



Differentiable Learning of Logical Rules for Knowledge Base Reasoning

31st Conference on NIPS 2017

Fan Yang, Zhilin Yang, William W. Cohen

2024-08-20

HoonUi Lee

Contents

Logical rules

Previous work

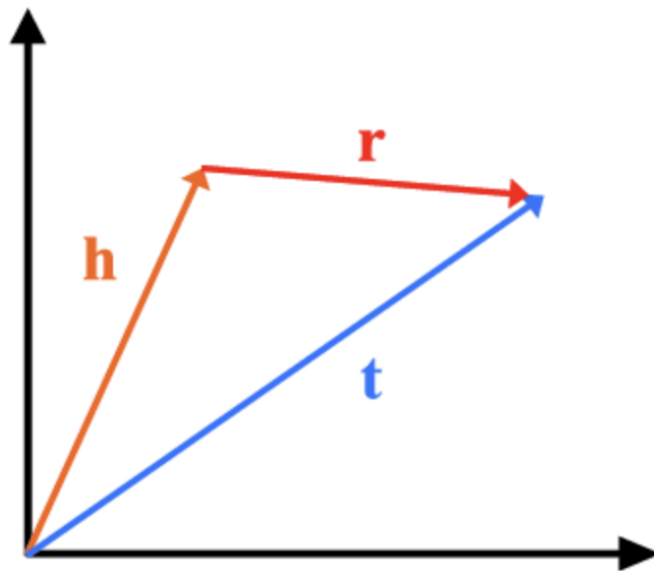
Neural LP

Experiment

Conclusion

Logical rules

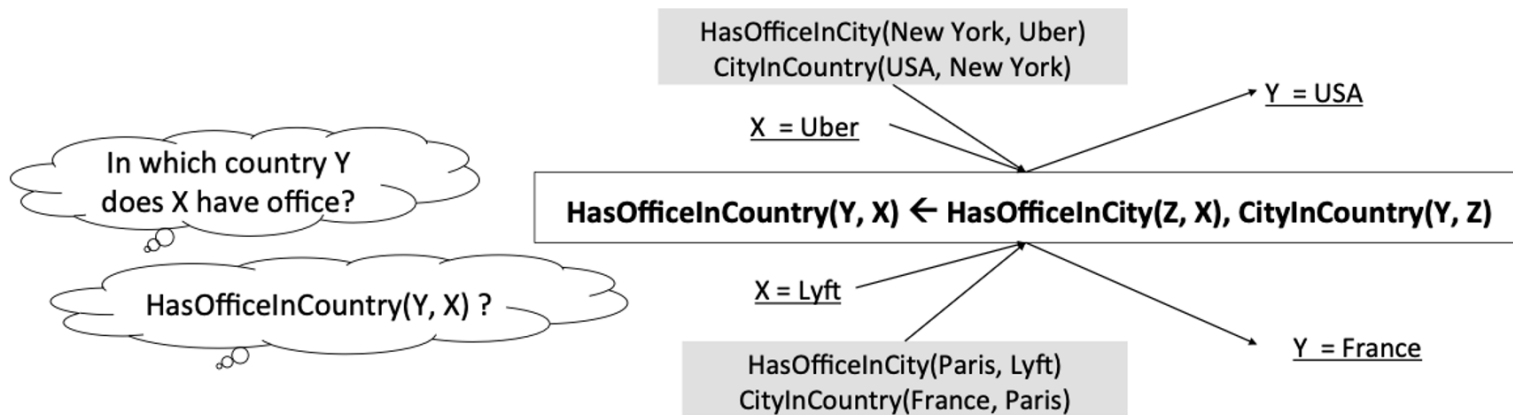
TransE



*Transductive learning*에서 효과적

In Inductive?

Logical rules



어떠한 과정을 거쳐 이런 결과로 추론한 것일까?

“과정”에 대한 해석력 => model의 robustness

Previous work

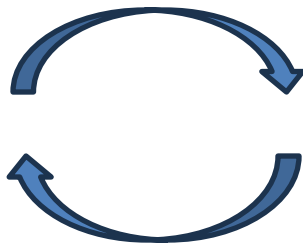
Markov Logic Networks, ProPPR in *Statistical relational learning*

Structure

(Model이 포함하는 rule 조합)



이산적 탐색을 통해
최적의 rule 조합 구성



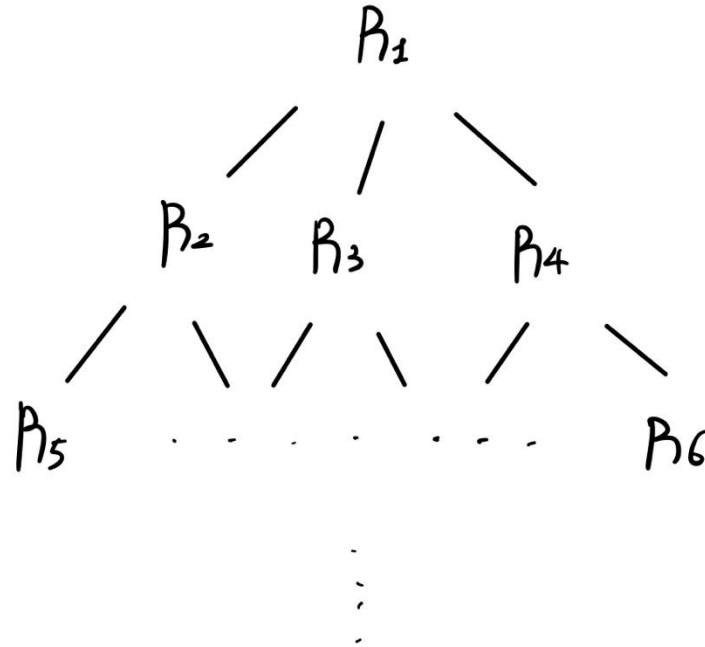
Parameter

(각 rule에 대한 Confidence)



조합된 rule 집합을 바탕으로
Rule들의 confidence 조정

Previous work



생성된 Rule 조합 : $R_1 - R_2 - R_5 - \dots$

NeuralLP

$$\alpha \text{ query } (Y, X) \leftarrow R_n(Y, Z_n) \wedge \cdots \wedge R_1(Z_1, X)$$

X : Uber

Query : HasOfficeInCity

Y 후보 : New York, San Francisco

=> 여러 Y 들의 rank 반환

NeuralLP

Learning problem using TensorLog

$$\max_{\{\alpha_l, \beta_l\}} \sum_{\{\mathbf{x}, \mathbf{y}\}} \text{score}(\mathbf{y} \mid \mathbf{x}) = \max_{\{\alpha_l, \beta_l\}} \sum_{\{\mathbf{x}, \mathbf{y}\}} \mathbf{v}_y^T \left(\sum_l (\alpha_l (\prod_{\mathbf{k} \in \beta_l} \mathbf{M}_{\mathbf{R}_k} \mathbf{v}_x)) \right)$$

$$\mathbf{v}_i \in \{0, 1\}^{|\mathbf{E}|} \quad \mathbf{M}_R : \{0, 1\}^{|\mathbf{E}| \times |\mathbf{E}|}$$

α_l : rule l 의 confidence

β_l : rule l 을 구성하는 relation 조합



Rule의 고정된 조합에 대한 confidence 최적화

여전히 이산적이다!

NeuralLP

logical rule 기반의 추론을 **미분 가능한 형태 (행렬 연산)** 로 변환

=> Gradient 기반의 최적화 가능

$$\prod_{t=1}^T \sum_k a_t^k \mathbf{M}_{\mathbf{R}_k}$$

Rule을 구성하는 **relation**에 대해 **가중치** 부여

=> Rule 나열과 confidence 할당을 결합

T : rule의 최대 길이

$|\mathbf{R}|$: Knowledge base 내에 존재하는 relation 개수

NeuralLP

Rule들의 최대 길이 τ 가 동일해야 한다는 전제 존재

$\mathbf{u}_0 = \mathbf{v}_x$ 이전 단계 메모리 벡터들의 가중합 계산

$$\mathbf{u}_t = \sum_k a_t^k \mathbf{M}_{R_k} \left(\sum_{\tau=0}^{t-1} b_t^\tau \mathbf{u}_\tau \right) \quad \text{for } 1 \leq t \leq T$$

$$\mathbf{u}_{T+1} = \sum_{\tau=0}^T b_{T+1}^\tau \mathbf{u}_\tau$$

Recurrent formulation 사용

가장 적절한 t

NeuralLP

In *RNN*

$$\mathbf{h}_t = \text{update}(\mathbf{h}_{t-1}, \text{input})$$

$$\mathbf{a}_t = \text{softmax}(W\mathbf{h}_t + b)$$

$$\mathbf{b}_t = \text{softmax}([\mathbf{h}_0, \dots, \mathbf{h}_{t-1}]^T \mathbf{h}_t)$$

NeuralLP

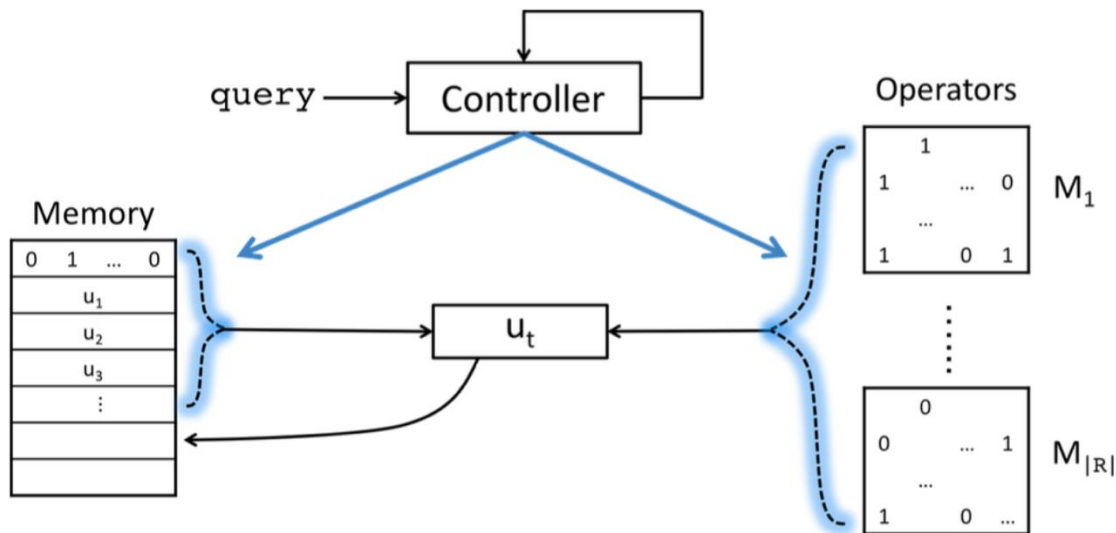


Figure 2: The neural controller system.

Experiment: Statistical relation learning

Table 1: Datasets statistics.

	# Data	# Relation	# Entity
UMLS	5960	46	135
Kinship	9587	25	104

Table 2: Experiment results. T indicates the maximum rule length.

	ISG		Neural LP	
	$T = 2$	$T = 3$	$T = 2$	$T = 3$
UMLS	43.5	43.3	92.0	93.2
Kinship	59.2	59.0	90.2	90.1

ISG (Iterative Structural Gradient)

: structure과 parameter 최적화 분리된 모델

Rule 길이가 각각 2, 3 일 때

Hits@10 비교

Experiment: Grid path finding

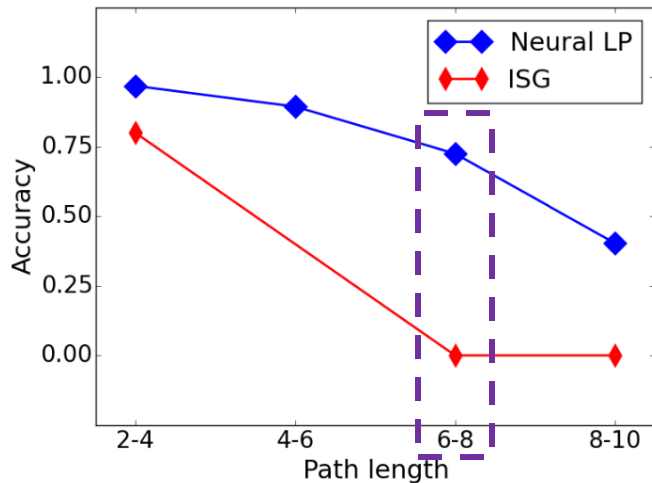


Figure 3: Accuracy on grid path finding.

16x16 Grid에서의 예측 Path 정확도 비교

Query: North_SouthWest_SouthEast_South_East ...

Rule 길이가 길어져도 robust함

Experiment: Knowledge base completion

Confidence

1.00	$\text{partially_contains}(C, A) \leftarrow \text{contains}(B, A) \wedge \text{contains}(B, C)$
0.45	$\text{partially_contains}(C, A) \leftarrow \text{contains}(A, B) \wedge \text{contains}(B, C)$
0.35	$\text{partially_contains}(C, A) \leftarrow \text{contains}(C, B) \wedge \text{contains}(B, A)$
1.00	$\text{marriage_location}(C, A) \leftarrow \text{nationality}(C, B) \wedge \text{contains}(B, A)$
0.35	$\text{marriage_location}(B, A) \leftarrow \text{nationality}(B, A)$
0.24	$\text{marriage_location}(C, A) \leftarrow \text{place_lived}(C, B) \wedge \text{contains}(B, A)$
1.00	$\text{film_edited_by}(B, A) \leftarrow \text{nominated_for}(A, B)$
0.20	$\text{film_edited_by}(C, A) \leftarrow \text{award_nominee}(B, A) \wedge \text{nominated_for}(B, C)$

HasOfficeInCounty(USA, Uber)  *HasOfficeInCounty, Uber => USA* 찾기

한 Rule에 대한 여러 시도들의 예측 과정과
가장 높은 confidence를 가진 rule에 대한 정규화된 confidence 값 도출

Experiment: Knowledge base completion

near-duplicate 관계, 역관계 제거 ver.

	WN18		FB15K		FB15KSelected	
	MRR	Hits@10	MRR	Hits@10	MRR	Hits@10
Neural Tensor Network	0.53	66.1	0.25	41.4	-	-
TransE	0.38	90.9	0.32	53.9	-	-
DISTMULT [29]	0.83	94.2	0.35	57.7	0.25	40.8
Node+LinkFeat [25]	0.94	94.3	0.82	87.0	0.23	34.7
Implicit Reasoners [23]	-	95.3	-	92.7	-	-
Neural LP	0.94	94.5	0.76	83.7	0.24	36.2

각 데이터셋에서 state-of-the-art하거나 거의 가까운 레벨의 결과를 보임

Experiment: Knowledge base completion

Table 6: Inductive knowledge base completion. The metric is Hits@10.

	WN18	FB15K	FB15KSelected
TransE	0.01	0.48	0.53
Neural LP	94.49	73.28	27.97

Training set과 Test set을 서로 다른 entity 집합으로 구성

Inductive learning에 대한 평가 진행

Experiment: Question answering

Table 7: A subset of the WIKIMOVIES dataset.

Knowledge base	<code>directed_by(Blade Runner,Ridley Scott)</code> <code>written_by(Blade Runner,Philip K. Dick)</code> <code>starred_actors(Blade Runner,Harrison Ford)</code> <code>starred_actors(Blade Runner,Sean Young)</code>
Questions	What year was the movie Blade Runner released? Who is the writer of the film Blade Runner?

제시된 Question처럼 **자연어로 된 Query**에서 정답을 잘 찾는가

- * Query를 neural controller에 대해 연속적인 input으로 처리하기 위해
 - 단어에 대해 embedding lookup table 학습 후
 - embedding의 arithmetic mean으로 query를 표현

Experiment: Question answering

Model	Accuracy
Memory Network	78.5
QA system	93.5
Key-Value Memory Network [16]	93.9
Neural LP	94.6

타 QA 모델들은 KG의 entity와 relation을 embedding하여 유사도 계산을 통해 추론

Rule 기반의 Neural LP가 더 높은 정확도를 보임

Conclusion

기존의 logical rule 활용 방식은 structure, parameter 최적화가 분리되어 있다는 문제 존재

Neural LP는 TensorLog를 기반으로한 structure과 parameter의 계산을 통해

미분이 가능한 과정을 바탕으로 Gradient 기반의 효율적인 최적화 가능하게 함

Experiment를 통해 Statistical relation learning, Rule length,
Question answering, Completion 측면에서 우수한 성능을 보임