A Hyperparameters

For the NELL dataset, we set embedding size as 100. For Wikidata, we set the embedding size as 50 for faster training with millions of triples. The embeddings are trained for 1,000 epochs. The other hyperparamters are tuned using the Hits@10 metric[6] on the validation tasks. For matching steps, the optimal setting is 2 for NELL-One and 4 for Wiki-One. For the number of neighbors, we find that the maximum limit 50 works the best for both datasets. For parameter updates, we use Adam (Kingma and Ba, 2014) with the initial learning rate 0.001 and we half the learning rate after 200k update steps. The margin used in our loss function is 5.0. The dimension of LSTM's hidden size is 200.

B Few-Shot Experiments

Metrics	GMatching		ComplEx	
	1-shot	5-shot	1-shot	5-shot
MRR	.132/.185	.178/.201	.072/.131	.113/.200
Hits@10	.308/.313	.307/.311	.128/.223	.221/.325
Hits@5	.232/.260	.241/.264	.041/.086	.160/.269
Hits@1	.049/.119	.109/.143	.041/.086	.113/.133

Table 5: 5-shot experiments on NELL-One.

Although our model is designed for one-shot settings, it can be directly applied to the k-shot ($k > 1$) setting by applying an aggregation function over the hidden states or scores of all the k examples in the support set.

This section considers the 5-shot case on NELL-One, and simply ranks the candidates with the maximum of the five scores $score_K^{s_i}$ according to Eq. 4, where s_i ($i \in 1, 5$) are 5 support entity-pairs. Table 5 shows the results. We can see that with more training triples, the performance of the KG embedding model improves a lot, while our model's improvement is limited. We think this is because (1) our model handles each example individually and ignores their interactions; (2) our model does not perform any training on the meta-testing relations, while the KG embedding methods benefit more from re-training when the number of labeled tuples k increases. Future research could consider using more advanced aggregation module such as attention to model the interaction of multiple training examples; as well as exploring meta-learning approaches (Ravi and Larochelle, 2017) to enable fast adaptation of metrics during meta-testing.

[6] The percentage of correct answer ranks within top10.