Uncertainty-Aware Bootstrap Learning for Joint Extraction on Distantly-Supervised Data

Yufei Li[1], **Xiao Yu**[2], **Yanchi Liu**[3], **Haifeng Chen**[3], **Cong Liu**[1]
[1]University of California, Riverside [2]Stellar Cyber [3]NEC Labs America
[1]{yli927,congl}@ucr.edu, [2]xyu@stellarcyber.ai,
[3]{yanchi,haifeng}@nec-labs.com

Abstract

Jointly extracting entity pairs and their relations is challenging when working on distantly-supervised data with ambiguous or noisy labels. To mitigate such impact, we propose *uncertainty-aware bootstrap learning*, which is motivated by the intuition that the higher uncertainty of an instance, the more likely the model confidence is inconsistent with the ground truths. Specifically, we first explore instance-level data uncertainty to create an initial high-confident examples. Such subset serves as filtering noisy instances and facilitating the model to converge fast at the early stage. During bootstrap learning, we propose self-ensembling as a regularizer to alleviate inter-model uncertainty produced by noisy labels. We further define probability variance of joint tagging probabilities to estimate inner-model parametric uncertainty, which is used to select and build up new reliable training instances for the next iteration. Experimental results on two large datasets reveal that our approach outperforms existing strong baselines and related methods.

1 Introduction

Joint extraction involves extracting multiple types of entities and relations between them using a single model, which is necessary in automatic knowledge base construction (Yu et al., 2020). One way to cheaply acquire a large amount of labeled data for training joint extraction models is through distant supervision (DS) (Mintz et al., 2009). DS involves aligning a knowledge base (KB) with an unlabeled corpus using hand-crafted rules or logic constraints. Due to the lack of human annotators, DS brings a large proportion of noisy labels, e.g., over 30% noisy instances in some cases (Mintz et al., 2009), making it impossible to learn useful features. The noise can be either false relations due to the aforementioned rule-based matching assumption or wrong entity tags due to limited coverage over entities in open-domain KBs.

Existing distantly-supervised approaches model noise relying either on heuristics such as reinforcement learning (RL) (Nooralahzadeh et al., 2019; Hu et al., 2021) and adversarial learning (Chen et al., 2021), or pattern-based methods (Jia et al., 2019; Shang et al., 2022) to select trustable instances. Nevertheless, these methods require designing heuristics or hand-crafted patterns which may encourage a model to leverage spurious features without considering the confidence or uncertainty of its predictions.

In response to these problems, we propose **UnBED**—**Un**certainty-aware **B**ootstrap learning for joint **E**xtraction on **D**istantly-supervised data. UnBED assumes that 1) low data uncertainty indicates reliable instances using a pre-trained language model (PLM) in the initial stage, 2) model should be aware of trustable entity and relation labels regarding its uncertainty after training. Our bootstrap serves uncertainty as a principle to mitigate the impact of noise labels on model learning and validate input sequences to control the number of training examples in each step. Particularly, we quantify data uncertainty of an instance according to its *winning score* (Hendrycks and Gimpel, 2017) and *entropy* (Shannon, 1948). We define averaged maximum probability that is estimated by a joint PLM over each token in a sequence to adapt previous techniques in joint extraction scheme. Instances with low data uncertainty are collected to form an initial subset, which is used to tune the joint PLM tagger and facilitate fast convergence. Then, we define parametric uncertainty in two perspectives—inter-model and inner-model uncertainty. The former is quantified by self-ensembling (Wang and Wang, 2022) and serves as a regularizer to improve model robustness against noisy labels during training. The latter is captured by probability variance in MC Dropout (Gal and Ghahramani, 2016) for selecting new confident instances for the next training iteration. Such two-

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 2: Short Papers, pages 1349–1358
July 9-14, 2023 ©2023 Association for Computational Linguistics

fold model uncertainties reinforce with each other to guide the model to iteratively improve its robustness and learn from reliable knowledge.

2 Related Work

Joint Extraction Methods Joint extraction detects entities and their relations using a single model, which effectively integrates the information from both sources and therefore achieves better results in both subtasks compared to pipelined methods (Zheng et al., 2017). For example, unified methods tag entities and relation simultaneously, e.g., (Zheng et al., 2017) proposes a novel tagging scheme which converts joint extraction to a sequence labeling problem; (Dai et al., 2019) introduces query position and sequential tagging to extract overlapping relations. Such methods avoid producing redundant information compared to parameter-sharing neural models (Miwa and Bansal, 2016; Gupta et al., 2016), and require no hand-crafted features that are used in structured systems (Yu et al., 2020).

To address the challenge of learning from DS, pre-trained transformers (e.g., BERT, GPT-2) have gain much attention. They model strong expressive context-aware representations for text sequence through multiple attention layers, and achieve state-of-the-art performance on various NLP tasks (Radford et al., 2019; Devlin et al., 2019; Li et al., 2022). They can be cheaply fine-tuned to solve different downstream tasks including NER and RC. Specifically, BERT is trained on large English corpus using masked language modeling. The multi-head attention weights indicate interactions between each pair of words and its hidden states integrate semantic information of the whole sentence, which are used to decode different tagging results.

Uncertainty Methods Uncertainty generally comes from two sources—aleatoric uncertainty and epistemic uncertainty. The former is also referred to as data uncertainty, describing noise inherent in the data generation. Methods mitigating such uncertainty include data interpolation (Dong et al., 2018), winning score, and temperature scale (Guo et al., 2017). The latter is also called model uncertainty, describing whether the structure choice and model parameters best describe the data distribution. One main solution to mitigate model uncertainty is Bayesian Neural Network (BNN) (Klein et al., 2017) that puts a prior distribution on its weights. To save computational cost, Monte Carlo

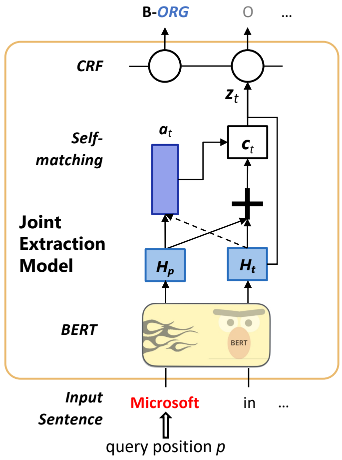

Figure 1: Joint extraction as a token classification task.

dropout is proposed as an approximation of variational Bayesian inference (Gal and Ghahramani, 2016), realized by training models with dropout layers and testing with stochastic inference to quantify probability variance. Besides BNN, self-ensembling (Wang and Wang, 2022) which measures the outputs variance between models with the same architecture has been shown effective to reduce parametric uncertainty across models.

3 Joint Extraction Model

Tagging Scheme For an input sequence $\mathcal{X} = \{x_1, ..., x_n\}$, we tag n sequences according to different query position p following (Dai et al., 2019). If p is the start of an entity (query entity e_1), the sequence is an instance. The entity type is labeled at p and other entities e_2 which have relationship with the query entity are labeled with relation types re. The rest of tokens are labeled "O" (Outside), meaning they do not correspond to the query entity. Accordingly, we convert joint extraction into a token classification task and extract relation triplets $\{e_1, re, e_2\}$ in each instance, as shown in Figure 1.

Position-Attentive Encoder we use BERT (Devlin et al., 2019) to encode a sentence $\mathcal{X}$ into token-level representations $\boldsymbol{h} = \{\boldsymbol{h}_1, .., \boldsymbol{h}_n\}$, where $\boldsymbol{h}_i \in \mathbb{R}^d$ is a d-dimensional vector corresponding to the i-th token in $\mathcal{X}$. For each query p, self-matching is applied to calculate the position-attention $\boldsymbol{a}_t \in \mathbb{R}^T$ between token at p and each token at target position t, which compares the sentence representations against itself to collect context information (Tan et al., 2018). The produced position-aware representation $\boldsymbol{c}_t \in \mathbb{R}^{T \times d}$ is an attention-weighted sentence vector $\boldsymbol{c}_t = \boldsymbol{a}_t^\top \boldsymbol{h}$. Finally, we concatenate $\boldsymbol{h}_t$ and $\boldsymbol{c}_t$ to generate position-aware and context-aware representations $\boldsymbol{u}_t = [\boldsymbol{h}_t | \boldsymbol{c}_t]$.

CRF Decoder (Lafferty et al., 2001) For each position-aware representation $\boldsymbol{u}_t$, we first learn a linear transformation $\boldsymbol{z}_t = \boldsymbol{W}\boldsymbol{u}_t \in \mathbb{R}^C$ to represent tag scores for the t-th token. Here C is the number of distinct tags. For an instance with labels $\boldsymbol{y} = \{y_1, ..., y_n\}$, the decoding score $s(\boldsymbol{z}, \boldsymbol{y})$ is the sum of transition score from tag y_t to tag y_{t+1} plus the input score $z_t^{y_t}$. The conditional probability $p(\boldsymbol{y}|\boldsymbol{z})$ is the softmax over $s(\boldsymbol{z}, \boldsymbol{y})$ for all possible label sequences $\boldsymbol{y}'$. We maximize the log-likelihood of correct tag sequences during training $\mathcal{L}_c = \sum \log p(\boldsymbol{y}|\boldsymbol{z})$.

4 Uncertainty-Aware Bootstrap Learning

Motivation One of the main challenges in bootstrap learning is to evaluate the "correctness" of a labeled instance. We consider this problem from an uncertainty perspective and assume instances with lower uncertainty are more likely to be correctly labeled. In this section, we first propose instance-level data uncertainty which is used to filter noisy examples and build an initial subset. Then, we introduce our two-fold model uncertainties which helps iteratively mitigate DS effect and build up trustable examples during bootstrap learning.

4.1 Data Uncertainty

Presenting examples in an easy-to-hard order at different training stages can benefit models (Platanios et al., 2019; Zhou et al., 2020), we propose data uncertainty as a way to quantify the "hardness" of an instance. To better estimate the data uncertainty, we use pre-trained language models (PLMs) to generate tag probability for each token in a sequence. Our intuition is that higher uncertain inputs are "harder" to be generated by a PLM, as it already has rationales of language. Accordingly, we propose two data uncertainties, which can be used individually or combined together:

Winning Score (WS) The maximum softmax probability reflects data uncertainty of an input (Hendrycks and Gimpel, 2017). Given an input instance $\mathcal{I} = \{x_1, ..., x_n\}$, we define data uncertainty $u^d(\mathcal{I})$ as the minus averaged token classification winning score:

$$u^d(\mathcal{I}) = -\frac{1}{n}\sum_{t=1}^{n} \max_{c\in[1,C]} P(y_t = c|x_t) \quad (1)$$

Entropy Shannon entropy (Shannon, 1948) is widely used to reflect information uncertainty. We propose data uncertainty $u^d(\mathcal{I})$ as the averaged token classification entropy:

$$u^d(\mathcal{I}) = \frac{1}{n}\sum_{t=1}^{n}\sum_{c=1}^{C} P(y_t = c|x_t) \log P(y_t = c|x_t) \quad (2)$$

We filter out examples with high uncertainty scores and build an initial subset with "simple" examples. At the early training stage, a model is not aware of what a decent distribution $P(y|x)$ should be, thus data uncertainty facilitates it to converge fast by tuning on a fairly "simple" subset.

4.2 Model Uncertainty

In our bootstrap learning, we define model uncertainty, i.e., epistemic uncertainty (Kendall and Gal, 2017), to measure whether model parameters can best describe the data distribution following (Zhou et al., 2020). A small model uncertainty indicates the model is confident that the current training data has been well learned (Wang et al., 2019). We adopt Monte Carlo Dropout (Gal and Ghahramani, 2016) to approximate Bayesian inference which captures inner-model parametric uncertainty. Specifically, we perform K forward passes through our joint model. In each pass, part of network neurons θ are randomly deactivated. Finally, we yield K samples on model parameters $\{\hat{\theta}_1, ..., \hat{\theta}_K\}$. We use the averaged token classification **Probability Variance (PV)** (Shelmanov et al., 2021) over all tags for instance $\mathcal{I}$:

$$u^m(\theta) = \frac{1}{n}\sum_{t=1}^{n}\sum_{c=1}^{C} \mathrm{Var}\left[P(y_t = c|x_t, \hat{\theta}_k)\right]_{k=1}^{K} \quad (3)$$

where Var[.] is the variance of distribution over the K passes following the common settings in (Dong et al., 2018; Xiao and Wang, 2019). Accordingly, model is aware of its confidence over each instance and how likely the label is noisy.

4.3 Training Strategy

Uncertainty-Aware Loss Besides MC Dropout which measures parametric uncertainty within a model, we also consider mitigating parametric uncertainty between models to stabilize the weights during training. Specifically, we use self-ensembling (He et al., 2020; Wang and Wang, 2022) to calculate the loss between the same models to improve model robustness and reduce the label noise effect on model performance.

Algorithm 1 Bootstrap Learning

Input: Original dataset $\mathcal{D} = \{(\mathcal{I}^n, y^n)\}_{n=1}^N$, two joint models f_1, f_2 with parameters θ_1, θ_2;

Compute data uncertainty $u^d(\mathcal{I})$ for each instance $\mathcal{I}$ in $\mathcal{D}$;

Initial dataset $\mathcal{C} \leftarrow$ Select data pairs $(\mathcal{I}^n, y^n)$ such that $u^d(\mathcal{I}) < \tau^d$ from $\mathcal{D}$;

for *epoch* $e = 1, ...$ **do**

    Train f_1, f_2 on $\mathcal{C}$ using Eq. (5);

    Calculate model uncertainty $u^m(\theta_1)$ on $\mathcal{D}$;

    $\mathcal{C} \leftarrow$ Select data pairs $(\mathcal{I}^n, y^n)$ such that $u^m(\mathcal{I}; \theta_1) < \tau^m$ from $\mathcal{D}$;

We create another joint model with identical framework, e.g., architecture, loss functions, hyperparameters, and compute a self-ensemble loss $\mathcal{L}_e$ to minimize the difference between two outputs from the two models regarding the same inputs:

$$\mathcal{L}_e = \sum KL(f(\mathcal{I}; \theta_1), f(\mathcal{I}; \theta_2)) \tag{4}$$

where $KL(.)$ is the Kullback-Leibler divergence between two probabilistic distributions, θ_1, θ_2 denote the parameters of first and second models. We formulate our final uncertainty-aware objective $\mathcal{L}$ as the sum of CRF and self-ensemble loss:

$$\mathcal{L} = \mathcal{L}_c + \alpha \mathcal{L}_e \tag{5}$$

where α denotes the weight of self-ensembling, and $\mathcal{L}_c$ means the token classification loss.

Bootstrap Learning Procedure To mitigate the DS effect on model performance, we propose a twofold bootstrap learning strategy (see Algorithm 1). Specifically, we first apply data uncertainty to filter "harder" examples and redistribute a reliable initial training data $\mathcal{M}$. Then, we iteratively feed examples following an easy-to-hard order to the model. In each training iteration, we regularize the joint model with self-ensembling loss to reduce the impact of noisy labels on model parameters. Then we use probability variance to select new confident training instances $\mathcal{D}'$ that can be explained by the model as the next training inputs. The more certain examples are selected, the more likely the model will learn beneficial information and will converge faster. We repeat the above procedure until the F1 score on the validation set converges.

5 Experiments

5.1 Setup

We evaluate the performance of UnBED on two datasets, NYT and Wiki-KBP. The NYT (Riedel et al., 2010) dataset collects news from New York Times and its training data is automatically labeled by DS. We use the revised test dataset (Jia et al., 2019) that is manually annotated to ensure quality. The Wiki-KBP (Ling and Weld, 2012) dataset collects articles from Wikipedia. Its training data is labeled by DS (Liu et al., 2017), and the test set is manually annotated (Ellis et al., 2013).

We compare UnBED with the following baselines: **ARNOR** (Jia et al., 2019), a pattern-based method to reduce noise for distantly-supervised triplet extraction. **PURE** (Zhong and Chen, 2021), a pipeline approach that uses pre-trained BERT entity model to first recognize entities and then employs a relation model to detect underlying relations. **FAN** (Hao et al., 2021), an adversarial method including a transformers encoder to reduce noise for distantly-supervised triplet extraction.

Evaluation We evaluate the extracted triplets for each sentence based on Precision (Prec.), Recall (Rec.), and F1. A triplet $\{e_1, re, e_2\}$ is marked correct if the relation type re, two entities e_1, e_2 are all correct. We build a validation set by randomly sampling 10% sentences from the test set.

Implementation Details We use Hugging Face *bert-large-uncased* (Devlin et al., 2019) pre-trained model as backbone. For ARNOR, the hidden vector size is set to 300. In regularization training, we find optimal parameters α as 1 for both datasets. We implement UnBED and all baselines in PyTorch, with Adam optimizer, initial learning rate 10^{-5}, dropout rate 0.1, and batch size 8. For initial subset configuration, we choose data uncertainty threshold 0.5. For bootstrap learning, an empirical model uncertainty threshold is set to 0.6 with the best validation F1.

5.2 Overall Results

As shown in Table 1, UnBED significantly outperforms all baselines in precision and F1 metric. Specifically, UnBED achieves 8% F1 improvement on NYT (3% on Wiki-KBP) over denoising approaches—ARNOR and FAN. Our approach also outperforms baselines using pretrained transformers (PURE and FAN), showing that uncertainty-aware bootstrap learning effectively reduces the impact of noisy labels.

Method	NYT			Wiki-KBP		
	Prec.	Rec.	F1	Prec.	Rec.	F1
ARNOR (Jia et al., 2019)	0.588	0.614	0.600	0.402	0.471	0.434
PURE (Zhong and Chen, 2021)	0.536	0.664	0.593	0.395	0.433	0.413
FAN (Hao et al., 2021)	0.579	0.646	0.611	0.391	0.467	0.426
UnBED-WS	**0.662**	0.730	**0.694**	**0.429**	0.501	**0.462**
UnBED-Entropy	0.651	**0.741**	0.693	0.422	**0.509**	0.461

Table 1: Evaluation results on NYT and Wiki-KBP datasets. **Bold** numbers denote the best metrics. UnBED-WS and UnBED-Entropy denote UnBED with winning score and entropy as the data uncertainty, respectively.

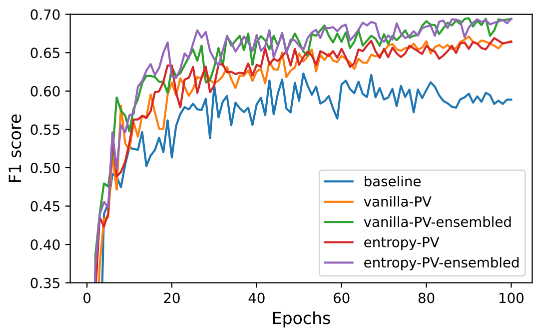

Figure 2: F1 score vs. Epochs under different settings. Vanilla-PV-ensembled denotes UnBED-WS, and entropy-PV-ensembled denotes UnBED-Entropy.

5.3 Further Analysis

We analyze the functionality of different components in Figure 2. We observe that both the entropy-PV and vanilla-PV outperform the baseline (joint model directly trained on the original DS dataset) in terms of F1 (5∼7% increase), demonstrating the effect of filtering noisy labels and selecting trustable instance using probability variance. Besides, self-ensembling further enhances the performance in later training stage (2∼4 F1 increase), proving that mitigating the inter-model uncertainty benefits model robustness against noisy labels.

6 Conclusions

In this paper, we propose a novel uncertainty-aware bootstrap learning framework for distantly-supervised joint extraction. Specifically, we define data uncertainty in generally token classification to filter out highly-error-prone instances and build an initial high-confident subset, which is used to tune the joint extraction model for fast convergence. We then propose a two-fold bootstrap learning procedure which iteratively mitigates the DS impact on model robustness and selects new trustable training instances. Experimental results on two benchmark datasets show that UnBED significantly outperforms other denoising techniques.

Limitations

In this work we propose an uncertainty-aware bootstrap learning framework for joint extraction. Though it achieves state-of-the-art performance compared to other denoising techniques, UnBED requires large training resources considering the ensemble loss calculated between two large PLMs and the probability variance calculated on the PLM joint extraction model. In our future work, we hope to incorporate pruning techniques during training to improve the efficiency. We will also consider more complex relations between entities, e.g., relations beyond the sentence boundary, to fit in real-world information extraction scenarios.

Acknowledgements

This work was supported by NSF CNS 2135625, CPS 2038727, CNS Career 1750263, and a Darpa Shell grant.

References

Tao Chen, Haochen Shi, Liyuan Liu, Siliang Tang, Jian Shao, Zhigang Chen, and Yueting Zhuang. 2021. Empower distantly supervised relation extraction with collaborative adversarial training. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12675–12682. AAAI Press.

Dai Dai, Xinyan Xiao, Yajuan Lyu, Shan Dou, Qiaoqiao She, and Haifeng Wang. 2019. Joint extraction of entities and overlapping relations using position-attentive sequence 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 6300–6308. AAAI Press.

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.

Li Dong, Chris Quirk, and Mirella Lapata. 2018. Confidence modeling for neural semantic parsing. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 743–753, Melbourne, Australia. Association for Computational Linguistics.

Joe Ellis, Jeremy Getman, Justin Mott, Xuansong Li, Kira Griffitt, Stephanie M. Strassel, and Jonathan Wright. 2013. Linguistic resources for 2013 knowledge base population evaluations. In *Proceedings of the Sixth Text Analysis Conference, TAC 2013, Gaithersburg, Maryland, USA, November 18-19, 2013*. NIST.

Yarin Gal and Zoubin Ghahramani. 2016. 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.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR.

Pankaj Gupta, Hinrich Schütze, and Bernt Andrassy. 2016. Table filling multi-task recurrent neural network for joint entity and relation extraction. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2537–2547, Osaka, Japan. The COLING 2016 Organizing Committee.

Kailong Hao, Botao Yu, and Wei Hu. 2021. Knowing false negatives: An adversarial training method for distantly supervised relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9661–9672, Online and Punta Cana, Dominican Republic. 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.

Dan Hendrycks and Kevin Gimpel. 2017. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Xuming Hu, Chenwei Zhang, Yawen Yang, Xiaohe Li, Li Lin, Lijie Wen, and Philip S. Yu. 2021. Gradient imitation reinforcement learning for low resource relation extraction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 2737–2746. Association for Computational Linguistics.

Wei Jia, Dai Dai, Xinyan Xiao, and Hua Wu. 2019. ARNOR: Attention regularization based noise reduction for distant supervision relation classification. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1399–1408, Florence, Italy. Association for Computational Linguistics.

Alex Kendall and Yarin Gal. 2017. What uncertainties do we need in bayesian deep learning for computer vision? 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 5574–5584.

Aaron Klein, Stefan Falkner, Jost Tobias Springenberg, and Frank Hutter. 2017. Learning curve prediction with bayesian neural networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

John D. Lafferty, Andrew McCallum, and Fernando C. N. Pereira. 2001. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, page 282–289, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Shuyang Li, Yufei Li, Jianmo Ni, and Julian McAuley. 2022. SHARE: a system for hierarchical assistive recipe editing. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11077–11090, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xiao Ling and Daniel S. Weld. 2012. Fine-grained entity recognition. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence, July 22-26, 2012, Toronto, Ontario, Canada*. AAAI Press.

Liyuan Liu, Xiang Ren, Qi Zhu, Shi Zhi, Huan Gui, Heng Ji, and Jiawei Han. 2017. Heterogeneous supervision for relation extraction: A representation learning approach. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 46–56, Copenhagen, Denmark. Association for Computational Linguistics.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. 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*, pages 1003–1011, Suntec, Singapore. Association for Computational Linguistics.

Makoto Miwa and Mohit Bansal. 2016. End-to-end relation extraction using lstms on sequences and tree structures. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Farhad Nooralahzadeh, Jan Tore Lønning, and Lilja Øvrelid. 2019. Reinforcement-based denoising of distantly supervised NER with partial annotation. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP, DeepLo@EMNLP-IJCNLP 2019, Hong Kong, China, November 3, 2019*, pages 225–233. Association for Computational Linguistics.

Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabás Póczos, and Tom M. Mitchell. 2019. Competence-based curriculum learning for neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1162–1172. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Sebastian Riedel, Limin Yao, and Andrew McCallum. 2010. Modeling relations and their mentions without labeled text. In *Machine Learning and Knowledge Discovery in Databases, European Conference, ECML PKDD 2010, Barcelona, Spain, September 20-24, 2010, Proceedings, Part III*, volume 6323 of *Lecture Notes in Computer Science*, pages 148–163. Springer.

Yuming Shang, Heyan Huang, Xin Sun, Wei Wei, and Xian-Ling Mao. 2022. A pattern-aware self-attention network for distant supervised relation extraction. *Inf. Sci.*, 584:269–279.

Claude E. Shannon. 1948. A mathematical theory of communication. *Bell Syst. Tech. J.*, 27(3):379–423.

Artem Shelmanov, Evgenii Tsymbalov, Dmitri Puzyrev, Kirill Fedyanin, Alexander Panchenko, and Maxim Panov. 2021. How certain is your Transformer? In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1833–1840, Online. Association for Computational Linguistics.

Zhixing Tan, Mingxuan Wang, Jun Xie, Yidong Chen, and Xiaodong Shi. 2018. Deep semantic role labeling with self-attention. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press.

Hongjun Wang and Yisen Wang. 2022. Self-ensemble adversarial training for improved robustness. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

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 2019, Hong Kong, China, November 3-7, 2019*, pages 791–802. Association for Computational Linguistics.

Yijun Xiao and William Yang Wang. 2019. Quantifying uncertainties in natural language processing tasks. 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 7322–7329. AAAI Press.

Bowen Yu, Zhenyu Zhang, Xiaobo Shu, Tingwen Liu, Yubin Wang, Bin Wang, and Sujian Li. 2020. Joint extraction of entities and relations based on a novel decomposition strategy. In *ECAI 2020 - 24th European Conference on Artificial Intelligence, 29 August-8 September 2020, Santiago de Compostela, Spain, August 29 - September 8, 2020 - Including 10th Conference on Prestigious Applications of Artificial Intelligence (PAIS 2020)*, volume 325 of *Frontiers in Artificial Intelligence and Applications*, pages 2282–2289. IOS Press.

Suncong Zheng, Feng Wang, Hongyun Bao, Yuexing Hao, Peng Zhou, and Bo Xu. 2017. Joint extraction of entities and relations based on a novel tagging scheme. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1227–1236, Vancouver, Canada. Association for Computational Linguistics.

Zexuan Zhong and Danqi Chen. 2021. A frustratingly easy approach for entity and relation extraction. In

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 50–61, Online. Association for Computational Linguistics.

Yikai Zhou, Baosong Yang, Derek F. Wong, Yu Wan, and Lidia S. Chao. 2020. 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.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 7

☒ A2. Did you discuss any potential risks of your work?
We study open-domain information extraction for researches in this area

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Abstract and Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☒ Did you use or create scientific artifacts?

Left blank.

☐ B1. Did you cite the creators of artifacts you used?
No response.

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
No response.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
No response.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
No response.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
No response.

☐ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
No response.

C ☑ Did you run computational experiments?

Section 5

☐ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
No response.

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 5

☐ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
No response.

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 5

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.