

Halo: Learning Semantics-Aware Representations for Cross-Lingual Information Extraction

Hongyuan Mei*, Sheng Zhang*, Kevin Duh, Benjamin Van Durme
Center for Language and Speech Processing, Johns Hopkins University
{hmei, s.zhang, kevinduh, vandurme}@cs.jhu.edu

Abstract

Cross-lingual information extraction (CLIE) is an important and challenging task, especially in low resource scenarios. To tackle this challenge, we propose a training method, called *Halo*, which enforces the local region of each hidden state of a neural model to only generate target tokens with the same semantic structure tag. This simple but powerful technique enables a neural model to learn semantics-aware representations that are robust to noise, without introducing any extra parameter, thus yielding better generalization in both high and low resource settings.

1 Introduction

Cross-lingual information extraction (CLIE) is the task of distilling and representing factual information in a target language from the textual input in a source language (Sudo et al., 2004; Zhang et al., 2017b). For example, Fig. 1 illustrates a pair of input Chinese sentence and its English predicate-argument information¹, where predicate and argument are well used *semantic structure tags*.

It is of great importance to solve the task, as to provide viable solutions to extracting information from the text of languages that suffer from no or little existing information extraction tools. Neural models have empirically proven successful in this task (Zhang et al., 2017b,c), but still remain unsatisfactory in low resource (i.e. small number of training samples) settings. These neural models learn to summarize a given source sentence and target prefix into a hidden state, which aims to generate the correct next target token after being

*equal contribution

¹The predicate-argument information is usually denoted by relation tuples. In this work, we adopt the tree-structured representation generated by PredPatt (White et al., 2016; Zhang et al., 2017d), which was a lightweight tool available at <https://github.com/hltcoe/PredPatt>.

士兵们开始发射迫击炮。
(Soldiers started firing mortars.)

(a)

[(Soldiers:a) started:p [(Soldiers:a) firing:p (mortars:a)]]
(b)

Figure 1: Example of cross-lingual information extraction: Chinese input text (a) and linearized English PredPatt output (b), where ‘:p’ and blue stand for predicate while ‘:a’ and purple denote argument.

passed through an output layer. As each member in the target vocabulary is essentially either predicate or argument, a random perturbation on the hidden state should still be able to yield a token with the same semantic structure tag. This inductive bias motivates an extra term in training objective, as shown in Fig. 2, which enforces the surroundings of any learned hidden state to generate tokens with the same semantic structure tag (either predicate or argument) as the centroid. We call this technique *Halo*, because the process of each hidden state taking up its surroundings is analogous to how the halo is formed around the sun. The method is believed to help the model generalize better, by learning more semantics-aware and noise-insensitive hidden states without introducing extra parameters.

2 The Problem

We are interested in learning a probabilistic model that directly maps an input sentence $\{x_i\}_{i=1}^I = x_1x_2 \dots x_I$ of the source language \mathcal{S} into an output sequence $\{y_t\}_{t=1}^T = y_1y_2 \dots y_T$ of the target language \mathcal{T} , where \mathcal{S} can be any human natural language (e.g. Chinese) and \mathcal{T} is the English PredPatt (White et al., 2016). In the latter vocabulary, each type is tagged as either predicate or argument—those with “:p” are predicates while those with “:a” are arguments.

For any distribution P in our proposed family, the *log-likelihood* ℓ of the model P given any

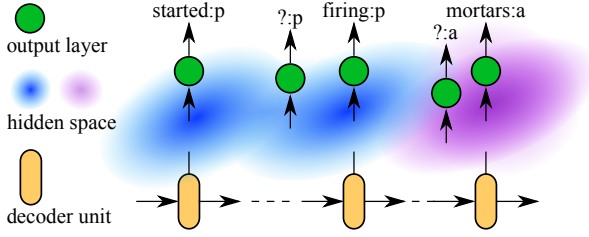


Figure 2: Visualization of *Halo* method. While a neural model learns to summarize the current known information into a hidden state and predict the next target token, the surroundings of this hidden state in the same space (two-dimensional in this example) are supervised to generate tokens with the same semantic structure tag. For example, at the last shown step, the centroid of purple area is the summarized hidden state and learns to predict ‘mortars:a’, while a randomly sampled neighbor is enforced to generate an argument, although it may not be ‘mortars’ (thus denoted by ‘?’). Similar remarks apply to the blue regions.

$(\{y_t\}_{t=1}^T \mid \{x_i\}_{i=1}^I)$ pair is:

$$\sum_{t=1}^T \log P(y_t \mid y_{t-1}, \dots, y_0, \{x_i\}_{i=1}^I) \quad (1)$$

where y_0 is a special beginning of sequence token.

We denote vectors by bold lowercase Roman letters such as \mathbf{h} , and matrices by bold capital Roman letters such as \mathbf{W} throughout the paper. Subscripted bold letters denote distinct vectors or matrices (e.g., \mathbf{p}_t). Scalar quantities, including vector and matrix elements such as h_d and p_{t,y_t} , are written without bold. Capitalized scalars represent upper limits on lowercase scalars, e.g., $1 \leq d \leq D$. Function symbols are notated like their return type. All $\mathbb{R} \rightarrow \mathbb{R}$ functions are extended to apply elementwise to vectors and matrices.

3 The Method

In this section, we first briefly review how the baseline neural encoder-decoder models work on this task, and then introduce our novel and well-suited training method *Halo*.

3.1 Baseline Neural Models

Previous neural models on this task (Zhang et al., 2017b,c) all adopt an encoder-decoder architecture with recurrent neural networks, particularly LSTMs (Hochreiter and Schmidhuber, 1997). At each step t in decoding, the models summarize the input $\{x_i\}_{i=1}^I$ and output prefix y_1, \dots, y_{t-1} into a hidden state $\mathbf{h}_t \in (-1, 1)^D$, and then project it with a transformation matrix $\mathbf{W} \in \mathbb{R}^{|\mathcal{V}| \times D}$ to a distribution \mathbf{p}_t over the target English PredPat

vocabulary \mathcal{V} :

$$\mathbf{p}_t = \mathbf{o}_t / (\mathbf{1}^\top \mathbf{o}_t) \quad (2a)$$

$$\mathbf{o}_t = \exp \mathbf{W} \mathbf{h}_t \in \mathbb{R}_+^{|\mathcal{V}|} \quad (2b)$$

where $\mathbf{1}$ is a $|\mathcal{V}|$ -dimensional one vector such that \mathbf{p}_t is a valid distribution.

Suppose that the ground truth target token at this step is y_t , the probability of generating y_t under the current model is p_{t,y_t} , obtained by accessing the y_t -th element in the vector \mathbf{p}_t . Then the log-likelihood is constructed as $\ell = \sum_{t=1}^T \log p_{t,y_t}$, and the model is trained by maximizing this objective over all the training pairs.

3.2 Halo

Our method adopts a property of this task—the vocabulary \mathcal{V} is partitioned into \mathcal{P} , set of predicates that end with “:p”, and \mathcal{A} , set of arguments that end with “:a”. As a neural model would summarize everything known up to step t into \mathbf{h}_t , would a perturbation \mathbf{h}'_t around \mathbf{h}_t still generate the same token y_t ? This bias seems too strong, but we can still reasonably assume that \mathbf{h}'_t would generate a token with the same semantic structure tag (i.e. predicate or argument). That is, the prediction made by \mathbf{h}'_t should end with “:p” if y_t is a predicate, and with “:a” otherwise.

This inductive bias provides us with another level of supervision. Suppose that at step t , a neighboring \mathbf{h}'_t is randomly sampled around \mathbf{h}_t , and is then used to generate a distribution \mathbf{p}'_t in the same way as equation (2). Then we can get a distribution q'_t over $\mathcal{C} = \{\text{predicate, argument}\}$, by summing all the probabilities of predicates and those of arguments:

$$q'_{t,\text{predicate}} = \sum_{v \in \mathcal{P}} p'_{t,v} \quad (3a)$$

$$q'_{t,\text{argument}} = \sum_{v \in \mathcal{A}} p'_{t,v} \quad (3b)$$

This aggregation is shown in Fig. 3. Then the extra objective is $\ell' = \sum_{t=1}^T \log q'_{t,c_t}$, where $c_t = \text{predicate}$ if the target token $y_t \in \mathcal{P}$ (i.e. ending with “:p”) and $c_t = \text{argument}$ otherwise.

Therefore, we get the joint objective to maximize by adding ℓ and ℓ' :

$$\ell + \ell' = \sum_{t=1}^T \log p_{t,y_t} + \sum_{t=1}^T \log q'_{t,c_t} \quad (4)$$

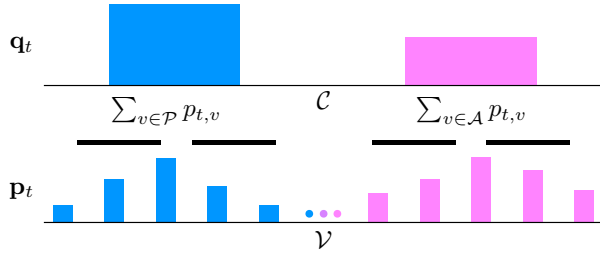


Figure 3: Visualization of how \mathbf{q}_t (distribution over \mathcal{C}) is obtained by aggregating \mathbf{p}_t (distribution over \mathcal{V}).

which enables the model to learn more semantics-aware and noise-insensitive hidden states by enforcing the hidden states within a region to share the same semantic structure tag.²

3.2.1 Sampling Neighbors

Sampling a neighbor around \mathbf{h}_t is essentially equivalent to adding noise to it. Note that in a LSTM decoder that previous work used, $\mathbf{h}_t \in (-1, 1)^D$ because $\mathbf{h}_t = \mathbf{o}_t \odot \tanh(\mathbf{c}_t)$ where $\mathbf{o}_t \in (0, 1)^D$ and $\tanh(\mathbf{c}_t) \in (-1, 1)^D$. Therefore, extra work is needed to ensure $\mathbf{h}'_t \in (-1, 1)^D$. For this purpose, we follow the recipe³:

- Sample $\mathbf{h}''_t \in (-1, 1)^D$ by independently sampling each entry from a uniform distribution over $(-1, 1)$;
- Sample a scalar $\lambda_t \in (0, 1)$ from a Beta distribution $B(\alpha, \beta)$ where α and β are hyperparameters to be tuned;
- Compute $\mathbf{h}'_t = \mathbf{h}_t + \lambda_t(\mathbf{h}''_t - \mathbf{h}_t)$ such that $\mathbf{h}'_t \in (-1, 1)^D$ lies on the line segment between \mathbf{h}_t and \mathbf{h}''_t .

Note that the sampled hidden state \mathbf{h}'_t is only used to compute \mathbf{q}'_t , but not to update the LSTM hidden state, i.e., \mathbf{h}_{t+1} is independent of \mathbf{h}'_t .

3.2.2 Roles of Hyperparameters

The *Halo* technique adds an inductive bias into the model, and its magnitude is controlled by λ_t :

- $\lambda_t \in (0, 1)$ to ensure $\mathbf{h}'_t \in (-1, 1)^D$;
- $\lambda_t \rightarrow 0$ makes $\mathbf{h}'_t \rightarrow \mathbf{h}_t$, thus providing no extra supervision on the model;

²One can also sample multiple, rather than one, neighbors for one hidden state and then average their log q'_{t,c_t} . In our experimental study, we only try one for computational cost and found it effective enough.

³Alternatives do exist. For example, one can transform \mathbf{h}_t from $(-1, 1)^D$ to $(-\infty, \infty)^D$, add random (e.g. Gaussian) noise in the latter space and then transform back to $(-1, 1)^D$. These tricks are valid as long as they find neighbors within the same space $(-1, 1)^D$ as \mathbf{h}_t is.

- $\lambda_t \rightarrow 1$ makes \mathbf{h}'_t uniformly sampled in entire $(-1, 1)^D$, and causes underfitting just like a L_2 regularization coefficient goes to infinity.

We sample a valid λ_t from a Beta distribution with $\alpha > 0$ and $\beta > 0$, and their magnitude can be tuned on the development set:

- When $\alpha \rightarrow 0$ and β is finite, or α is finite and $\beta \rightarrow \infty$, we have $\lambda_t \rightarrow 0$;
- When $\alpha \rightarrow \infty$ and β is finite, or α is finite and $\beta \rightarrow 0$, we have $\lambda_t \rightarrow 1$;
- Larger α and β yield larger variance of λ_t , and setting λ_t to be a constant is a special case that $\alpha \rightarrow \infty$, $\beta \rightarrow \infty$ and α/β is fixed.

Besides α and β , the way of partitioning \mathcal{V} (i.e. the definition of \mathcal{C}) also serves as a knob for tuning the bias strength. Although on this task, the predicate and argument tags naturally partition the vocabulary, we are still able to explore other possibilities. For example, an extreme is to partition \mathcal{V} into $|\mathcal{V}|$ different singletons, meaning that $\mathcal{C} = \mathcal{V}$ —a perturbation around \mathbf{h}_t should still predict the same token. But this extreme case does not work well in our experiments, verifying the importance of the semantic structure tags on this task.

4 Related Work

Cross-lingual information extraction has drawn a great deal of attention from researchers. Some (Sudo et al., 2004; Parton et al., 2009; Ji, 2009; Snover et al., 2011; Ji and Nothman, 2016) worked in closed domains, i.e. on a predefined set of events and/or entities, Zhang et al. (2017b) explored this problem in open domain and their attentional encoder-decoder model significantly outperformed a baseline system that does translation and parsing in a pipeline. Zhang et al. (2017c) further improved the results by inventing a hierarchical architecture that learns to first predict the next semantic structure tag and then select a tag-dependent decoder for token generation. Orthogonal to these efforts, *Halo* aims to help *all neural models on this task*, rather than any specific model architecture.

Halo can be understood as a data augmentation technique (Chapelle et al., 2001; Van der Maaten et al., 2013; Srivastava et al., 2014; Szegedy et al., 2016; Gal and Ghahramani, 2016). Such tricks have been used in training neural networks to achieve better generalization, in applications like image classification (Simard et al.,

DATASET	NUMBER OF PAIRS			VOCABULARY SIZE		TOKEN/TYPE
	TRAIN	DEV	TEST	SOURCE	TARGET	
CHINESE	941040	10000	39626	258364	234832	91.94
UZBEK	31581	1373	1373	69255	37914	12.18
TURKISH	20774	903	903	51248	32009	11.97
SOMALI	10702	465	465	29591	18616	12.78

Table 1: Statistics of each dataset.

METHOD	CHINESE			UZBEK			TURKISH			SOMALI		
	BLEU	F1		BLEU	F1		BLEU	F1		BLEU	F1	
		PRED	ARG		PRED	ARG		PRED	ARG		PRED	ARG
MODELZ	22.07	30.06	39.06	10.76	12.46	24.08	7.47	6.49	17.76	13.06	13.91	25.38
MODEL P	22.10	30.04	39.83	12.50	18.81	25.93	9.04	12.90	21.13	13.22	16.71	26.83
MODEL P- <i>Halo</i>	23.18	30.85	41.23	12.95	19.23	27.63	10.21	12.55	22.57	14.26	17.06	27.73

Table 2: BLEU and F1 scores of different models on all these datasets, where PRED stands for predicate and ARG for argument. Best numbers are highlighted as **bold**.

2000; Simonyan and Zisserman, 2015; Arpit et al., 2017; Zhang et al., 2017a) and speech recognition (Graves et al., 2013; Amodei et al., 2016). *Halo* differs from these methods because 1) it makes use of the task-specific information—vocabulary is partitioned by semantic structure tags; and 2) it makes use of the human belief that the hidden representations of tokens with the same semantic structure tag should stay close to each other. Some

5 Experiments

We evaluate our method on several real-world CLIE datasets measured by BLEU (Papineni et al., 2002) and F1, as proposed by Zhang et al. (2017b). For the generated linearized PredPatt outputs and their references, the former metric⁴ measures their n-gram similarity, and the latter measures their token-level overlap. In fact, F1 is computed separately for predicate and argument, as F1 PRED and F1 ARG respectively.

5.1 Datasets

Multiple datasets were used to demonstrate the effectiveness of our proposed method, where one sample in each dataset is a source language sentence paired with its linearized English PredPatt output. These datasets were first introduced as the DARPA LORELEI Language Packs (Strassel and Tracey, 2016), and then used for this task by Zhang et al. (2017b,c). As shown in table 1, the CHINESE dataset has almost one million training samples and a high token/type ratio, while the oth-

⁴The MOSES implementation (Koehn et al., 2007) was used as in all the previous work on this task.

ers are *low resourced*, meaning they have much fewer samples and lower token/type ratios.

5.2 Model Implementation

Before applying our *Halo* technique, we first improved the current state-of-the-art neural model of Zhang et al. (2017c) by using residual connections (He et al., 2016) and multiplicative attention (Luong et al., 2015), which effectively improved the model performance. We refer to the model of Zhang et al. (2017c) and our improved version as ModelZ and ModelP respectively⁵.

5.3 Experimental Details

In experiments, instead of using the full vocabularies shown in table 1, we set a minimum count threshold for each dataset, to replace the rare words by a special out-of-vocabulary symbol. These thresholds were tuned on dev sets.

The Beta distribution is very flexible. In general, its variance is a decreasing function of $\alpha + \beta$, and when $\alpha + \beta$ is fixed, the mean is an increasing function of α . In our experiments, we fixed $\alpha + \beta = 20$ and only lightly tuned α on dev sets. Optimal values of α stay close to 1.

5.4 Results

As shown in Table 2, ModelP outperforms ModelZ on all the datasets measured by all the metrics, except for F1 PRED on CHINESE dataset. Our *Halo* technique consistently boosts the model performance of MODEL P except for F1 PRED on TURKISH.

⁵Z stands for Zhang and P for Plus.

Additionally, experiments were also conducted on two other low resource datasets AMHARIC and YORUBA that Zhang et al. (2017c) included, and $\alpha = 0$ in *Halo* was found optimal on the dev sets. In such cases, this regularization was not helpful so no comparison need be made on the held-out test sets.

6 Conclusion and Future Work

We present a simple and effective training technique *Halo* for the task of cross-lingual information extraction. Our method aims to enforce the local surroundings of each hidden state of a neural model to only generate tokens with the same semantic structure tag, thus enabling the learned hidden states to be more aware of semantics and robust to random noise. Our method provides new state-of-the-art results on several benchmark cross-lingual information extraction datasets, including both high and low resource scenarios.

As future work, we plan to extend this technique to similar tasks such as POS tagging and Semantic Role Labeling. One straightforward way of working on these tasks is to define the vocabularies as set of ‘word-type:POS-tag’ (so $c_t = \text{POS tag}$) and ‘word-type:SR’ (so $c_t = \text{semantic role}$), such that our method is directly applicable. It would also be interesting to apply *Halo* widely to other tasks as a general regularization technique.

Acknowledgments

This work was supported in part by the JHU Human Language Technology Center of Excellence (HLTCOE), and DARPA LORELEI. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes. The views and conclusions contained in this publication are those of the authors and should not be interpreted as representing official policies or endorsements of DARPA or the U.S. Government.

References

Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. 2016. Deep speech 2: End-to-end speech recognition in english and mandarin. In *International Conference on Machine Learning*. pages 173–182.

Devansh Arpit, Stanislaw Jastrzebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxin-

der S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. 2017. A closer look at memorization in deep networks. In *International Conference on Machine Learning*. pages 233–242.

Olivier Chapelle, Jason Weston, Léon Bottou, and Vladimir Vapnik. 2001. Vicinal risk minimization. In *Advances in neural information processing systems*. pages 416–422.

Yarin Gal and Zoubin Ghahramani. 2016. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in neural information processing systems*. pages 1019–1027.

Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. 2013. Speech recognition with deep recurrent neural networks. In *Acoustics, speech and signal processing (icassp), 2013 IEEE international conference on*. IEEE, pages 6645–6649.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9(8):1735–1780.

Heng Ji. 2009. Cross-lingual predicate cluster acquisition to improve bilingual event extraction by inductive learning. In *Proceedings of the Workshop on Unsupervised and Minimally Supervised Learning of Lexical Semantics*. Association for Computational Linguistics, Boulder, Colorado, USA, pages 27–35. <http://www.aclweb.org/anthology/W09-1704>.

Heng Ji and Joel Nothman. 2016. Overview of TAC-KBP2016 Tri-lingual EDL and its impact on end-to-end cold-start KBP. In *Proceedings of the Text Analysis Conference (TAC)*.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the ACL on interactive poster and demonstration sessions*. Association for Computational Linguistics, pages 177–180.

Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, pages 311–318.

- Kristen Parton, Kathleen R. McKeown, Bob Coyne, Mona T. Diab, Ralph Grishman, Dilek Hakkani-Tür, Mary Harper, Heng Ji, Wei Yun Ma, Adam Meyers, Sara Stolbach, Ang Sun, Gokhan Tur, Wei Xu, and Sibel Yaman. 2009. [Who, what, when, where, why? comparing multiple approaches to the cross-lingual 5w task.](#) In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*. Association for Computational Linguistics, Suntec, Singapore, pages 423–431. <http://www.aclweb.org/anthology/P/P09/P09-1048>.
- Patrice Y Simard, Yann A Le Cun, John S Denker, and Bernard Victorri. 2000. Transformation invariance in pattern recognition: Tangent distance and propagation. *International Journal of Imaging Systems and Technology* 11(3):181–197.
- Karen Simonyan and Andrew Zisserman. 2015. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*.
- Matthew Snover, Xiang Li, Wen-Pin Lin, Zheng Chen, Suzanne Tamang, Mingmin Ge, Adam Lee, Qi Li, Hao Li, Sam Anzaroot, and Heng Ji. 2011. [Cross-lingual slot filling from comparable corpora.](#) In *Proceedings of the 4th Workshop on Building and Using Comparable Corpora: Comparable Corpora and the Web*. Association for Computational Linguistics, Stroudsburg, PA, USA, BUCC '11, pages 110–119. <http://dl.acm.org/citation.cfm?id=2024236.2024256>.
- Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research* 15(1):1929–1958.
- Stephanie Strassel and Jennifer Tracey. 2016. Lorelei language packs: Data, tools, and resources for technology development in low resource languages. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Sara Goggi, Marko Grobelnik, Bente Maegaard, Joseph Mariani, Helene Mazo, Asuncion Moreno, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA), Paris, France.
- Kiyoshi Sudo, Satoshi Sekine, and Ralph Grishman. 2004. Cross-lingual information extraction system evaluation. In *Proceedings of the 20th international Conference on Computational Linguistics*. Association for Computational Linguistics, page 882.
- Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. 2016. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. pages 2818–2826.
- Laurens Van der Maaten, Minmin Chen, Stephen Tyree, and Kilian Weinberger. 2013. Learning with marginalized corrupted features. In *Proceedings of The 30th International Conference on Machine Learning*.
- Aaron Steven White, Drew Reisinger, Keisuke Sakaguchi, Tim Vieira, Sheng Zhang, Rachel Rudinger, Kyle Rawlins, and Benjamin Van Durme. 2016. [Universal Decompositional Semantics on Universal Dependencies.](#) In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Austin, Texas, pages 1713–1723. <https://aclweb.org/anthology/D16-1177>.
- Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2017a. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.
- Sheng Zhang, Kevin Duh, and Benjamin Van Durme. 2017b. [MT/IE: Cross-lingual Open Information Extraction with Neural Sequence-to-Sequence Models.](#) In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*. Association for Computational Linguistics, Valencia, Spain, pages 64–70. <http://www.aclweb.org/anthology/E17-2011>.
- Sheng Zhang, Kevin Duh, and Benjamin Van Durme. 2017c. [Selective Decoding for Cross-lingual Open Information Extraction.](#) In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Asian Federation of Natural Language Processing, Taipei, Taiwan, pages 832–842. <http://www.aclweb.org/anthology/I17-1084>.
- Sheng Zhang, Rachel Rudinger, and Benjamin Van Durme. 2017d. [An Evaluation of PredPatt and Open IE via Stage 1 Semantic Role Labeling.](#) In *IWCS 2017 — 12th International Conference on Computational Semantics — Short papers*. <http://aclweb.org/anthology/W17-6944>.