A Training and Hyperparameters

We conduct experiments based on the 200-D pre-trained GloVe, the 300-D pre-trained FastText, 300-D randomly initialized word vectors and uncased BERT-base representation respectively. For the SemEval-2018 dataset, we train the domain-specific word embeddings, like (Rotsztejn et al., 2018), using GloVe and Fasttext respectively. Our corpus is composed of all the abstracts since 2001 (5.4 million tokens) collected using the API [1] on arXiv.org and the ACL ARC corpus [2] (90 million tokens). We trained the word embeddings for 500 epochs with 60 threads. We use 50-D randomly initialized character embeddings. For the TACRED dataset, we use the same embeddings (300-D GloVe) following (Zhang et al., 2018).

For regularization we apply dropout with $p = 0.5$. The output dimension of character CNN is 100-D. We set LSTM hidden size to 300. We set $\lambda = 1.0$. We employ the ReLU function for non-linearities. To select better models, we divide the NYT training data into 100 pieces and the training data of SemEval-2018 into four pieces. For the NYT and SemEval-2018 dataset, we use the Adam optimization algorithm to update the model parameters with an initial learning rate of 0.001 and a decay rate of 0.9. For the TACRED dataset, we use Stochastic Gradient Descent with an initial learning rate of 0.3 and a decay rate of 0.9. We use a cutoff of 5.0 for gradient clipping. We conducted experiments on an Intel(R) Xeon(R) CPU E7-4830 v3 @ 2.10GHz (Mem: 976G) and the GPU Tesla K40c.

B Ablation Study

Table 1 shows an ablation study of multi-task training and pipeline training. Two system denotes a fine-tuned NER system and a GRNN system. Here we use a BERT model as the NER system. A first observation is that a fine-tuned NER system can slightly improve the final results by ?%. Because it reduces the unknowns in the testing stage by explicitly indicate the entities. The draw back is that the training process is a multi-stage pipeline and relation candidates are determined by the NER system. We first fine tune the NER system and then predict the label sequence $\hat{Y}_{ner}$ which are also written to disk. Then we train the joint entity and relation extraction model that can use the $\hat{Y}_{ner}$ information. Although the NER system is enhanced, some entities still hurt the final precision. This is because extracting more entities does not always help the relation extraction subtask and some entities may increase the risk of predicting false positives. When we replace $\hat{Y}_{ner}$ with the ground truth NER labels, the result is significantly improved (?%). This means that a high quality label sequence can help imporve the final results.

Removing the multi-task training degrades the performance by ?%. This means that multi-task training benefits the RE subtask from the NER subtask. When we remove the NER system, performance drops slightly (?%). Then, we remove the NER system and multi-task training. We first train the NER subtask, then freeze the parameters of shared layer, and then train the RE subtask. The result is reduced by ?% because the RE subtask cannot be encoded at the lower layer to interact with the NER subtask. This means that multi-task joint training is critical for subtask interaction. When we do not freeze the lower layer, the result is much lower. Because when we train the RE subtask, the memory for the NER subtask will be reduced.

We also apply our training pipeline to BERT, a pre-trained bidirecftional transformer that has proven very effective in many NLP tasks. BERT has hundreds of millions parameters and has been trained on large-scale corpus. We fine tune a simple BERT model for TE. Table 2 shows the result. When only using multi-task training, BERT achieves slightly higher results than our GRNN model. The downside is the cost of computation. When we remove the multi-task, NER system and Frozen NER, this model almost forgets all the NER memory, so the RE subtask fails. Model fine-tuning can achieve different functions, which also means that BERT is sensitive to parameter changes.

C Case Study

It is instructive to analyze which words the model is attending for relation representation. We hand-picked some examples in the SemEval-2018 dataset and visualize the attention patterns of these samples.

Figure 1 shows how our model extracts informative words for relation representation. The first column is sample id. The second column contains

[1] https://arxiv.org/help/api/index
[2] http://acl-arc.comp.nus.edu.sg/

Table 1: SwitchNet setting ablation

Model	Precision	Recall	F1
Two system + multi-task training	56.78	51.89	54.23
+ NER label	61.18	57.46	59.26
–multi-task	55.64	51.13	53.29
–NER system	60.28	52.27	55.95
–multi-task, NER system	50.33	56.96	53.44
–multi-task, NER system, Frozen NER	55.88	43.29	48.78

Table 2: BERT model setting ablation

Model	Precision	Recall	F1
Two system + multi-task training	51.30	54.93	53.05
+ NER label	54.89	61.01	57.79
–multi-task	50.46	54.43	52.37
–NER system	50.68	56.20	53.30
–multi-task, NER system, Frozen NER	–	–	–

Id	Triple	Sentences
1	(English-Chinese bitexts, PART_WHOLE, Web)	This piece of work has also laid a foundation for exploring and harvesting English-Chinese bitexts in a larger volume from the Web .
		This piece of work has also laid a foundation for exploring and harvesting English-Chinese bitexts in a larger volume from the Web .
2	(domains, COMPARE, MUC-4 terrorism domain)	These previous domains were much narrower than the MUC-4 terrorism domain .
		These previous domains were much narrower than the MUC-4 terrorism domain .
3	(paper, TOPIC, overview)	An overview of HowNet and information structure are described in this paper .
		An overview of HowNet and information structure are described in this paper .
4	(beam-search decoder, MODEL-FEATURE, Translations)	Translations are produced by means of a beam-search decoder .
		Translations are produced by means of a beam-search decoder .
5	(Bayesian classifiers, RESULT, recall performance)	In our evaluation , Bayesian classifiers produce the best recall performance of 80 % but the precision is low (60%) .
		In our evaluation , Bayesian classifiers produce the best recall performance of 80 % but the precision is low (60%) .
6	(WordNet, USAGE, Word Sense Disambiguation (WSD) task)	WordNet has been used extensively as a resource for the Word Sense Disambiguation (WSD) task , both as a sense inventory and a repository of semantic relationships .
		WordNet has been used extensively as a resource for the Word Sense Disambiguation (WSD) task , both as a sense inventory and a repository of semantic relationships .

Figure 1: Visualization of some cases

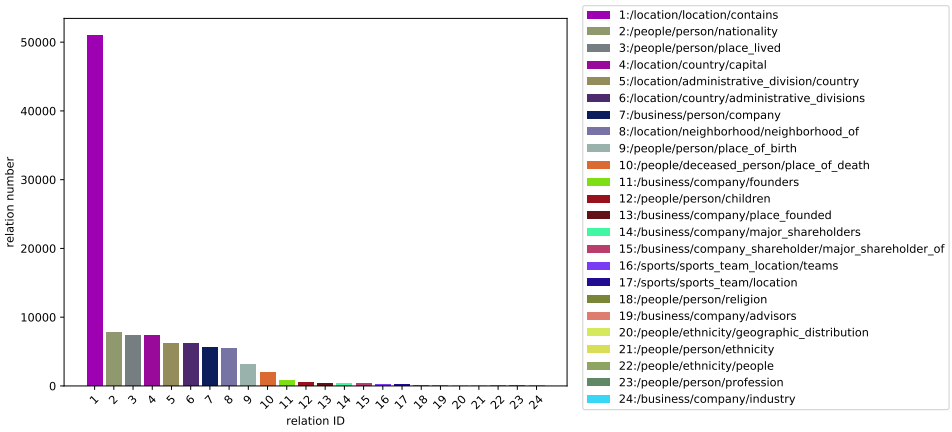
the extracted triples, and the third column shows the textual input. The underlined phrase represents an entity in the extracted triple. The red degree denotes the word weight for the relation representation. The first sentence shows that this model is capable of focusing on informative words to identify the "*PART_WHOLE*" relation type for "*English-Chinese Bitexts*" and "*Web*". The second sentence shows this model resolves the comparative relation by attending to *narrower than*. The third sentence shows that "*are described in*" means the "*TOPIC*" relation.

The fourth sentence shows that "*produced by means*" for "*Translations*" and "*beam search decoder*" denotes the "*MODEL-FEATURE*" relation, while the fifth sentence shows this model extracts the relation type "*RESULT*" by attending to "*produce best*". This suggests that this model considers the entity semantics and sentence context. The sixth sentence shows that this model can extract the informative tokens "*resource for*" instead of the important verb in a long context. These results indicate that the model considers the entity semantics and the relation context while attending to the informative words for relation representation.

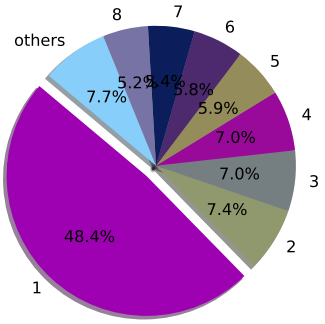
This model have two insights. We incorporate inference and interpretability into the process of model learning. First, we propose to use inference scheme to make it a lifelong learing model. This model can adaptively switch between tasks and continuously optimize model parameters based on historical memory. Second, we present a scheme for extracting informative tokens for relation representation. SwithcNet learns the attention function, and then we can use the distribution of attention to explain the relation classification process in a deep learning model. Multi-task learning is a convenient way to unfiy two subtasks in the learning process.

D Dataset Overview

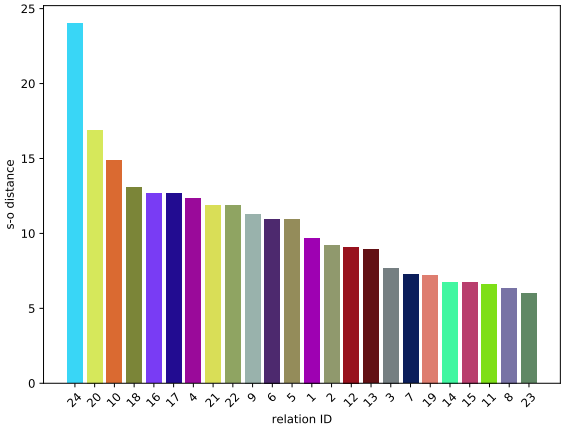
Figure 2a shows the relation distribution of the NYT dataset. The x- and y-axes are the relation



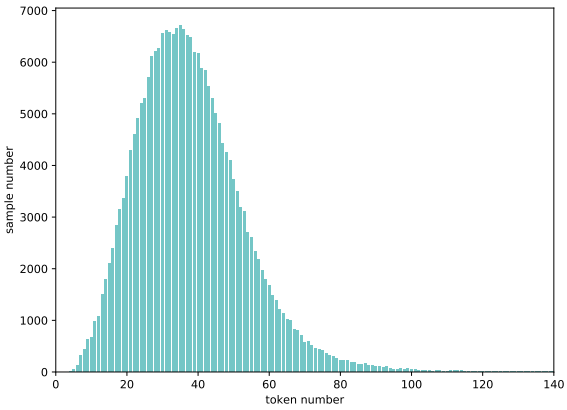
(a) Relation number



(b) Relation portion



(c) s-o distance



(d) Token number

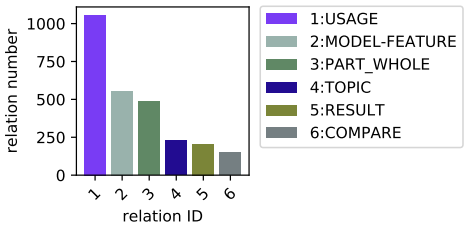
Figure 2: NYT Dataset visualization

ID and relation number respectively. We first observe there are more than 50,000 "*contains*" relations while others are less than 9,000. Each relations from 11 to 24 occurs less than 1,000 times. Figure 2b shows that "*contains*" relation occupies the main part (48.4%), while the relations from 9 to 24 account for 7.7%. This dataset has class imbalanace problem, which poses a challenge to model performance. Figure 2c shows the average distance of each relation type between subject and object. The x- and y-axes are relation ID and token number respectively. We observe that "*industry*", "*geographic_distribution*", "*place_of_death*" relations are often described in a long context, and "*profession*", "*neighborhood_of*" and "*founders*" are more likely described in a short context. Figure 2d shows the token number of samples. The x- and y- axes denote the token number and sam-

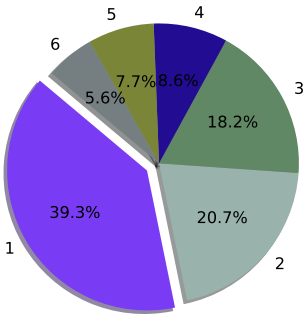
ple number respectively. We observe the average token number is 37.8.

Figure 3a shows the relation distribution of the Semeval-2018 dataset. "*USAGE*" relation occupies the main part (39.3%). In this data set, the distribution of relation types is relatively uniform. Figure 3c shows the "*COMPARE*" relation is more likely described with more words, while the "*MODEL-FEATURE*" and "*PART_WHOLE*" relations are more likely expressed with less words. Figure 3d shows the token number of samples. We observe the average token number is 25.8.

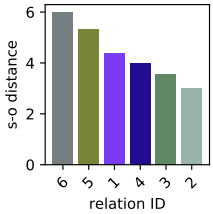
Figure 4a shows the relation distribution of the TACRED. "*per:title*" relation occupies the main part (17.7%). Figure 4c shows the "*per:stateorprovince_of_death*" relation is more likely described with more words, while the "*org:website*" and "*org:alternate_names*" relations are more likely expressed with less words. Figure 4d shows the token number of samples. We observe the average token number is 36.4.



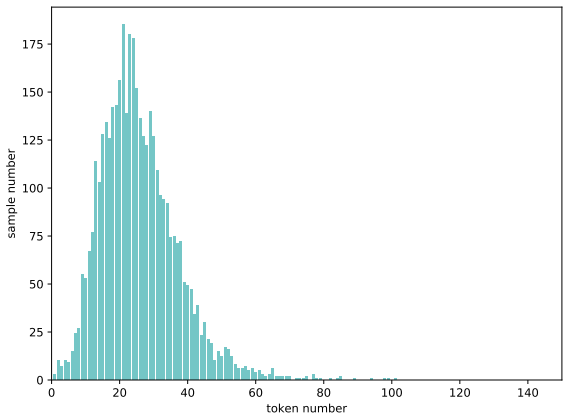
(a) Relation number



(b) Relation portion

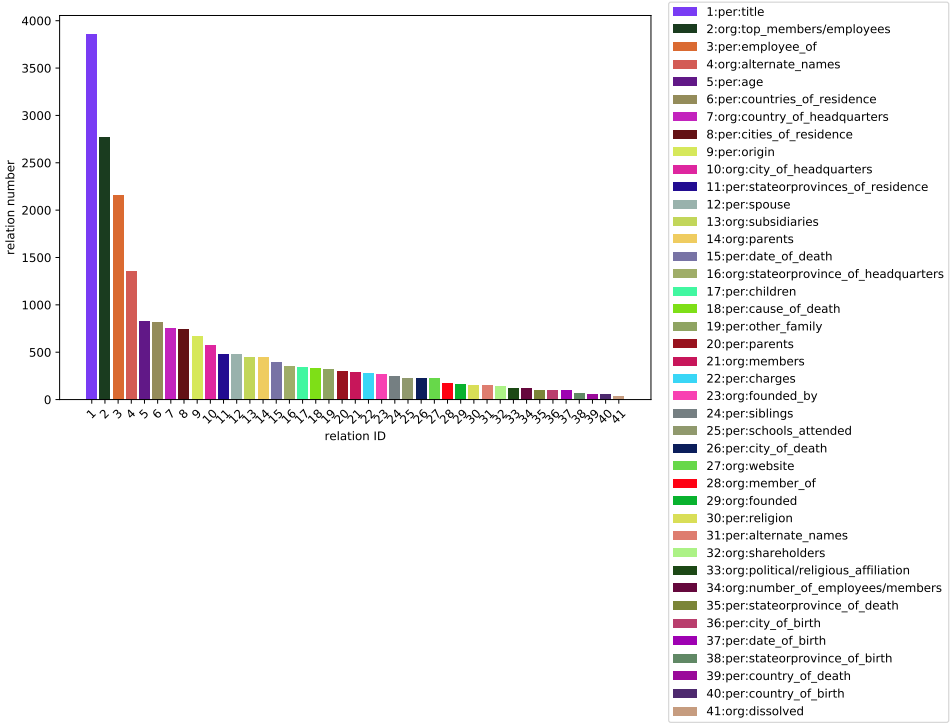


(c) s-o distance

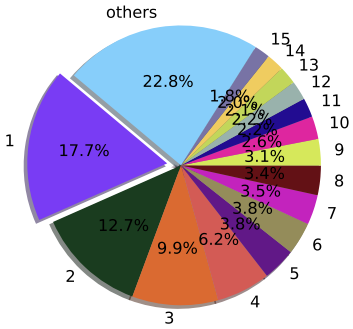


(d) Token number

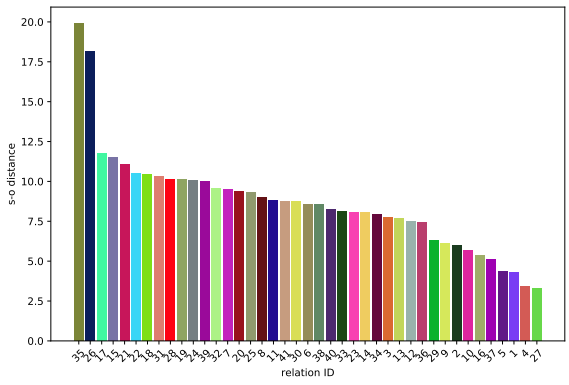
Figure 3: SemEval-2018 dataset visualization



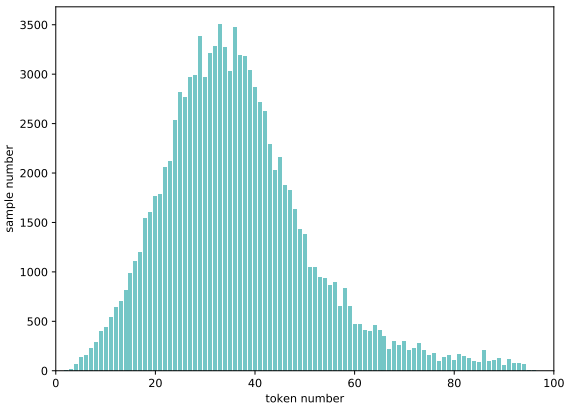
(a) Relation number



(b) Relation portion



(c) s-o distance



(d) Token number

Figure 4: TACRED visualization