Remove Noise and Keep Truth:
A Noisy Channel Model for Semantic Role Labeling

Anonymous ACL submission

Abstract

Semantic role labeling usually models structures using sequences, trees, or graphs. Past works focused on researching novel modeling methods and neural structures and integrating more features. In this paper, we re-examined the noise in neural semantic role labeling models, a problem that has been long-ignored. By proposing a noisy channel model structure, we effectively eliminate the noise in the labeling flow and thus improve performance. Without relying on additional features, our proposed novel model significantly outperforms a strong baseline on multiple popular semantic role labeling benchmarks, which demonstrates the effectiveness and robustness of our proposed model.

1 Introduction

Semantic role labeling (SRL) extracts shallow semantic structures such as agents, goals, temporal, patient/receiver, or locative arguments for predicates. It is a popular task in natural language processing and can be useful in a variety of downstream tasks, such as information extraction (Christensen et al., 2010), machine reading comprehension (Zhang et al., 2019b), and machine translation (Liu and Gildea, 2010).

SRL's development has paralleled that of syntax and transferred from constituency to dependency structures. As a result, SRL is typically subdivided into span (constituency) SRL and dependency SRL based on the argument formalism. In span SRL, arguments are the constituent spans of the sentence, while in dependency SRL, the head words of the constituent spans are the arguments.

A number of modeling approaches have been studied in recent work. SRL can be abstracted as the identification of predicates and arguments and the classification of their pairs, so SRL can be considered to a sequence-based labeling problem either by identifying/giving the predicate in advance (Zhou and Xu, 2015; Marcheggiani et al., 2017a; He et al., 2017, 2018b; Li et al., 2018) or, modeling SRL as a graph, the predicate is used as the root node of the tree, the arguments are treated as its child nodes, and the predicate-argument relationships are used for edge labels (Cai et al., 2018). In methods using pre-identified predicates, arguments are labeled one predicate at a time, while when modeling SRL as a graph, all predicates, arguments, and their pairs are identified and classified in one-shot (He et al., 2018a; Li et al., 2019). These modeling approaches, when coupled with large pre-trained language models, currently comprise the state-of-the-art SRL models.

While novel models are still introduced, few studies have focused on SRL's noise issue, an important performance bottleneck in SRL that we focus on and aim to alleviate using a noisy channel model. Neural models often introduce features that are either do not help or even actively hurt target prediction during representation encoding and neural network scoring; we call the inclusion of these features the noise issue. Given an input sentence X, an SRL model given as a channel $P(Y|X)$ would ideally transform the input X into the correct target Y. The model is noisy, however, making this channel a noisy channel. The noisy channel model refers to the models that can reduce the noise in the channel. Using the premise that low probability predictions (i.e. with larger uncertainty) are more likely to contain errors resulting from noise than are high probability predictions, we aim to minimize the noise of this channel and thus call our model the noisy channel model. We utilize this premise and allow the model for modeling the likelihood of making particular errors itself, instead of only relying it as loss.

Specifically, we propose a novel hierarchical network structure consisting of a traditional SRL model that provides the noisy prediction and a noise-estimating component that estimates the amount of noisy errors caused by this prediction. In order to make the noise controllable and removable, we introduce an external noise generator to produce and model noise for the input, giving us a source on which to base noise estimations. Based on the bottom noise estimator, we build a denoising SRL model in which a two-stream self-attention mechanism is adopted to incorporate the noisy prediction and the model noise-independent word representations of the bottom model. In our model, noise is explicitly added, estimated, and eventually eliminated. Our model differs from the traditional noise channel model as we do not seek to simply restore the original input and therefore provides a new alternative model for NLP labeling tasks. Furthermore, the noise within the model is random, so performing direct modeling is extremely difficult; however, we account for this by adding artificially synthetic noise for better denoising, a critical step for ensuring performance improvement.

Our empirical evaluation is conducted on the popular multilingual dependency SRL benchmark CoNLL-2009 for multiple settings. The results show that our proposed model can effectively alleviate the noise in the baseline model and consistently improve SRL performance. Notably, our model achieves the new state-of-the-art on several datasets. Additional ablation studies demonstrate that our proposed noisy channel model can effectively remove the inherent noise in the model; and thus obtain a higher quality output.

2 The Method

2.1 Overview

We present our noisy channel model for SRL in this section. First, our full model is split into bottom and top components. The lower component is a variant of a regular SRL model. We choose a simple and intuitive BiLSTM+MLP sequence labeling model as our basic model in the bottom component. We use this for noisy label prediction and noise estimation. The top component is designed to denoise the base model's noisy predicted probabilities and result in a more accurate prediction. Specifically, for this top component, we adopt a two-stream self-attention denoiser. On the one hand, it encode the word-level representation with a word-based self-attention; and on the other hand it denoises the label representation using a word-label cross-attention based on a probability-soft embedding of the noisy label prediction. It then combines the two streams to make the final prediction. The overall architecture of the noisy channel model is shown in Figure 1.

2.2 Base Model

First, we explain the base SRL model of our bottom component. Formally, given an input sentence $\mathbf{X} = \{x_1, x_2, ..., x_n\}$, the SRL model predicts a semantic triple of the predicate and argument and the relationship between them; i.e. $\mathbf{Y} = \{(p, a, r)\}, p \in \mathbf{X}, a \in \mathbf{X}, r \in \mathcal{R}$, where $\mathcal{R}$ is the vocabulary of semantic relationships. Although the target prediction is based on triples, in the sequence-based modeling method, the triple is transformed into several label sequences using a task decomposition, and then sequence labeling is performed separately. SRL is typically decomposed into four subtasks: predicate identification, predicate disambiguation, argument recognition, and argument classification. If the predicate is prespecified, the problem is changed to then only entails identifying and classifying its arguments.

Following He et al. (2018b)'s practice, we adopted a similar model structure and made some necessary changes to meet our overall needs. To vectorize sentence input X, we employed word embeddings e^{word} and a character CNN encoding network e^{char}, which not only takes into account word information but also better handles the out-of-vocabulary (OOV) problem. Other features like Parts-of-Speech (POS, e^{pos}) and lemmas (e^{lem}) are also integrated into the embeddings. Since the labeling of arguments is related to the predicate, predicate awareness is crucial to the implementation of the sequence labeling. Therefore, we use additional predicate indicator embedding e^{ind} is used to indicate which predicate is currently being processed. A word is then represented by concatenating its embeddings:

$$e_i^w = [e_i^{word}; e_i^{char}; e_i^{pos}; e_i^{lem}; e_i^{ind}],$$

where $[\cdot; \cdot]$ denotes a concatenation operation. Recently, pre-trained language models like ELMo (Peters et al., 2018), BERT have further improved the performance of many NLP tasks, our method can also further enhance its embeddings by concatenating language model features e^{plm}.

SRL is a context-related task, while the vector representation e_i of word w_i is context-independent. To further contextualize the representation, we encode the word representation $h_i \in \mathbf{H}$

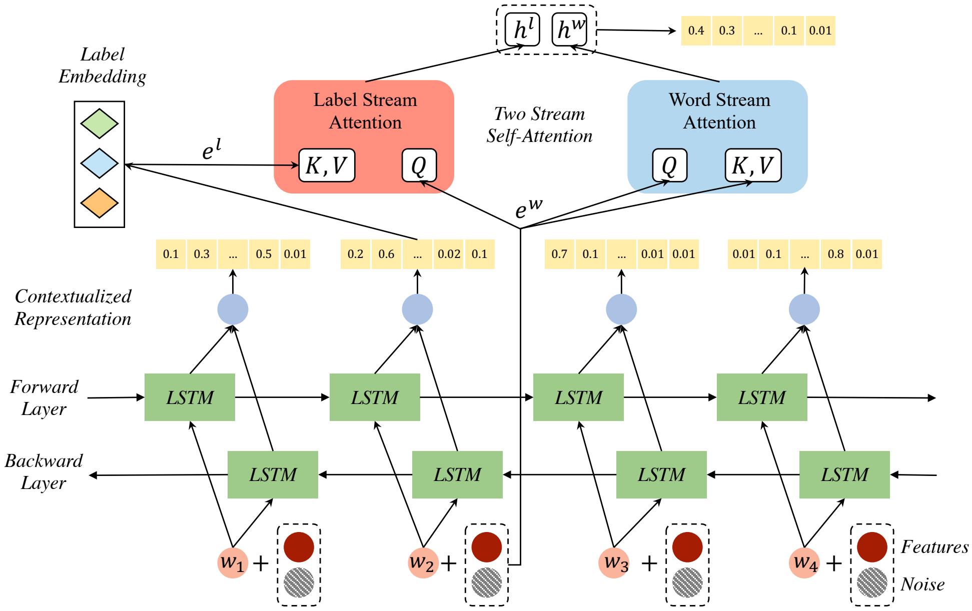


Figure 1: The overall architecture of our noisy channel model for semantic role labeling.

using a Bidirectional Long Short-Term Memory (LSTM) encoder (Hochreiter and Schmidhuber, 1997):

$$\mathbf{H} = \texttt{BiLSTM}(e_1^w, ..., e_n^w).$$

The BiLSTM encoder was chosen to facilitate a more fair comparison with LSTM-based SRL works. Encoders such as CNN or Transformer can obviously also be adopted for contextualizing representations.

We can employ Multi-layer Perceptron (MLP) layers to project the contextualized representation into the predicted probability distribution of each position:

$$P(y_i|\mathbf{X}, \theta) = \texttt{Softmax}(\text{MLP}(h_i)),$$

where θ is the parameters of base model.

2.3 Noise Estimation

Since the inherent noise of the model will have a negative impact on the model's prediction, further denoising is beneficial to performance improvement. We define the inherent noise of the base model as ζ. Since there is no direct way to model the real inherent noise as it may be unstructured and changing, we artificially synthesize a number of different noises and apply them to the same example so that the model can learn to capture and remove this noise.

Following (Gui et al., 2020), we use sampling based on Monte Carlo Dropout (Gal and Ghahramani, 2016a) to create rational noise, which Gal and Ghahramani refer to as uncertainty. We sample the dropout distribution M times for a single example. Assuming that the noise generated by sampling M times is $N = \{\eta_1, \eta_2, ..., \eta_M\}$, then the predicted probability of the instance with the k-th sampling noise can be written as:

$$P(y_i|\mathbf{X}, \theta, \zeta + \eta_k) = \texttt{Softmax}(\text{MLP}(h_i(\eta_k))),$$

where $h_i(\eta_k)$ represents the contextual representation of w_i with noise η_k.

According to the idea of boosting (Schapire, 2003; Wang et al., 2008), we combine these predicted probabilities with various synthetic noises,

$$P(y_i|\mathbf{X}, \theta) = \frac{1}{M}\sum_{k=1}^{M} P(y_i|\mathbf{X}, \theta, \zeta + \eta_k).$$

In terms of implementation, we repeat the input batch M times to allow parallelization on the GPU; and then use the standard dropout on the sentence length dimension. Notably, we also enable this dropout for synthesizing noise in the inference phase.

The synthesized noise is thus integrated into the predicted distribution. Since these probability scores are computed by a probaiblistically-weighted average of various noises, synthesized noise that better resembles the true noise will be emphasized in this averaging operation, and the parts of true noise that are inconsistent with the synthesized noise will be reduced by average operation. Reducing the artificial noise (as seen in the next step), should then lead to a reduction in real noise as well because of the isomorphism.

Based on predicted probabilities, we obtain the noisy label predictions $\hat{y}_i$ and calculate entropy as their noise estimation:

$$\hat{y}_i = \texttt{Argmax}(P(y_i|\mathbf{X}, \theta)),$$

$$\tau_i = -\sum_{r\in\mathcal{R}} P(y_i = r|\mathbf{X}, \theta)\log P(y_i = r|\mathbf{X}, \theta).$$

Entropy τ_i is a good noise estimation since when τ_i is larger, the predicted label $\hat{y}_i$ has a greater probability of being wrong, which means that the label in position i needs to be further processed by the denoiser.

2.4 Denoiser

The denoiser eliminates noise from the base model's label prediction. We leverage a two-stream self-attention structure to be able to focus on both the original word sequence and the noisy label sequence. Two-stream self-attention was first proposed in (Yang et al., 2019) to use two sets of hidden representations and model the two-stream interactions. In our work, the original word embedding (which lacks the inherent noise of the model) interacts with the soft embedding of the noisy label using the two-stream attention mechanism, which helps to remove the noise in the noisy label predition from the base model.

For the two-stream self-attention structure, we implement a multi-head self-attention (Vaswani et al., 2017) with relative position encoding following (Yang et al., 2019) as the basis. The calculation of two-stream attention between word sequence and noise label sequence can then be expressed as follows:

$$\begin{aligned} o^{w2w} &= \texttt{LayerNorm}\big(e^w + \\ &\qquad \texttt{RelMHAttn}(e^w W_Q, e^w W_K, e^w W_V)\big), \\ h^w &= \texttt{FeedForward}(o^{w2w}), \\ o^{w2l} &= \texttt{LayerNorm}\big(e^w + \\ &\qquad \texttt{RelMHAttn}(e^w W_Q, e^l W_K, e^l W_V)\big), \\ h^l &= \texttt{FeedForward}(o^{w2l}), \end{aligned}$$

where RelMHAttn denotes relative multi-head attention, LayerNorm denotes layer normalization, and FeedForward denotes a feed forward layer. There are two reasons for using relative multi-head attention rather than ordinary multi-head attention. On the one hand, no additional position encoding features are introduced. New features like those could corrupt the original features and have a negative impact on the denoising effect. On the other hand, the relative distance between labels is a valuable feature that can be beneficial for denoising.

To make the gradient for the noisy label embedding differentiable in the training phase, we did not use the embedding of predicted label from argmax operation directly; but rather adopted a soft embedding technique, which can be expressed as:

$$e_i^l = \sum_{r\in\mathcal{R}} P(y_i = r|\mathbf{X}, \theta)\texttt{Emb}^{(label)}(r),$$

in which $\texttt{Emb}^{(label)}$ represents the embedding space for semantic role labels. The basic idea is to weight sum all label embeddings using the predicted probabilities of each label as weights.

After two-stream encoding and denosing, we concatenate the output features in the two streams and use the MLP layer to project the features to the label probability space:

$$P(y_i|\mathbf{X}, \theta, \phi) = \texttt{Softmax}(\mathbf{MLP}([h_i^l; h_i^w])),$$

where ϕ denotes the parameters of the denoiser.

2.5 Training and Inference

Since our model makes two label predictions during the training process, the total training loss naturally consists of two parts:

$$\begin{aligned} \mathcal{L}(\theta) &= -\sum_{i=1}^{n} P(y_i = y_i^*|\mathbf{X}, \theta)\log P(y_i = y_i^*|\mathbf{X}, \theta), \\ \mathcal{L}(\theta, \phi) &= -\sum_{i=1}^{n} P(y_i = y_i^*|\mathbf{X}, \theta, \phi)\log P(y_i = y_i^*|\mathbf{X}, \theta, \phi), \\ \mathcal{L} &= \mathcal{L}(\theta) + \mathcal{L}(\theta, \phi). \end{aligned}$$

To optimize the model, we use cross-entropy to calculate the loss. The loss of the base model is denoted by $\mathcal{L}(\theta)$ and is used to make the base model predict the correct label as much as possible. $\mathcal{L}(\theta, \phi)$ is the loss of denoising during training and not only trains the denoiser; but also optimizes the whole model jointly.

In the inference stage, we do not directly take the final prediction of the denoiser as the output of the model. According to Ockham's razor, "*entities should not be multiplied without necessity*;" we therefore only use the output of the denoiser for some labels that are affected by noise and keep the rest. In terms of implementation, we set a threshold ρ for noise estimation τ and combine the two predictions thus:

$$\hat{y}_i^f = \begin{cases} \texttt{Argmax}(P(y_i|\mathbf{X}, \theta)), & \tau_i < \rho, \\ \texttt{Argmax}(P(y_i|\mathbf{X}, \theta, \phi)), & \tau_i \geq \rho. \end{cases}$$

3 Experiments and Analysis

3.1 Setup

We conducted experiments on the CoNLL-2009 shared task's multilingual dataset, which includes Catalan, Chinese, Czech, English, German, Japanese, and Spanish. In the experiments, we used two settings: *predicate-given* and *end-to-end*. In the *predicate-given* setting, we use the official dataset's pre-specified predicate but predict the predicate sense, argument, and semantic roles. In the *end-to-end* setting, all of the predicate and argument must be predicted since they are all unknown. Additionally, in keeping with (He et al., 2018b), we use POS and lemma features in the model. These are the predicted POS tags and lemma as given by the CoNLL-2009 shared task for each language. To keep the model concise, we did not leverage syntactic tree information, which makes our model syntax-agnostic.

Our model uses pre-trained fastText (Grave et al., 2018) embeddings as a word embedding initialization. Other POS embeddings, lemma embeddings, and label embeddings are initialized randomly. In the case of using a pre-trained language model, the ELMo-original-5.5B model is used for ELMo feature extraction, while for BERT (Devlin et al., 2019), BERT-large-cased is used for English and BERT-base-chinese for Chinese. To keep the results comparable to (Li et al., 2020a), BERT for other languages in multilingual benchmarks adopted is the same as them. All models are trained for up to 400 epochs, early stopping patient is set to 20, and the batch size is 64. We use the categorical cross-entropy as the objective and the Adam optimizer (Kingma and Ba, 2015). For other model hyper-parameters, please see Appendix A.1.

3.2 Analysis

Predicate-given Results In the CoNLL-2009 multilingual benchmark, the English and Chinese datasets are used in the majority of SRL works. To compare with these works, we list the results from recent works and our models in Table 1. When comparing the baseline to our proposed full model, in the case when not using any pre-trained language models, our full model obtained 1.0+ Sem-F1 improvement on both the English in-domain (ID) and out-of-domain (OOD) tests, as well as the Chinese test set, demonstrating the effectiveness of our proposed method.

As previously mentioned, when we compared the SRL performance of previous works, we found that the baseline results of different modeling methods differed. There is generally a trend in the performance of the models: Graph > Tree > Sequence. This trend may be caused by the more complex modeling methods (i.e, graph-based methods) taking into account more features. Conversely, however, decoding speed follows its own trend: Sequence > Tree > Graph in terms of decoding speed (for details see our speed analysis in Appendix A.3). Without using pre-trained language models or additional features such as syntax parse trees, our method first achieves state-of-the-art among sequence-based modeling approaches. Furthermore, our sequence-based model outperformed the best results of the tree-based modeling approach (Cai et al., 2018) and achieved results comparable to those of the state-of-the-art graph-based modeling work (Fei et al., 2021). This shows that our proposed method is both fast and effective.

Whereas previous works that typically integrated more features had disparities in their improvements on ID and OOD results, our approach interestingly boasts similar improvements both the ID and OOD settings. That our approach provides an even improvement across both settings suggests that it does reduce the noise inherent in the model, as this is a problem that affects both settings. This contrasts the development of new features, which are typically biased towards one of the two settings and thus give disparate performance improvements.

The results in the *predicate-given* setting on CoNLL-09 multilingual are shown in Table 2. The study of multilingual SRL has resurfaced in recent years, especially after the introduction of the multilingual pre-trained language model - BERT. The improvements of our full models over the baselines on the multilingual test sets are consistent. Although the performance of our baseline (sequence-based) lags behind that of tree-based (He et al., 2019) and graph-based models (Li et al., 2020), our full model achieves comparable results to these models in most cases and further obtains state-of-the-art results in Catalan, Czech, and Japanese languages with the help of BERT. Furthermore, our

System	Modeling	English ID			English OOD			Chinese		
		P	R	F_1	P	R	F_1	P	R	F_1
Roth and Lapata (2016)	S	90.0	85.5	87.7	78.6	73.8	76.1	83.2	75.9	79.4
Marcheggiani et al. (2017a)	S	88.7	86.8	87.7	79.4	76.2	**77.7**	83.4	79.1	81.2
Marcheggiani and Titov (2017a)	S	89.1	86.8	88.0	78.5	75.9	77.2	84.6	80.4	**82.5**
He et al. (2018b) (w/ ELMo)	S	89.7	89.3	89.5	81.9	76.9	79.3	84.2	81.5	82.8
Cai et al. (2018)	T	89.9	89.2	89.6	79.8	78.3	79.0	84.7	84.0	84.3
Li et al. (2018) (w/ ELMo)	S	90.3	89.3	89.8	80.6	79.0	79.8	84.8	81.2	83.0
Li et al. (2019) (w/ ELMo)	G	89.6	91.2	90.4	81.7	81.4	81.5	–	–	–
He et al. (2019) (w/ ELMo)	T	90.00	90.65	90.32	–	–	–	84.44	84.95	84.70
He et al. (2019) (w/ BERT)	T	90.41	91.32	90.86	–	–	–	86.15	86.70	86.42
Lyu et al. (2019a) (w/ ELMo)	S	–	–	90.99	–	–	82.18	–	–	83.31
Chen et al. (2019) (w/ ELMo)	S	–	–	91.06	–	–	82.72	–	–	81.65
Cai and Lapata (2019) (w/ ELMo)	T	91.7	90.8	91.2	83.2	81.9	82.5	85.4	84.6	85.0
Kasai et al. (2019)	S	89.0	88.2	**88.6**	78.0	77.2	77.6	–	–	–
Kasai et al. (2019) (w/ ELMo)	S	90.3	90.0	90.2	81.0	80.5	80.8	–	–	–
Zhou et al. (2020a)	G	88.73	89.83	89.28	82.46	83.20	82.82	–	–	–
Zhou et al. (2020a) (w/ BERT)	G	91.21	91.19	91.20	85.65	86.09	85.87	–	–	–
Li et al. (2020a)	G	91.60	88.95	90.26	82.6	78.75	80.63	88.35	83.82	86.02
Li et al. (2020a) (w/ BERT)	G	92.59	90.98	91.77	86.49	83.80	85.13	89.07	87.71	88.38
Fei et al. (2021)	G	90.8	90.0	90.4	80.7	79.3	80.2	–	–	–
Fei et al. (2021) (w/ BERT)	G	92.5	92.5	92.5	85.6	85.3	85.4	–	–	–
Baseline	S	89.06	88.54	88.80	78.57	77.10	77.83	84.87	80.46	82.61
Full Model	S	90.66	89.01	**89.83**	80.35	77.46	**78.88**	86.28	81.83	**84.00**
Full Model (w/ ELMo)	S	91.47	89.82	90.63	83.95	81.61	82.76	–	–	–
Full Model (w/ BERT)	S	92.11	91.95	92.03	86.45	83.96	85.19	88.91	86.85	87.87

Table 1: Precision, recall and semantic F_1-score on CoNLL-2009 English in-domain (ID), out-of-domain (OOD) data and Chinese test sets with *predicate-given* setting. S, T, and G in the modeling column refers to sequence-based, tree-based, and graph-based modeling approaches.

approach is neither limited to the sequence modeling model nor the SRL task. The noise channel model is a task-independent method of alleviating a model's inherent noise, and our approach model- and task-independent. Thus, our method also can be transplanted to a tree or graph-based baseline. We leave this to future work since in this paper, we are primarily interested in enhancing performance by alleviating the model's inherent noise in this paper.

End-to-end Results The end-to-end setting necessitates fewer external preset conditions and therefore better resembles realistic applications. Table 3 shows the results of our approach in this more challenging setting. The outcome of predicate identification has a significant impact on the overall F_1, but this has usually been overlooked in previous work. Inconsistent choices in predicate identifier often render findings significantly incomparable, so in this paper, we advocate for reporting the predicate's F1 score as well so that we can ensure the overall Sem-F_1 increase is due to better role labeling rather than better predicate recognition. The results show not only is our model's improvement is stable in this challenge setting, but when compared to the previous results, our results are comparable or even superior, despite using a simpler modeling method and simpler neural structures.

Where did Denoising Work? For the base model, the longer the sequence is, the more likely it is affected by noise, so we hypothesize that denoising should bring greater effect to longer sequences. To verify our hypothesis, we compared the performance of the baseline and the full model on different length sentences, as shown in Figure 2(a).

According to the figure, the baseline model's performance is obviously better when handling short sentences, which is common in sequence-based modeling. This suggests that long sentences may be more influenced by noise. The figure also shows that our full model improves on this baseline and its improvement grows when handling longer sentences, which is consistent with our hypothesis.

Denoising, Refining, or Smoothing? Generally speaking, if we do not use synthesized noise, our approach can be thought of as refining; additionally, if we do not use dropout but instead sum noise features from distribution sampling, it can be thought of as model smoothing. We thus performed two experiments to contrast our method with these similar

Model	Catalan	Chinese	Czech	English	German	Japanese	Spanish
Hajič et al. (2009)	80.3	78.6	85.4	85.6	79.7	78.2	80.5
Zhao et al. (2009)	80.3	77.7	85.2	86.2	76.0	78.2	80.5
Roth and Lapata (2016)	–	79.4	–	87.7	80.1	–	80.2
Marcheggiani et al. (2017a)	–	81.2	86.0	87.7	–	–	80.3
Mulcaire et al. (2018)	79.45	81.89	85.14	87.24	69.97	76.00	77.32
Kasai et al. (2019) (w/ ELMo)	–	–	–	90.2	–	–	83.0
Lyu et al. (2019a) (w/ ELMo)	80.91	83.31	87.62	90.99	75.87	82.54	80.53
Cai and Lapata (2019)	–	85.00	–	91.20	83.80	–	82.90
He et al. (2019)	84.35	84.55	88.76	89.96	78.54	83.12	83.70
He et al. (2019) (w/ BERT)	85.14	86.42	89.66	90.86	80.87	83.76	84.60
Li et al. (2020a)†	85.37	86.02	90.60	90.26	76.41	83.25	84.39
Li et al. (2020a) (w/ BERT)	86.90	**88.69**	91.93	91.77	**85.54**	85.90	**86.96**
Baseline	81.42	83.20	88.60	88.80	77.26	81.34	80.52
Full Model	82.58	84.56	89.74	89.83	79.30	82.53	81.47
Full Model (w/ BERT)	**87.05**	87.96	**92.24**	92.03	83.55	**86.04**	85.65

Table 2: Semantic F_1-score on the CoNLL-2009 in-domain multilingual test sets with the *predicate-given* setting. † The predicate disambiguators in (Li et al., 2020a) in w/ BERT and w/ o BERT setting use the same sequence labeling model (w/ BERT) which improves the overall Sem-F_1, while two separate disambiguators are used in our work, so the w/o BERT results are not entirely comparable.

Model	Catalan		Chinese		Czech		English		German		Japanese		Spanish	
	PF_1	F_1	PF_1	F_1	PF_1	F_1	PF_1	F_1	PF_1	F_1	PF_1	F_1	PF_1	F_1
Li et al. (2020a)†	–	84.07	–	82.01	–	89.45	–	86.16	–	60.48	–	74.20	–	83.11
Li et al. (2020a) (w/ BERT)	–	85.82	–	85.68	–	91.22	–	88.70	–	67.15	–	78.88	–	86.00
Baseline	90.82	81.45	90.47	80.05	96.92	88.18	91.62	85.06	65.28	55.61	84.64	72.45	85.83	80.11
Full Model	90.82	82.91	90.47	81.22	96.92	89.50	91.62	85.98	65.28	57.02	84.64	73.86	85.83	81.27
Full Model (w/ BERT)	95.05	**85.96**	93.11	85.22	97.67	**91.45**	92.59	88.63	70.30	65.48	86.74	78.70	92.41	84.83

Table 3: Semantic F_1-score on the CoNLL-2009 multilingual test sets with the *end-to-end* setting. PF_1 denotes the F_1 score (includes predicate sense). † The predicate recognizer in (Li et al., 2020a) in w/ BERT and w/ o BERT setting use the same sequence labeling model (w/ BERT) which improves the overall Sem-F_1, while two separate recognizer are used in our work, so the w/o BERT results are not entirely comparable.

approaches: 1) exclude noise sampling (Refining) and 2) sample Gaussian noise distribution as features (Smoothing).

Table 4 presents this comparison. Refining (*w/o Noise Sampling*) and Smoothing (*w/ Gaussian Noise*) both obtained better results than the baseline, indicating that these two approaches are effective methods of improving SRL performance; however, in comparison to our full model, these approaches' lead to lesser improvements, which demonstrates the superiority of our method.

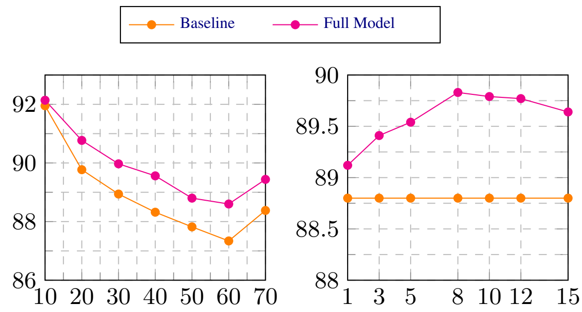
Method	ID			OOD		
	P	R	F_1	P	R	F_1
Baseline	89.06	88.54	88.80	78.57	77.10	77.83
Boosting Only	88.97	88.85	88.90	78.52	77.24	77.87
Full Model	90.66	89.01	89.83	80.35	77.46	78.88
w/o Noise Sampling	89.34	88.92	89.12	78.92	76.95	77.92
w/ Gaussian Noise	90.53	88.72	89.61	80.44	76.87	78.61
w/o Relative Pos	90.27	88.92	89.58	80.16	77.30	78.70
w/o Two-stream Self-attn	89.51	88.79	89.14	79.98	76.55	78.22
w/o Soft Label Emb	90.16	88.66	89.40	80.08	77.13	78.57

Table 4: Ablation study on CoNLL-09 English test set.

3.3 Ablation Study

In our full model, we augmented the base model with several novel techniques. To illustrate the importance of introducing these novel techniques, we performed an ablation study as shown in the bottom of Table 4. Removing the relative position encoding, two-stream attention mechanism (using two separate Transformers instead), and soft label embedding (using the embedding obtained on the argmax prediction) led to performance reductions of varying degrees. Among them, the removal of the two-stream attention mechanism affected model performance the most, which shows that the interaction between the word representations (without model noise) and the label representations (with model noise) is critical for denoising.

Furthermore, we explore the sampling size M in order to determine the optimal sampling size for improving performance. Figure 2(b) depicts



(a) Sem-F_1 score vs. sentence length

(b) Sem-F_1 score vs. sampling size M

Figure 2: Sem-F_1 score vs. sentence length and Sem-F_1 score vs. sampling size M. The CoNLL-09 in-domain test set is used for both comparisons.

the SRL performance curve with various sampling sizes M. The optimal sampling size, as shown in the figure, is 8. When it is less than 8, the performance improves as M becomes larger. After M reaches 8, the performance is essentially stable. which demonstrates that 8 is sufficient for denoising, and no additional gain will be available by increasing M further. Please refer to Appendix A.2 for other ablation studies.

4 Related Work

Semantic Role Labeling SRL has been a heated research realm since the introduction of neural networks. Early neural network-based methods (Wang et al., 2015) simply modeled semantic role labeling as a word classification task and employed recurrent networks for annotation. Leveraging syntax is a common way of boosting performance for SRL. While using syntactic treebanks, Graph Convolutional Networks (GCN) can be applied for SRL for syntax-aware labeling (Marcheggiani and Titov, 2017b), though syntax-agnostic models were also argued efficient by (Marcheggiani et al., 2017b). Still, there remains a strong connection between the studies of syntax and SRL (He et al., 2018c; Marcheggiani and Titov, 2020; Shi et al., 2020).

Refinement on output from SRL models has been increasingly popular for research. Iterative refinement on SRL has been shown to outperform base models (Lyu et al., 2019b). Higher order scorers have also been used as a source of more accurate arc scores in semantic graph (Li et al., 2020b). Our model is similar to those refining models in process, but rather than just refining, our model also specifically focuses on removing noise.

Noise Processing Noise, in NLP tasks, represented in the model as uncertainty when processing complex information or structures. Denoising can be leveraged to produce better results, as high certainty implicates faults for refinement. The noise channel model, which applies Bayesian approximation constraints to eliminate noise in generated outputs, has become a popular method in NLG tasks, including NMT (Wang et al., 2019; Zhou et al., 2020b) and summary generation (Xu et al., 2020). In domain of linguistic parsing, noise refers to labels predicted with high uncertainty. (Zhang et al., 2019a) applied an adaptive uncertainty-aware decoder for semantic parsing. Dependency parsing can also benefit from adaptive strategy based on uncertainty detection, as demonstrated in (van der Goot and van Noord, 2018). Uncertainty mechanism has also been applied in suspense prediction (Wilmot and Keller, 2020), spoken language assessment (Malinin et al., 2017), and document class prevalence inference (Keith and O'Connor, 2018).

Specifically speaking, mainstream noise processing can be categorized into two topics: evaluation and elimination. Noise elimination generally refers to he works introduced above that discuss about result refinement, while noise evaluation refers to modeling the uncertainty of a model. In this topic, Gal and Ghahramani (2016b) have suggested that the softmax function may not be a solid indication of model uncertainty, which suggests that this topic needs more research. Also on this topic, (He et al., 2020) rectified confidence scores using their MSD model to better evaluate result uncertainty, and Ethayarajh (2020) measured the bias in classification models using Bernstein-bounded unfairness.

5 Conclusion

In this paper, we propose a noisy channel model for the SRL model's inherent noise problem. In our proposed model, synthesized noise is combined and then averaged to best emphasize similar authentic noise while weaken the different inherent noise, and this noise is then reduced by a denoiser based on two-stream attention to obtain the final output. We demonstrated the effectiveness of our approach by evaluating our models on the CoNLL-09 multilingual benchmark, and we also investigated the differences between our method and refining and smoothing techniques. Apart from being successful in SRL, our approach is also notable because it broadly applicable to other NLP tasks.

References

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. Association for Computational Linguistics.

Rui Cai and Mirella Lapata. 2019. Semi-supervised semantic role labeling with cross-view training. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1018–1027, Hong Kong, China. Association for Computational Linguistics.

Xinchi Chen, Chunchuan Lyu, and Ivan Titov. 2019. Capturing argument interaction in semantic role labeling with capsule networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5415–5425, Hong Kong, China. Association for Computational Linguistics.

Janara Christensen, Mausam, Stephen Soderland, and Oren Etzioni. 2010. Semantic role labeling for open information extraction. In *Proceedings of the NAACL HLT 2010 First International Workshop on Formalisms and Methodology for Learning by Reading*, pages 52–60, Los Angeles, California. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Kawin Ethayarajh. 2020. Is your classifier actually biased? measuring fairness under uncertainty with bernstein bounds. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2914–2919, Online. Association for Computational Linguistics.

Hao Fei, Fei Li, Bobo Li, and Donghong Ji. 2021. Encoder-decoder based unified semantic role labeling with label-aware syntax.

Yarin Gal and Zoubin Ghahramani. 2016a. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1050–1059. JMLR.org.

Yarin Gal and Zoubin Ghahramani. 2016b. A theoretically grounded application of dropout in recurrent neural networks. In *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 1019–1027.

Edouard Grave, Piotr Bojanowski, Prakhar Gupta, Armand Joulin, and Tomas Mikolov. 2018. Learning word vectors for 157 languages. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Tao Gui, Jiacheng Ye, Qi Zhang, Zhengyan Li, Zichu Fei, Yeyun Gong, and Xuanjing Huang. 2020. Uncertainty-aware label refinement for sequence labeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2316–2326, Online. Association for Computational Linguistics.

Jan Hajič, Massimiliano Ciaramita, Richard Johansson, Daisuke Kawahara, Maria Antònia Martí, Lluís Màrquez, Adam Meyers, Joakim Nivre, Sebastian Padó, Jan Štěpánek, Pavel Straňák, Mihai Surdeanu, Nianwen Xue, and Yi Zhang. 2009. The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pages 1–18, Boulder, Colorado. Association for Computational Linguistics.

Jianfeng He, Xuchao Zhang, Shuo Lei, Zhiqian Chen, Fanglan Chen, Abdulaziz Alhamadani, Bei Xiao, and ChangTien Lu. 2020. Towards more accurate uncertainty estimation in text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8362–8372, Online. Association for Computational Linguistics.

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. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Shexia He, Zuchao Li, and Hai Zhao. 2019. Syntax-aware multilingual semantic role labeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5350–5359, Hong Kong, China. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Shexia He, Zuchao Li, Hai Zhao, and Hongxiao Bai. 2018c. Syntax for semantic role labeling, to be, or not to be. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 1: Long Papers*, pages 2061–2071. Association for Computational Linguistics.

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

Jungo Kasai, Dan Friedman, Robert Frank, Dragomir Radev, and Owen Rambow. 2019. Syntax-aware neural semantic role labeling with supertags. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 701–709, Minneapolis, Minnesota. Association for Computational Linguistics.

Katherine Keith and Brendan O'Connor. 2018. Uncertainty-aware generative models for inferring document class prevalence. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4575–4585, Brussels, Belgium. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

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. Association for Computational Linguistics.

Zuchao Li, Shexia He, Hai Zhao, Yiqing Zhang, Zhuosheng Zhang, Xi Zhou, and Xiang Zhou. 2019. Dependency or span, end-to-end uniform semantic role labeling. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 6730–6737. AAAI Press.

Zuchao Li, Hai Zhao, Rui Wang, and Kevin Parnow. 2020a. High-order semantic role labeling. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1134–1151, Online. Association for Computational Linguistics.

Zuchao Li, Hai Zhao, Rui Wang, and Kevin Parnow. 2020b. High-order semantic role labeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings, EMNLP 2020, Online Event, 16-20 November 2020*, pages 1134–1151. Association for Computational Linguistics.

Ding Liu and Daniel Gildea. 2010. Semantic role features for machine translation. In *Proceedings of the 23rd International Conference on Computational Linguistics (Coling 2010)*, pages 716–724, Beijing, China. Coling 2010 Organizing Committee.

Chunchuan Lyu, Shay B. Cohen, and Ivan Titov. 2019a. Semantic role labeling with iterative structure refinement. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1071–1082, Hong Kong, China. Association for Computational Linguistics.

Chunchuan Lyu, Shay B. Cohen, and Ivan Titov. 2019b. Semantic role labeling with iterative structure refinement. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 1071–1082. Association for Computational Linguistics.

Andrey Malinin, Anton Ragni, Kate Knill, and Mark Gales. 2017. Incorporating uncertainty into deep learning for spoken language assessment. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 45–50, Vancouver, Canada. Association for Computational Linguistics.

Diego Marcheggiani, Anton Frolov, and Ivan Titov. 2017a. 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. Association for Computational Linguistics.

Diego Marcheggiani, Anton Frolov, and Ivan Titov. 2017b. 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), Vancouver, Canada, August 3-4, 2017*, pages 411–420. Association for Computational Linguistics.

Diego Marcheggiani and Ivan Titov. 2017a. 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. Association for Computational Linguistics.

Diego Marcheggiani and Ivan Titov. 2017b. Encoding sentences with graph convolutional networks for semantic role labeling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 1506–1515. Association for Computational Linguistics.

Diego Marcheggiani and Ivan Titov. 2020. Graph convolutions over constituent trees for syntax-aware semantic role labeling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 3915–3928. Association for Computational Linguistics.

Phoebe Mulcaire, Swabha Swayamdipta, and Noah A. Smith. 2018. Polyglot semantic role labeling. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 667–672, Melbourne, Australia. Association for Computational Linguistics.

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. Association for Computational Linguistics.

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. Association for Computational Linguistics.

Robert E Schapire. 2003. The boosting approach to machine learning: An overview. *Nonlinear estimation and classification*, pages 149–171.

Tianze Shi, Igor Malioutov, and Ozan Irsoy. 2020. Semantic role labeling as syntactic dependency parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7551–7571. Association for Computational Linguistics.

Rob van der Goot and Gertjan van Noord. 2018. Modeling input uncertainty in neural network dependency parsing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4984–4991, Brussels, Belgium. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Liwei Wang, Masashi Sugiyama, Cheng Yang, Zhi-Hua Zhou, and Jufu Feng. 2008. On the margin explanation of boosting algorithms. In *21st Annual Conference on Learning Theory - COLT 2008, Helsinki, Finland, July 9-12, 2008*, pages 479–490. Omnipress.

Shuo Wang, Yang Liu, Chao Wang, Huanbo Luan, and Maosong Sun. 2019. Improving back-translation with uncertainty-based confidence estimation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 791–802, Hong Kong, China. Association for Computational Linguistics.

Zhen Wang, Tingsong Jiang, Baobao Chang, and Zhifang Sui. 2015. Chinese semantic role labeling with bidirectional recurrent neural networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1626–1631, Lisbon, Portugal. Association for Computational Linguistics.

David Wilmot and Frank Keller. 2020. Modelling suspense in short stories as uncertainty reduction over neural representation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1763–1788, Online. Association for Computational Linguistics.

Jiacheng Xu, Shrey Desai, and Greg Durrett. 2020. Understanding neural abstractive summarization models via uncertainty. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6275–6281, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

Xiang Zhang, Shizhu He, Kang Liu, and Jun Zhao. 2019a. Adansp: Uncertainty-driven adaptive decoding in neural semantic parsing. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4265–4270. Association for Computational Linguistics.

Zhuosheng Zhang, Yuwei Wu, Zuchao Li, and Hai Zhao. 2019b. Explicit contextual semantics for text comprehension. In *Proceedings of the 33rd Pacific Asia Conference on Language, Information and Computation*, pages 298–308. Waseda Institute for the Study of Language and Information.

Hai Zhao, Wenliang Chen, Jun'ichi Kazama, Kiyotaka Uchimoto, and Kentaro Torisawa. 2009. Multilingual dependency learning: Exploiting rich features for tagging syntactic and semantic dependencies. In *Proceedings of the Thirteenth Conference on Computational Natural Language Learning (CoNLL 2009): Shared Task*, pages 61–66, Boulder, Colorado. Association for Computational Linguistics.

Jie Zhou and Wei Xu. 2015. End-to-end learning of semantic role labeling using recurrent neural networks. 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 1127–1137, Beijing, China. Association for Computational Linguistics.

Junru Zhou, Zuchao Li, and Hai Zhao. 2020a. Parsing all: Syntax and semantics, dependencies and spans. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4438–4449, Online. Association for Computational Linguistics.

Yikai Zhou, Baosong Yang, Derek F. Wong, Yu Wan, and Lidia S. Chao. 2020b. Uncertainty-aware curriculum learning for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6934–6944, Online. Association for Computational Linguistics.

A Appendix

A.1 Hyper-parameters

Embedding	**Dimension**
Word Embed	100
Char	100
POS	64
Lemma	100
Label	400
Predicate Indicator	32
ELMo†	300
BERT†	300
Encoder	**Size**
BiLSTMs	256×2
BiLSTMs Layers	3
BiLSTMs Out MLP	128
Two Stream Attention	**Size**
Transformer Hidden	512
Transformer FFN	1024
Transformer Heads	8
Transformer Layers	2
Dropout	**Probability**
BiLSTM Input	0.33
BiLSTM Output	0.33
BiLSTMs	$[0.33, 0.33]$
Transformers	0.05
Noise	0.15
Optimizer	
Learning Rate	0.001
Adam μ	0.9
Adam ν	0.9999
Batch Size	64
Decay Rate	0.999995
Warmup Steps	100

Table 5: Model hyper-parameters. † denotes optional.

A.2 Noisy Channel Model on BERT Baseline

The main experimental results show that BERT can significantly boost the performance of SRL. To demonstrate that our approach will continue to work on the strong BERT baseline, we present the w/ BERT baseline results in Table 6. The results show that, while BERT is a great help to the baseline performance, the use of our Noisy Channel Model can further play a useful role.

A.3 Inference Speed Analysis

To analyze the inference speed of different modeling approaches, we measured the total inference time on the CoNLL-2009 English in-domain test set with the scale of model parameters similar (i.e., same hidden size, model layers). 5 runs are performed and then reported the average speed for better stability. The comparison results are shown in Table 7. From the comparison, the inference speed order is Sequence > Tree > Graph, and our full model only slightly decreases the speed due to a good parallel design compared to the baseline.

Method	ID			OOD		
	P	R	F_1	P	R	F_1
Baseline	89.06	88.54	88.80	78.57	77.10	77.83
Baseline (w/ BERT)	91.97	91.23	91.59	85.50	83.87	84.67
Full Model	90.66	89.01	89.83	80.35	77.46	78.88
Full Model (w/ BERT)	92.11	91.95	92.03	86.45	83.96	85.19

Table 6: Performance comparison between baseline and noisy channel model with BERT enhancement.

Method	Modeling	Speed (sent./s)
(Cai et al., 2018)	T	199.5
(Li et al., 2020a)	G	165.9
Baseline	S	245.1
Full Model	S	240.6

Table 7: Inference speed for different modeling approaches.