MCMH: Learning Multi-Chain Multi-Hop Rules for Knowledge Graph Reasoning

Lu Zhang ♣ Mo Yu ♡ Tian Gao ♡ Yue Yu ♣
♣ Lehigh University ♡ IBM Research
luz319@lehigh.edu {yum, tgao}@us.ibm.com yuy214@lehigh.edu

Abstract

Multi-hop reasoning approaches over knowledge graphs infer a missing relationship between entities with a multi-hop rule, which corresponds to a chain of relationships.

We extend existing works to consider a generalized form of multi-hop rules, where each rule is a set of relation chains. We propose a two-step approach that first selects a small set of relation chains as a rule and then evaluates the confidence of the target relationship by jointly scoring the selected chains.

A game-theoretical framework is proposed to this end to simultaneously optimize the rule selection and prediction steps. Empirical results show that our **multi-chain multi-hop (MCMH)** rules result in superior results compared to the standard single-chain approaches.

Introduction

Our model utilizes deducible paths of tuples to do reasoning. We propose the concept of MCMH rule set, a set of multi-hop rules. Multiple chains leverage the inference in two ways:

- Logic conjunction of multiple chains (Figure 1b);
- Aggregation of multiple pieces of evidence (Figure 1c);

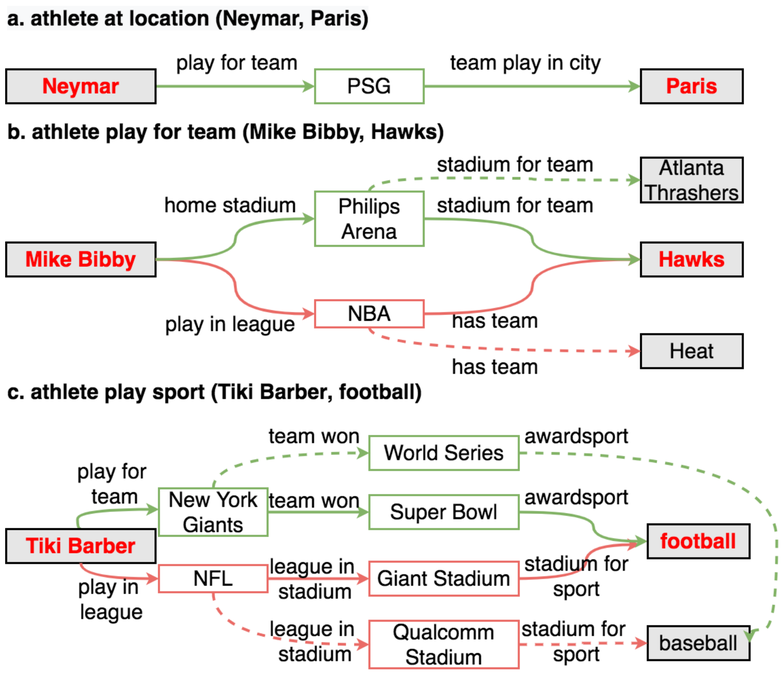

Figure 1:Examples of reasoning with multiple paths. Solid lines are selected paths (different colors indicate different paths), and dotted lines are unselected paths.

A Three-Player Game for Rule Learning

The input is a set of chains $\mathcal{R}_i \subset \mathcal{R}$ for relation $\hat{r}$ and each training sample $(\hat{h}_i, \hat{r}, \hat{t}_i)$.

Workflow of Our Method:

1. **Extract a fixed hop k sub-graph from the original KB**. Each sub-graph starts with an entity $\hat{h}$, ends with an entity $\hat{t}$, and satisfies $(\hat{h}, \hat{r}, \hat{t}) \in \mathcal{G}$. The sub-graph consists of multiple m-hop paths connecting the two ends, where $1 \leq m \leq k$. Each of the m-hop paths has the form $(\hat{h}, r^1, t^1), (t^1, r^2, t^2), \cdots (t^{m-1}, r^m, \hat{t})$. We call $r^1 \rightarrow r^2 \cdots \rightarrow r^m$ a candidate relation chain R.
2. **One-hot encoding.** For every sample $(\hat{h}_i, \hat{r}, \hat{t}_i)$. We encode chains between $(\hat{h}_i, \hat{t}_i)$ with one-hot encoding as a $m * m$ 0-1 matrix.
3. **Rule set generator** that selects the set of chains $\mathcal{S}_i$ as a rule.
4. **Predictor** that predicts the probability of $\hat{r}$ based on $\mathcal{S}_i$,denoted as $\hat{p}(\hat{r}|\mathcal{S}_i)$.
5. **Complement predictor** that estimates probability of $\hat{r}$ conditioned on $\mathcal{S}_i^c$, denoted as $\hat{p}^c(\hat{r}|\mathcal{S}_i^c)$.

Optimize Predictor and Complement Predictor:

$$\mathcal{L}_p = \min_{\hat{p}} -H(p(\hat{r}|\mathcal{S}_i); \hat{p}(\hat{r}|\mathcal{S}_i)), \qquad \mathcal{L}_c = \min_{\hat{p}^c} -H(p(\hat{r}|\mathcal{S}_i^c); \hat{p}^c(\hat{r}|\mathcal{S}_i^c)), \tag{1}$$

$H(p;q)$ denotes the cross entropy between p and q, and $p(\cdot|\cdot)$ denotes the empirical distribution.

Optimize Generator:

$$\min_{g(\cdot)} \mathcal{L}_p - \mathcal{L}_c + \lambda_s \mathcal{L}_s, \tag{2}$$

where $\mathcal{L}_p$ and $\mathcal{L}_c$ are the losses of the *predictor* and the *complement predictor*, respectively. $\mathcal{L}_s$ is a sparsity loss which aims to constrain the number of chains to be select to a desired size d:

$$\mathcal{L}_s = \max\{(|\mathcal{S}_i| - d)/|\mathcal{R}_i|, 0\}. \tag{3}$$

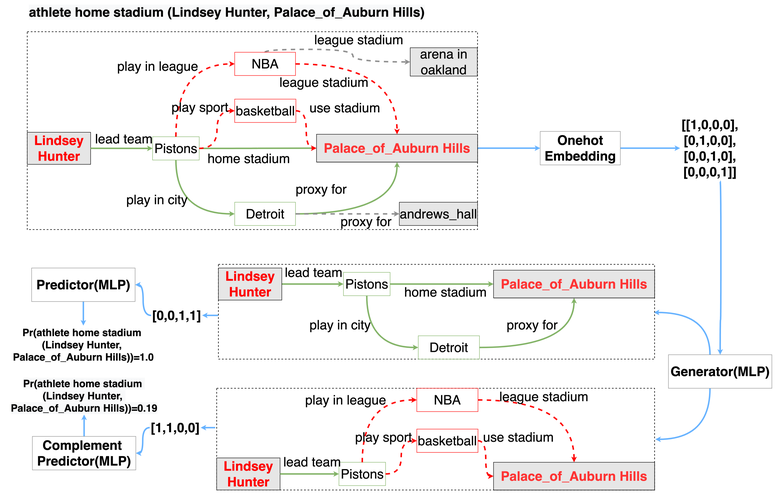

Figure 2:An example workflow of our model. The generator selects the green chains as the "critical information" for prediction. The predictor $\mathcal{S}_i$ is encoded as $\mathbf{v}_{\mathcal{S}_i}$=$[0, 0, 1, 1]$ and estimates probability of $\hat{r}$ being true as 100%. The complement predictor $\mathcal{S}_i^c$ is encoded as $\mathbf{v}_{\mathcal{S}_i^c}$=$[1, 1, 0, 0]$ and estimates the probablity as 19%.

Results

	Relation	Single-Chain Baseline	Ours d=2	Ours d=5	Ours (-conj) d=2	Ours (-conj) d=5	DeepPath	MINERVA
NELL-995	athletePlaysForTeam	0.872	0.940*	**0.947***	0.900	0.897	0.750	0.824
	athletePlaysInLeague	0.962	0.977*	**0.981***	0.957	0.975	0.960	0.970
	athleteHomeStadium	0.892	**0.896**	0.895	0.856	0.854	0.890	0.895
	athletePlaysSport	0.916	0.978*	0.982*	0.932	0.978	0.957	**0.985**
	teamPlaySports	0.728	0.769	0.782	0.669	0.771	0.738	**0.846**
	orgHeadquarterCity	0.957	0.932	0.907	**0.962**	0.903	0.790	0.946
	worksFor	0.794	0.842*	**0.849***	0.811	0.842	0.711	0.825
	bornLocation	0.823	**0.902***	0.850*	0.874	0.872	0.757	0.793
	personLeadsOrg	0.833	0.832	0.813	0.832	0.822	0.795	**0.851**
	orgHiredPerson	0.833	0.825	0.814	0.837	**0.855**	0.742	0.851
	Average	*0.861*	***0.890***	*0.882*	*0.863*	*0.877*	*0.809*	*0.879*
FB15K-237	teamSports	0.740	0.739	0.769*	0.758	0.765	**0.955**	-
	birthPlace	0.463	0.505*	**0.566***	0.443	0.512	0.531	-
	filmDirector	0.303	0.368	0.411*	0.363	0.413	**0.441**	-
	filmWrittenBy	0.498	0.516*	**0.553***	0.507	0.518	0.457	-
	filmLanguage	0.632	0.665*	**0.678***	0.667	0.675	0.670	-
	tvLanguage	**0.975**	0.962	0.957	0.957	0.956	0.969	-
	capitalOf	0.648	0.795	**0.825***	0.820	0.786	0.783	-
	orgFounded	0.465	0.407	**0.490***	0.431	0.485	0.309	-
	musicianOrigin	0.376	0.408*	**0.516***	0.390	0.476	0.514	-
	personNationality	0.713	0.806*	**0.828***	0.703	0.760	0.823	-
	Average	*0.581*	*0.617*	***0.659***	*0.604*	*0.635*	*0.645*	-

Overall Results (MAP) on NELL-995 and FB15K-237. * highlights the cases where our MLP model outperforms the baseline with statistical significance (p-value<0.01 in t-test).

Discussion

Effects of numbers of chains in one rule (d)
On NELL-995: d=2 is better than d=5. On FB15K-237 d=5 is much better. Because tuples in FB15K-237 contain more chains in average.

We select d=5 as evidence with following reasons:

- The average number of chains is 13.8 for NELL-995 and 63.3 for FB15K-237. d=5 chains are a significant portion of the whole input space.
- MAP of our model using all candidate chains is 0.671 for FB15K-237 and 0.892 for NELL-995, which are close to that of d=5

Effects of MLP versus linear predictors
Linear models **Ours (-conj)** improve a lot over single-chain baseline, which shows that most of the relations mainly benefit from the case of confidence enhancement.

Conclusion

- We formalize the concept of multi-hop rule sets with multiple relation chains from KGs.
- We propose a game-theoretical learning approach to efficiently select predictive relation chains for a query relation.
- Our model outperforms state of the art works by up to 11.21%.