Fine-tuning Pre-Trained Transformer Language Models to Distantly Supervised Relation Extraction

Christoph Alt   **Marc Hübner**   **Leonhard Hennig**

German Research Center for Artificial Intelligence (DFKI)
Speech and Language Technology Lab
{christoph.alt, marc.huebner, leonhard.hennig}@dfki.de

Abstract

Distantly supervised relation extraction is widely used to extract relational facts from text, but suffers from noisy labels. Current relation extraction methods try to alleviate the noise by multi-instance learning and by providing supporting linguistic and contextual information to more efficiently guide the relation classification. While achieving state-of-the-art results, we observed these models to be biased towards recognizing a limited set of relations with high precision, while ignoring those in the long tail. To address this gap, we utilize a pre-trained language model, the OpenAI Generative Pre-trained Transformer (GPT) (Radford et al., 2018). The GPT and similar models have been shown to capture semantic and syntactic features, and also a notable amount of "common-sense" knowledge, which we hypothesize are important features for recognizing a more diverse set of relations. By extending the GPT to the distantly supervised setting, and fine-tuning it on the NYT10 dataset, we show that it predicts a larger set of distinct relation types with high confidence. Manual and automated evaluation of our model shows that it achieves a state-of-the-art AUC score of 0.422 on the NYT10 dataset, and performs especially well at higher recall levels.

1 Introduction

Relation extraction (RE), defined as the task of identifying the relationship between concepts mentioned in text, is a key component of many natural language processing applications, such as knowledge base population (Ji and Grishman, 2011) and question answering (Yu et al., 2017). Distant supervision (Mintz et al., 2009; Hoffmann et al., 2011) is a popular approach to heuristically generate labeled data for training RE systems by aligning entity tuples in text with known relation instances from a knowledge base, but suffers from
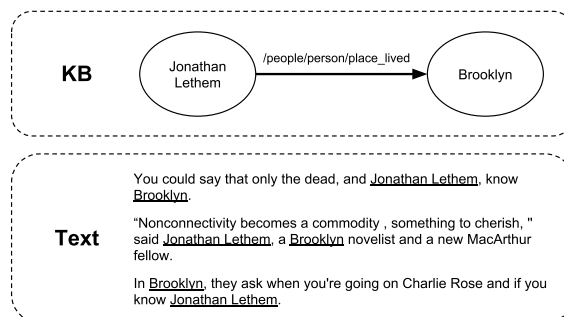

Figure 1: Distant supervision generates noisily labeled relation mentions by aligning entity tuples in a text corpus with relation instances from a knowledge base.

noisy labels and incomplete knowledge base information (Min et al., 2013; Fan et al., 2014). Figure 1 shows an example of three sentences labeled with an existing KB relation, two of which are false positives and do not actually express the relation.

Current state-of-the-art RE methods try to address these challenges by applying multi-instance learning methods (Mintz et al., 2009; Surdeanu et al., 2012; Lin et al., 2016) and guiding the model by *explicitly* provided semantic and syntactic knowledge, e.g. part-of-speech tags (Zeng et al., 2014) and dependency parse information (Surdeanu et al., 2012; Zhang et al., 2018b). Recent methods also utilize side information, e.g. paraphrases, relation aliases, and entity types (Vashishth et al., 2018). However, we observe that these models are often biased towards recognizing a limited set of relations with high precision, while ignoring those in the long tail (see Section 5.2).

Deep language representations, e.g. those learned by the Transformer (Vaswani et al., 2017) via language modeling (Radford et al., 2018), have been shown to *implicitly* capture useful semantic and syntactic properties of text solely by

unsupervised pre-training (Peters et al., 2018), as demonstrated by state-of-the-art performance on a wide range of natural language processing tasks (Vaswani et al., 2017; Peters et al., 2018; Radford et al., 2018; Devlin et al., 2018), including supervised relation extraction (Alt et al., 2019). Radford et al. (2019) even found language models to perform fairly well on answering open-domain questions without being trained on the actual task, suggesting they capture a limited amount of "common-sense" knowledge. We hypothesize that pre-trained language models provide a stronger signal for distant supervision, better guiding relation extraction based on the knowledge acquired during unsupervised pre-training. Replacing explicit linguistic and side-information with implicit features improves domain and language independence and could increase the diversity of the recognized relations.

In this paper, we introduce a Distantly Supervised Transformer for Relation Extraction (DISTRE). We extend the standard Transformer architecture by a selective attention mechanism to handle multi-instance learning and prediction, which allows us to fine-tune the pre-trained Transformer language model directly on the distantly supervised RE task. This minimizes explicit feature extraction and reduces the risk of error accumulation. In addition, the self-attentive architecture allows the model to efficiently capture long-range dependencies and the language model to utilize knowledge about the relation between entities and concepts acquired during unsupervised pre-training. Our model achieves a state-of-the-art AUC score of 0.422 on the NYT10 dataset, and performs especially well at higher recall levels, when compared to competitive baseline models.

We selected the GPT as our language model because of its fine-tuning efficiency and reasonable hardware requirements, compared to e.g. LSTM-based language models (Ruder and Howard, 2018; Peters et al., 2018) or BERT (Devlin et al., 2018). The contributions of this paper can be summarized as follows:

- We extend the GPT to handle bag-level, multi-instance training and prediction for distantly supervised datasets, by aggregating sentence-level information with selective attention to produce bag-level predictions (§ 3).

- We evaluate our fine-tuned language model on the NYT10 dataset and show that it

achieves a state-of-the-art AUC compared to RESIDE (Vashishth et al., 2018) and PCNN+ATT (Lin et al., 2016) in held-out evaluation (§ 4, § 5.1).

- We follow up on these results with a manual evaluation of ranked predictions, demonstrating that our model predicts a more diverse set of relations and performs especially well at higher recall levels (§ 5.2).

- We make our code publicly available at https://github.com/DFKI-NLP/DISTRE.

2 Transformer Language Model

This section reviews the Transformer language model as introduced by Radford et al. (2018). We first define the Transformer-Decoder (Section 2.1), followed by an introduction on how contextualized representations are learned with a language modeling objective (Section 2.2).

2.1 Transformer-Decoder

The Transformer-Decoder (Liu et al., 2018a), shown in Figure 2, is a decoder-only variant of the original Transformer (Vaswani et al., 2017). Like the original Transformer, the model repeatedly encodes the given input representations over multiple layers (i.e., Transformer blocks), consisting of masked multi-head self-attention followed by a position-wise feedforward operation. In contrast to the original decoder blocks this version contains no form of unmasked self-attention since there are no encoder blocks. This is formalized as follows:

$$h_0 = TW_e + W_p$$
$$h_l = tf\_block(h_{l-1}) \; \forall \, l \, \in \, [1, L] \qquad (1)$$

Where T is a matrix of one-hot row vectors of the token indices in the sentence, W_e is the token embedding matrix, W_p is the positional embedding matrix, L is the number of Transformer blocks, and h_l is the state at layer l. Since the Transformer has no implicit notion of token positions, the first layer adds a learned positional embedding $e_p \in \mathbb{R}^d$ to each token embedding $e_t^p \in \mathbb{R}^d$ at position p in the input sequence. The self-attentive architecture allows an output state h_l^p of a block to be informed by all input states h_{l-1}, which is key to efficiently model long-range dependencies. For language modeling, however, self-attention must be constrained (masked) not to attend to positions
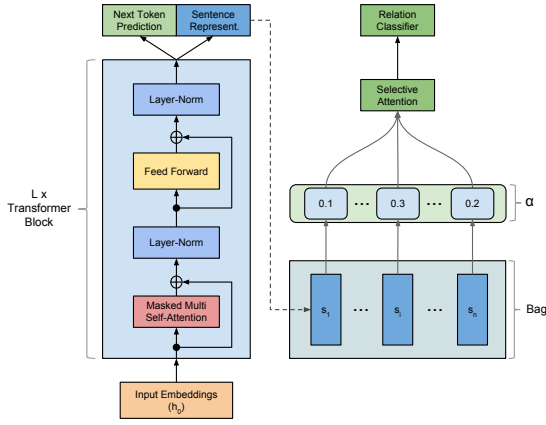
Figure 2: Transformer-Block architecture and training objectives. A Transformer-Block is applied at each of the L layers to produce states h_1 to h_L. After encoding each sentence in a bag into its representation s_i, selective attention informs the relation classifier with a representation aggregated over all sentences $[s_1, \ldots, s_n]$.

ahead of the current token. For a more exhaustive description of the architecture, we refer readers to Vaswani et al. (2017) and the excellent guide "The Annotated Transformer".[1]

2.2 Unsupervised Pre-training of Language Representations

Given a corpus $\mathcal{C} = \{c_1, \ldots, c_n\}$ of tokens c_i, the language modeling objective maximizes the likelihood

$$L_1(\mathcal{C}) = \sum_i \log P(c_i | c_{i-1}, \ldots, c_{i-k}; \theta), \quad (2)$$

where k is the context window considered for predicting the next token c_i via the conditional probability P. The distribution over the target tokens is modeled using the previously defined Transformer model as follows:

$$P(c) = softmax(h_L W_e^T), \quad (3)$$

where h_L is the sequence of states after the final layer L, W_e is the embedding matrix, and θ are the model parameters that are optimized by stochastic gradient descent. This results in a probability distribution for each token in the input sequence.

3 Multi-Instance Learning with the Transformer

This section introduces our extension to the original transformer architecture, enabling bag-level

---

[1] http://nlp.seas.harvard.edu/2018/04/03/attention.html

multi-instance learning on distantly supervised datasets (Section 3.1), followed by a description of our task-specific input representation for relation extraction (Section 3.2).

3.1 Distantly Supervised Fine-tuning on Relation Extraction

After pre-training with the objective in Eq. 2, the language model is fine-tuned on the relation extraction task. We assume a labeled dataset $\mathcal{D} = \{(x_i, head_i, tail_i, r_i)\}_{i=1}^{N}$, where each example consists of an input sequence of tokens $x_i = [x^1, \ldots, x^m]$, the positions $head_i$ and $tail_i$ of the relation's head and tail entity in the sequence of tokens, and the corresponding relation label r_i, assigned by distant supervision. Due to its noisy annotation, label r_i is an unreliable target for training. Instead, the relation classification is applied on a bag level, representing each entity pair $(head, tail)$ as a set $S = \{x_1, \ldots, x_n\}$ consisting of all sentences that contain the entity pair. A set representation s is then derived as a weighted sum over the individual sentence representations:

$$s = \sum_i \alpha_i s_i, \quad (4)$$

where α_i is the weight assigned to the corresponding sentence representation s_i. A sentence representation is obtained by feeding the token sequence x_i of a sentence to the pre-trained model and using the last state h_L^m of the final state representation h_L as its representation s_i. The set representation s is then used to inform the relation classifier.

We use selective attention (Lin et al., 2016), shown in Figure 2, as our approach for aggregating a bag-level representation s based on the individual sentence representations s_i. Compared to average selection, where each sentence representation contributes equally to the bag-level representation, selective attention learns to identify the sentences with features most clearly expressing a relation, while de-emphasizing those that contain noise. The weight α_i is obtained for each sentence by comparing its representation against a learned relation representation r:

$$\alpha_i = \frac{exp(s_i r)}{\sum_{j=1}^{n} exp(s_j r)} \quad (5)$$

To compute the output distribution $P(l)$ over relation labels, a linear layer followed by a softmax

is applied to s:

$$P(l|S,\theta) = softmax(W_r s + b), \qquad (6)$$

where W_r is the representation matrix of relations r and $b \in R^{d_r}$ is a bias vector. During fine-tuning we want to optimize the following objective:

$$L_2(\mathcal{D}) = \sum_{i=1}^{|\mathcal{S}|} \log P(l_i|S_i,\theta) \qquad (7)$$

According to Radford et al. (2018), introducing language modeling as an auxiliary objective during fine-tuning improves generalization and leads to faster convergence. Therefore, our final objective combines Eq. 2 and Eq. 7:

$$L(\mathcal{D}) = \lambda * L_1(\mathcal{D}) + L_2(\mathcal{D}), \qquad (8)$$

where the scalar value λ is the weight of the language model objective during fine-tuning.

3.2 Input Representation

Our input representation (see Figure 3) encodes each sentence as a sequence of tokens. To make use of sub-word information, we tokenize the input text using byte pair encoding (BPE) (Sennrich et al., 2016). The BPE algorithm creates a vocabulary of sub-word tokens, starting with single characters. Then, the algorithm iteratively merges the most frequently co-occurring tokens into a new token until a predefined vocabulary size is reached. For each token, we obtain its input representation by summing over the corresponding token embedding and positional embedding.

While the model is pre-trained on plain text sentences, relation extraction requires a structured input, namely a sentence and relation arguments. To avoid task-specific changes to the architecture, we adopt a traversal-style approach similar to Radford et al. (2018). The structured, task-specific input is converted to an ordered sequence to be directly fed to the model without architectural changes. Figure 3 provides a visual illustration of the input format. It starts with the tokens of the head and tail entity, separated by delimiters, followed by the token sequence of the sentence containing the entity pair, and ends with a special classification token. The classification token signals the model to generate a sentence representation for relation classification. Since our model processes the input left-to-right, we add the relation arguments to the beginning, to bias the attention mechanism towards their token representation while processing the sentence's token sequence.
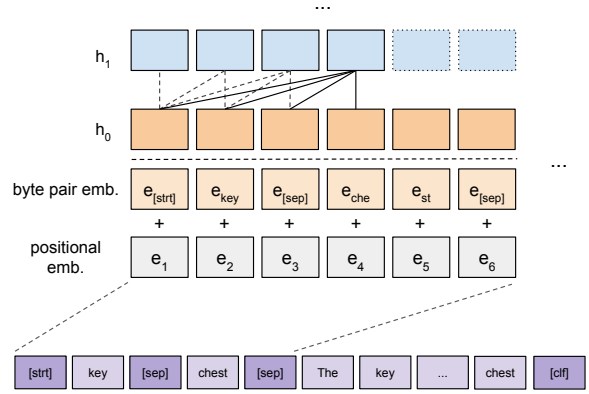

Figure 3: Relation extraction requires a structured input for fine-tuning, with special delimiters to assign different meanings to parts of the input. The input embedding h_0 is created by summing over the positional embedding and the byte pair embedding for each token. States h_l are obtained by self-attending over the states of the previous layer h_{l-1}.

4 Experiment Setup

In the following section we describe our experimental setup. We run our experiments on the distantly supervised NYT10 dataset and use PCNN+ATTN (Lin et al., 2016) and RE-SIDE (Vashishth et al., 2018) as the state-of-the-art baselines.

The piecewise convolutional neural network (PCNN) segments each input sentence into parts to the left, middle, and right of the entity pair, followed by convolutional encoding and selective attention to inform the relation classifier with a bag-level representation. RESIDE, on the other hand, uses a bidirectional gated recurrent unit (GRU) to encode the input sentence, followed by a graph convolutional neural network (GCN) to encode the explicitly provided dependency parse tree information. This is then combined with named entity type information to obtain a sentence representation that can be aggregated via selective attention and forwarded to the relation classifier.

4.1 NYT10 Dataset

The NYT10 dataset by Riedel et al. (2010) is a standard benchmark for distantly supervised relation extraction. It was generated by aligning Freebase relations with the New York Times corpus, with the years 2005–2006 reserved for training and 2007 for testing. We use the version of the dataset pre-processed by Lin et al. (2016), which

is openly accessible online.[2] The training data contains 522,611 sentences, 281,270 entity pairs and 18,252 relational facts. The test data contains 172,448 sentences, 96,678 entity pairs and 1,950 relational facts. There are 53 relation types, including NA if no relation holds for a given sentence and entity pair. Per convention we report *Precision@N* (precision scores for the top 100, top 200, and top 300 extracted relation instances) and a plot of the precision-recall curves. Since the test data is also generated via distant supervision, and can only provide an approximate measure of the performance, we also report *P@100*, *P@200*, and *P@300* based on a manual evaluation.

4.2 Pre-training

Since pre-training is computationally expensive, and our main goal is to show its effectiveness by fine-tuning on the distantly supervised relation extraction task, we reuse the language model[3] published by Radford et al. (2018) for our experiments. The model was trained on the BooksCorpus (Zhu et al., 2015), which contains around 7,000 unpublished books with a total of more than 800M words of different genres. The model consists of $L = 12$ decoder blocks with 12 attention heads and 768 dimensional states, and a feed-forward layer of 3072 dimensional states. We reuse the byte-pair encoding vocabulary of this model, but extend it with task-specific tokens (e.g., start, end, delimiter).

4.3 Hyperparameters

During our experiments we found the hyperparameters for fine-tuning, reported in Radford et al. (2018), to be very effective. We used the Adam optimization scheme (Kingma and Ba, 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.999$, a batch size of 8, a learning rate of 6.25e-5, and a linear learning rate decay schedule with warm-up over 0.2% of training updates. We trained the model for 3 epochs and applied residual and attention dropout with a rate of 0.1, and classifier dropout with a rate of 0.2.

5 Results

This section presents our experimental results. We compare DISTRE to other works on the NYT10 dataset, and show that it recognizes a more diverse
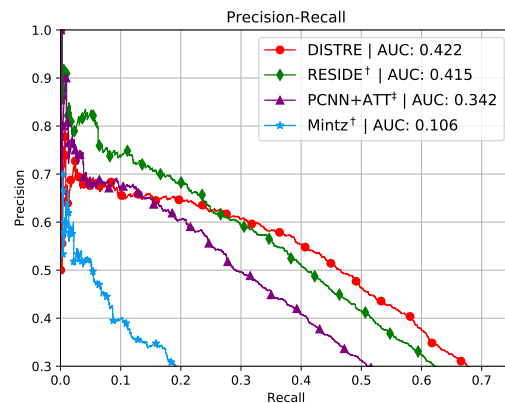

Figure 4: Precision-Recall curve on the NYT dataset. Our method (DISTRE) shows a more balanced performance across relations, especially in the long tail. † marks results reported by Vashishth et al. (2018). ‡ indicates results we obtained with the OpenNRE[4] implementation.

set of relations, while still achieving state-of-the-art AUC. Even without explicitly provided side information and linguistic features.

5.1 Held-out Evaluation

Table 1 shows the results of our model on the held-out dataset. DISTRE with selective attention achieves a new state-of-the-art AUC value of 0.422. The precision-recall curve in Figure 4 shows that it outperforms RESIDE and PCNN+ATT at higher recall levels, while precision is lower for top predicted relation instances. The results of the PCNN+ATT model indicate that its performance is only better in the very beginning of the curve, but its precision drops early and only achieves an AUC value of 0.341. Similar, RESIDE performs better in the beginning but drops in precision after a recall-level of approximately 0.25. This suggests that our method yields a more balanced overall performance, which we believe is important in many real-world applications.

Table 1 also shows detailed precision values measured at different points along the P-R curve. We again can observe that while DISTRE has lower precision for the top 500 predicted relation instances, it shows a state-of-the-art precision of 60.2% for the top 1000 and continues to perform higher for the remaining, much larger part of the predictions.

System	AUC	P@100	P@200	P@300	P@500	P@1000	P@2000
Mintz[†]	0.107	52.3	50.2	45.0	39.7	33.6	23.4
PCNN+ATT[‡]	0.341	73.0	68.0	67.3	63.6	53.3	40.0
RESIDE[†]	0.415	**81.8**	**75.4**	**74.3**	**69.7**	59.3	45.0
DISTRE	**0.422**	68.0	67.0	65.3	65.0	**60.2**	**47.9**

Table 1: Precision evaluated automatically for the top rated relation instances. † marks results reported in the original paper. ‡ marks our results using the OpenNRE implementation.

5.2 Manual Evaluation and Analysis

Since automated evaluation on a distantly supervised, held-out dataset does not reflect the actual performance of the models given false positive labels and incomplete knowledge base information, we also evaluate all models manually. This also allows us to gain a better understanding of the difference of the models in terms of their predictions. To this end, three human annotators manually rated the top 300 predicted relation instances for each model. Annotators were asked to label a predicted relation as correct only if it expressed a true fact at some point in time (e.g., for a */business/person/company* relationship, a person may have worked for a company in the past, but not currently), and if at least one sentence clearly expressed this relation, either via a syntactic pattern or via an indicator phrase.

Table 2 shows the *P@100*, *P@200*, *P@300* and average precision scores, averaged over all annotators. PCNN+ATT has the highest average precision at 94.3%, 3% higher than the 91.2% of RESIDE and 5% higher than our model. However, we see that this is mainly due to PCNN+ATT's very high *P@100* and *P@200* scores. For P@300, all models have very similar precision scores. PCNN+ATT's scores decrease considerably, reflecting the overall trend of its PR curve, whereas RESIDE's and DISTRE's manual precision scores remain at approximately the same level. Our model's precision scores for the top rated predictions are around 2% lower than those of RESIDE, confirming the results of the held-out evaluation. Manual inspection of DISTRE's output shows that most errors among the top predictions arise from wrongly labeled */location/country/capital* instances, which the other models do not predict among the top 300 relations.

Table 3 shows the distribution over relation types for the top 300 predictions of the different models. We see that DISTRE's top predictions encompass 10 distinct relation types, more than the other two models, with */location/location/contains* and */people/person/nationality* contributing 67% of the predictions. Compared to PCNN+ATT and RESIDE, DISTRE predicts additional relation types, such as e.g. */people/person/place_lived* (e.g., "Sen. PER, Republican/Democrat of LOC") and */location/neighborhood/neighborhood_of* (e.g., "the LOC neighborhood/area of LOC"), with high confidence.

RESIDE's top 300 predictions cover a smaller range of 7 distinct relation types, but also focus on */location/location/contains* and */people/person/nationality* (82% of the model's predictions). RESIDE's top predictions include e.g. the additional relation types */business/company/founders* (e.g., "PER, the founder of ORG") and */people/person/children* (e.g., "PER, the daughter/son of PER").

PCNN+ATT's high-confidence predictions are strongly biased towards a very small set of only four relation types. Of these, */location/location/contains* and */people/person/nationality* together make up 91% of the top 300 predictions. Manual inspection shows that for these relations, the PCNN+ATT model picks up on entity type signals and basic syntactic patterns, such as "LOC, LOC" (e.g., "Berlin, Germany") and "LOC in LOC" ("Green Mountain College in Vermont") for */location/location/contains*, and "PER of LOC" ("Stephen Harper of Canada") for */people/person/nationality*. This suggests that the PCNN model ranks short and simple patterns higher than more complex patterns where the distance between the arguments is larger. The two other models, RESIDE and DISTRE, also identify and utilize these syntactic patterns.

Table 4 lists some of the more challenging sentence-level predictions that our system cor-

System	P@100	P@200	P@300	Avg Prec
PCNN+ATT	97.3	94.7	90.8	94.3
RESIDE	91.3	91.2	91.0	91.2
DISTRE	88.0	89.8	89.2	89.0

Table 2: Precision evaluated manually for the top 300 relation instances, averaged across 3 human annotators.

relation	DIS	RES	PCNN
location/contains	168	182	214
person/nationality	32	65	59
person/company	31	26	19
person/place_lived	22	–	–
country/capital	17	–	–
admin_div/country	13	12	6
neighborhood/nbhd_of	10	3	2
location/team	3	–	–
company/founders	2	6	–
team/location	2	–	–
person/children	–	6	–

Table 3: Distribution over the top 300 predicted relations for each method. DISTRE achieves performance comparable to RESIDE, while predicting a more diverse set of relations with high confidence. PCNN+ATT shows a strong focus on two relations: */location/location/contains* and */people/person/nationality*.

rectly classified.

6 Related Work

Relation Extraction Initial work in RE uses statistical classifiers or kernel based methods in combination with discrete syntactic features, such as part-of-speech and named entities tags, morphological features, and WordNet hypernyms (Mintz et al., 2009; Hendrickx et al., 2010). These methods have been superseded by sequence based methods, including recurrent (Socher et al., 2012; Zhang and Wang, 2015) and convolutional neural networks (Zeng et al., 2014, 2015). Consequently, discrete features have been replaced by distributed representations of words and syntactic features (Turian et al., 2010; Pennington et al., 2014). Xu et al. (2015a,b) integrated shortest dependency path (SDP) information into a LSTM-based relation classification model. Considering the SDP is useful for relation classification, because it focuses on the action and agents in a sentence (Bunescu and Mooney, 2005; Socher et al.,

2014). Zhang et al. (2018b) established a new state-of-the-art for relation extraction on the TA-CRED dataset by applying a combination of pruning and graph convolutions to the dependency tree. Recently, Verga et al. (2018) extended the Transformer architecture by a custom architecture for supervised biomedical named entity and relation extraction. In comparison, we fine-tune pre-trained language representations and only require distantly supervised annotation labels.

Distantly Supervised Relation Extraction Early distantly supervised approaches (Mintz et al., 2009) use multi-instance learning (Riedel et al., 2010) and multi-instance multi-label learning (Surdeanu et al., 2012; Hoffmann et al., 2011) to model the assumption that at least one sentence per relation instance correctly expresses the relation. With the increasing popularity of neural networks, PCNN (Zeng et al., 2014) became the most widely used architecture, with extensions for multi-instance learning (Zeng et al., 2015), selective attention (Lin et al., 2016; Han et al., 2018), adversarial training (Wu et al., 2017; Qin et al., 2018), noise models (Luo et al., 2017), and soft labeling (Liu et al., 2017; Wang et al., 2018). Recent work showed graph convolutions (Vashishth et al., 2018) and capsule networks (Zhang et al., 2018a), previously applied to the supervised setting (Zhang et al., 2018b), to be also applicable in a distantly supervised setting. In addition, linguistic and semantic background knowledge is helpful for the task, but the proposed systems typically rely on explicit features, such as dependency trees, named entity types, and relation aliases (Vashishth et al., 2018; Yaghoobzadeh et al., 2017), or task- and domain-specific pre-training (Liu et al., 2018b; He et al., 2018), whereas our method only relies on features captured by a language model during unsupervised pre-training.

Language Representations and Transfer Learning Deep language representations have shown to be an effective form of unsupervised

Sentence	Relation
Mr. Snow asked, referring to Ayatollah **Ali Khamenei**, **Iran**'s supreme leader, and Mahmoud Ahmadinejad, **Iran**'s president.	/people/person/nationality
In **Oklahoma**, the Democratic governor, **Brad Henry**, vetoed legislation Wednesday that would ban state facilities and workers from performing abortions except to save the life of the pregnant woman.	/people/person/place_lived
Jakarta also boasts of having one of the oldest golf courses in **Asia**, Rawamangun , also known as the Jakarta Golf Club.	/location/location/contains
Cities like New York grow in their unbuilding: demolition tends to precede development, most urgently and particularly in **Lower Manhattan**, where **New York City** began.	/location/location/contains

Table 4: Examples of challenging relation mentions. These examples benefit from the ability to capture more complex features. Relation arguments are marked in bold.

pre-training. Peters et al. (2018) introduced embeddings from language models (ELMo), an approach to learn contextualized word representations by training a bidirectional LSTM to optimize a disjoint bidirectional language model objective. Their results show that replacing static pre-trained word vectors (Mikolov et al., 2013; Pennington et al., 2014) with contextualized word representations significantly improves performance on various natural language processing tasks, such as semantic similarity, coreference resolution, and semantic role labeling. Ruder and Howard (2018) found language representations learned by unsupervised language modeling to significantly improve text classification performance, to prevent overfitting, and to increase sample efficiency. Radford et al. (2018) demonstrated that general-domain pre-training and task-specific fine-tuning, which our model is based on, achieves state-of-the-art results on several question answering, text classification, textual entailment, and semantic similarity tasks. Devlin et al. (2018) further extended language model pre-training by introducing a slot-filling objective to jointly train a bidirectional language model. Most recently (Radford et al., 2019) found that considerably increasing the size of language models results in even better generalization to downstream tasks, while still underfitting large text corpora.

7 Conclusion

We proposed DISTRE, a Transformer which we extended with an attentive selection mechanism for the multi-instance learning scenario, common in distantly supervised relation extraction. While DISTRE achieves a lower precision for the 300 top ranked predictions, we observe a state-of-the-art AUC and an overall more balanced performance, especially for higher recall values. Similarly, our approach predicts a larger set of distinct relation types with high confidence among the top predictions. In contrast to RESIDE, which uses explicitly provided side information and linguistic features, our approach only utilizes features implicitly captured in pre-trained language representations. This allows for an increased domain and language independence, and an additional error reduction because pre-processing can be omitted.

In future work, we want to further investigate the extent of syntactic structure captured in deep language language representations. Because of its generic architecture, DISTRE allows for integration of additional contextual information, e.g. background knowledge about entities and relations, which could also prove useful to further improve performance.

Acknowledgments

References

Christoph Alt, Marc Hübner, and Leonhard Hennig. 2019. Improving relation extraction by pre-trained language representations. In *Proceedings of the 2019 Conference on Automated Knowledge Base Construction*, Amherst, Massachusetts.

Razvan C. Bunescu and Raymond J. Mooney. 2005. A Shortest Path Dependency Kernel for Relation Extraction. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 724–731. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *Computing Research Repository (CoRR)*, abs/1810.04805.

Miao Fan, Deli Zhao, Qiang Zhou, Zhiyuan Liu, Thomas Fang Zheng, and Edward Y. Chang. 2014. Distant Supervision for Relation Extraction with Matrix Completion. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 839–849, Baltimore, Maryland. Association for Computational Linguistics.

Xu Han, Pengfei Yu, Zhiyuan Liu, Maosong Sun, and Peng Li. 2018. Hierarchical relation extraction with coarse-to-fine grained attention. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2236–2245. Association for Computational Linguistics.

Zhengqiu He, Wenliang Chen, Zhenghua Li, Meishan Zhang, Wei Zhang, and Min Zhang. 2018. See: Syntax-aware entity embedding for neural relation extraction. In *AAAI*.

Iris Hendrickx, Su Nam Kim, Zornitsa Kozareva, Preslav Nakov, Diarmuid Ó Séaghdha, Sebastian Padó, Marco Pennacchiotti, Lorenza Romano, and Stan Szpakowicz. 2010. Semeval-2010 Task 8: Multi-Way Classification of Semantic Relations between Pairs of Nominals. In *SemEval@ACL*.

Raphael Hoffmann, Congle Zhang, Xiao Ling, Luke Zettlemoyer, and Daniel S. Weld. 2011. Knowledge-Based Weak Supervision for Information Extraction of Overlapping Relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 541–550, Portland, Oregon, USA. Association for Computational Linguistics.

Heng Ji and Ralph Grishman. 2011. Knowledge base population: Successful approaches and challenges. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 1148–1158, Stroudsburg, PA, USA. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, abs/1412.6980.

Yankai Lin, Shiqi Shen, Zhiyuan Liu, Huanbo Luan, and Maosong Sun. 2016. Neural Relation Extraction with Selective Attention over Instances. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2124–2133, Berlin, Germany. Association for Computational Linguistics.

Peter J. Liu, Mohammad Saleh, Etienne Pot, Ben Goodrich, Ryan Sepassi, Lukasz Kaiser, and Noam Shazeer. 2018a. Generating wikipedia by summarizing long sequences. *ICLR*.

Tianyi Liu, Xinsong Zhang, Wanhao Zhou, and Weijia Jia. 2018b. Neural relation extraction via inner-sentence noise reduction and transfer learning. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2195–2204. Association for Computational Linguistics.

Tianyu Liu, Kexiang Wang, Baobao Chang, and Zhifang Sui. 2017. A soft-label method for noise-tolerant distantly supervised relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1790–1795. Association for Computational Linguistics.

Bingfeng Luo, Yansong Feng, Zheng Wang, Zhanxing Zhu, Songfang Huang, Rui Yan, and Dongyan Zhao. 2017. Learning with noise: Enhance distantly supervised relation extraction with dynamic transition matrix. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 430–439. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Bonan Min, Ralph Grishman, Li Wan, Chang Wang, and David Gondek. 2013. Distant Supervision for Relation Extraction with an Incomplete Knowledge Base. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 777–782, Atlanta, Georgia. Association for Computational Linguistics.

Mike Mintz, Steven Bills, Rion Snow, and Daniel Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *ACL/IJCNLP*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke S. Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL-HLT*.

Pengda Qin, Weiran XU, and William Yang Wang. 2018. Dsgan: Generative adversarial training for distant supervision relation extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 496–505. Association for Computational Linguistics.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *available as a preprint*.

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 *Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases (ECML PKDD '10)*.

Sebastian Ruder and Jeremy Howard. 2018. Universal language model fine-tuning for text classification. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1715–1725.

Richard Socher, Brody Huval, Christopher D. Manning, and Andrew Y. Ng. 2012. Semantic compositionality through recursive matrix-vector spaces. In *EMNLP-CoNLL*.

Richard Socher, Andrej Karpathy, Quoc V. Le, Christopher D. Manning, and Andrew Y. Ng. 2014. Grounded compositional semantics for finding and describing images with sentences. *TACL*, 2:207–218.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D. Manning. 2012. Multi-instance Multi-label Learning for Relation Extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 455–465, Stroudsburg, PA, USA. Association for Computational Linguistics.

Joseph P. Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *ACL*.

Shikhar Vashishth, Rishabh Joshi, Sai Suman Prayaga, Chiranjib Bhattacharyya, and Partha Talukdar. 2018. Reside: Improving distantly-supervised neural relation extraction using side information. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1257–1266. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008.

Patrick Verga, Emma Strubell, and Andrew McCallum. 2018. Simultaneously self-attending to all mentions for full-abstract biological relation extraction. 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 872–884. Association for Computational Linguistics.

Guanying Wang, Wen Zhang, Ruoxu Wang, Yalin Zhou, Xi Chen, Wei Zhang, Hai Zhu, and Huajun Chen. 2018. Label-free distant supervision for relation extraction via knowledge graph embedding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2246–2255. Association for Computational Linguistics.

Yi Wu, David Bamman, and Stuart Russell. 2017. Adversarial training for relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1778–1783. Association for Computational Linguistics.

Kun Xu, Yansong Feng, Songfang Huang, and Dongyan Zhao. 2015a. Semantic relation classification via convolutional neural networks with simple negative sampling. In *EMNLP*.

Yan Xu, Lili Mou, Ge Li, Yunchuan Chen, Hao Peng, and Zhi Jin. 2015b. Classifying relations via long short term memory networks along shortest dependency paths. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1785–1794. Association for Computational Linguistics.

Yadollah Yaghoobzadeh, Heike Adel, and Hinrich Schütze. 2017. Noise mitigation for neural entity typing and relation extraction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1183–1194. Association for Computational Linguistics.

Mo Yu, Wenpeng Yin, Kazi Saidul Hasan, Cícero Nogueira dos Santos, Bing Xiang, and Bowen Zhou. 2017. Improved neural relation detection for knowledge base question answering. In *ACL*.

Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant Supervision for Relation Extraction via Piecewise Convolutional Neural Networks. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1753–1762, Lisbon, Portugal. Association for Computational Linguistics.

Daojian Zeng, Kang Liu, Siwei Lai, Guangyou Zhou, and Jun Zhao. 2014. Relation classification via convolutional deep neural network. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 2335–2344. Dublin City University and Association for Computational Linguistics.

Dongxu Zhang and Dong Wang. 2015. Relation classification via recurrent neural network. *arXiv preprint arXiv:1508.01006*.

Ningyu Zhang, Shumin Deng, Zhanling Sun, Xi Chen, Wei Zhang, and Huajun Chen. 2018a. Attention-based capsule networks with dynamic routing for relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 986–992. Association for Computational Linguistics.

Yuhao Zhang, Peng Qi, and Christopher D. Manning. 2018b. Graph Convolution over Pruned Dependency Trees Improves Relation Extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215, Brussels, Belgium. Association for Computational Linguistics.

Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.