrm, 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*.

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

Nikhil Garg and James Henderson. 2012. Unsupervised semantic role induction with global role ordering. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 145–149.

Daniel Gildea and Daniel Jurafsky. 2002. Automatic labeling of semantic roles. *Computational Linguistics*, 28(9):245–288.

David Graff. 1995. North american news text corpus.

Karl Moritz Hermann, Dipanjan Das, Jason Weston, and Kuzman Ganchev. 2014. Semantic frame identification with distributed word representations. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*.

Felix Hill, Roi Reichart, and Anna Korhonen. 2015. Simlex-999: Evaluating semantic models with (genuine) similarity estimation. *Computational Linguistics*.

Paul Kingsbury and Martha Palmer. 2002. From TreeBank to PropBank. In *Proceedings of the Third International Conference on Language Resources and Evaluation*, pages 1989–1993. Las Palmas.

Meghana Kshirsagar, Sam Thomson, Nathan Schneider, Jaime Carbonell, Noah A. Smith, and Chris Dyer. 2015. Frame-semantic role labeling with heterogeneous annotations. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, Beijing, China.

Joel Lang and Mirella Lapata. 2011. Unsupervised semantic role induction via split-merge clustering. In *Proceedings of Human Language Technologies Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1117–1126.

Joel Lang and Mirella Lapata. 2014. Similarity-driven semantic role induction via graph partitioning. *Computational Linguistics*, 40(3):633–669.

Omer Levy and Yoav Goldberg. 2014. Dependencybased word embeddings. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, volume 2, pages 302–308.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS*.

Joakim Nivre, Johan Hall, Jens Nilsson, Atanas Chanev, Gülsen Eryigit, Sandra Kübler, Svetoslav Marinov, and Erwin Marsi. 2007. Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Alexis Palmer and Caroline Sporleder. 2010. Evaluating FrameNet-style semantic parsing: the role of coverage gaps in FrameNet. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 928–936.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Sameer Pradhan, Kadri Hacioglu, Wayne Ward, James H. Martin, and Daniel Jurafsky. 2005. Semantic role chunking combining complementary syntactic views. In *Proceedings of the Ninth Conference on Computational Natural Language Learning*, pages 217–220.

Sameer S. Pradhan, Wayne Ward, and James H. Martin. 2008. Towards robust semantic role labeling. *Computational Linguistics*, 34(2):289–310.

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

Mihai Surdeanu, Richard Johansson, Adam Meyers, Lluís Màrquez, and Joakim Nivre. 2008. The conll-2008 shared task on joint parsing of syntactic and semantic dependencies. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning*, CoNLL '08, pages 159–177.

Ivan Titov and Ehsan Khoddam. 2015. Unsupervised induction of semantic roles within a reconstruction-error minimization framework. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics*.

Ivan Titov and Alexandre Klementiev. 2012. A Bayesian approach to unsupervised semantic role induction. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 12–22.

Kristian Woodsend and Mirella Lapata. 2015. Distributed representations for unsupervised semantic role labeling. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Nianwen Xue and Martha Palmer. 2004. Calibrating features for semantic role labeling. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*, pages 88–94.

Jie Zhou and Wei Xu. 2015. End-to-end learning of semantic role labeling using recurrent neural networks. In *Proceedings of the 53rd Conference of the Association for Computational Linguistics*, pages 1127–1137.