Knowledge Graph Reasoning: Recent Advances

William Wang
Department of Computer Science

UC SANTA BARBARA

CIPS Summer School 2018
Beijing, China
Agenda

• Motivation
• Path-Based Reasoning
• Embedding-Based Reasoning
• Bridging Path-Based and Embedding-Based Reasoning: DeepPath, MINERVA, and DIVA
• Conclusions
• Other Research Activities at UCSB NLP
Knowledge Graphs are Not Complete
Benefits of Knowledge Graph

• Support various applications
  • Structured Search
  • Question Answering
  • Dialogue Systems
  • Relation Extraction
  • Summarization

• Knowledge Graphs can be constructed via information extraction from text, but…
  • There will be a lot of missing links.
  • Goal: complete the knowledge graph.
Reasoning on Knowledge Graph

**Query node:** Band of brothers

**Query relation:** tvProgramLanguage

- English

- Caesars Entertainment

- United States

- Graham Yost

- HBO

- Mini-Series

- Neal McDonough

- Tom Hanks

- Graham Yost

- writtenBy

- tvProgramCreator

- HBO

- music

- tvProgramGenre

- Michael Kamen

- serviceLocation

- countrySpoken

- personLanguages

- nationality

- countryOfOrigin

- castActor

- awardWorkWinner

- profession

- tvProgramCreator

- personLanguages

- nationality

- countryOfOrigin

- castActor
KB Reasoning Tasks

- Predicting the missing link.
  - Given e1 and e2, predict the relation r.

- Predicting the missing entity.
  - Given e1 and relation r, predict the missing entity e2.

- Fact Prediction.
  - Given a triple, predict whether it is true or false.
Related Work

• **Path-based methods**
  - Path-Ranking Algorithm, Lao et al. 2011
  - ProPPR, Wang et al, 2013 (My PhD thesis)
  - Subgraph Feature Extraction, Gardner et al, 2015
  - RNN + PRA, Neelakantan et al, 2015
  - Chains of Reasoning, Das et al, 2017

Why do we need path-based methods?
It’s accurate and explainable!
Path-Ranking Algorithm (Lao et al., 2011)

1. Run random walk with restarts to derive many paths.
2. Use supervised training to rank different paths.
ProPPR (Wang et al., 2013; 2015)

- ProPPR generalizes PRA with recursive probabilistic logic programs.
- You may use other relations to jointly infer this target relation.

:\[\begin{array}{l}
\text{about}(X, Z) :- \text{handLabeled}(X, Z) \quad \# \text{base} \\
\text{about}(X, Z) :- \text{sim}(X, Y), \text{about}(Y, Z) \quad \# \text{prop} \\
\text{sim}(X, Y) :- \text{link}(X, Y) \quad \# \text{sim,link} \\
\text{sim}(X, Y) :- \\
\quad \text{hasWord}(X, W), \text{hasWord}(Y, W), \quad \# \text{sim,word} \\
\quad \text{linkedBy}(X, Y, W) \\
\text{linkedBy}(X, Y, W) :- \text{true} \quad \# \text{by}(W)
\end{array}\]
Chain of Reasoning (Das et al, 2017)

- 1. Use PRA to derive the path.
- 2. Use RNNs to perform reasoning of the target relation.
Related Work

• **Embedding-based method**
  - RESCAL, Nickel et al, 2011
  - TransE, Bordes et al, 2013
  - Neural Tensor Network, Socher et al, 2013
  - TransR/CTransR, Lin et al, 2015
  - Complex Embeddings, Trouillon et al, 2016

Embedding methods allow us to compare, and find similar entities in the vector space.
Bridging Path-Based and Embedding-Based Reasoning with Deep Reinforcement Learning: DeepPath (Xiong et al., 2017)
RL for KB Reasoning: DeepPath (Xiong et al., 2017)

- Learning the paths with RL, instead of using random walks with restart
- Model the path finding as a MDP
- Train a RL agent to find paths
- Represent the KG with pretrained KG embeddings
- Use the learned paths as logical formulas
Machine Learning

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
Supervised v.s. Reinforcement Learning

Supervised Learning
- Training based on supervisor/label/annotation
- Feedback is instantaneous
- Not much temporal aspects

Reinforcement Learning
- Training only based on reward signal
- Feedback is delayed
- Time matters
- Agent actions affect subsequent exploration
Reinforcement Learning

• RL is a general purpose framework for decision making
  • RL is for an *agent* with the capacity to *act*
  • Each *action* influences the agent’s future *state*
  • Success is measured by a scalar *reward* signal
  • Goal: *select actions to maximize future reward*
Reinforcement Learning

Multi-layer neural nets $\psi(s_t)$

KG modeled as a MDP
DeepPath: RL for KG Reasoning
Components of MDP

• Markov decision process $< S, A, P, R >$
  • $S$: continuous states represented with embeddings
  • $A$: action space (relations)
  • $P(S_{t+1} = s' | S_t = s, A_t = a)$: transition probability
  • $R(s, a)$: reward received for each taken step

• With pretrained KG embeddings
  • $s_t = e_t \oplus (e_{target} - e_t)$
  • $A = \{r_1, r_2, \ldots, r_n\}$, all relations in the KG
Reward Functions

• Global Accuracy

\[ r_{\text{GLOBAL}} = \begin{cases} 
+1, & \text{if the path reaches } e_{\text{target}} \\
-1, & \text{otherwise} 
\end{cases} \]

• Path Efficiency

\[ r_{\text{EFFICIENCY}} = \frac{1}{\text{length}(p)} \]

• Path Diversity

\[ r_{\text{DIVERSITY}} = -\frac{1}{|F|} \sum_{i=1}^{|F|} \cos(p, p_i) \]
Training with Policy Gradient

• Monte-Carlo Policy Gradient (REINFORCE, William, 1992)

\[ \nabla_\theta J(\theta) = \sum_t \sum_{a \in A} \pi(a|s_t; \theta) \nabla_\theta \log \pi(a|s_t; \theta) R(s_t, a_t) \]
\[ \approx \nabla_\theta \sum_t \log \pi(a = r_t|s_t; \theta) R(s_t, a_t) \]

\[ R(s_t, a_t) = \lambda_1 r_{global} + \lambda_2 r_{efficiency} + \lambda_3 r_{diversity} \]
Challenge

- Typical RL problems
  - Atari games (Mnih et al., 2015): 4~18 valid actions
  - AlphaGo (Silver et al. 2016): ~250 valid actions
  - Knowledge Graph reasoning: $\geq 400$ actions

Issue:
- large action (search) space $\rightarrow$ poor convergence properties
Supervised (Imitation) Policy Learning

- Use randomized BFS to retrieve a few paths
- Do imitation learning using the retrieved paths
- All the paths are assigned with +1 reward

\[
\nabla_\theta J(\theta) = \sum_t \sum_{a \in A} \pi(a|s_t; \theta) \nabla_\theta \log \pi(a|s_t; \theta)
\]

\[
\approx \nabla_\theta \sum_t \log \pi(a = r_t|s_t; \theta)
\]
Datasets and Preprocessing

<table>
<thead>
<tr>
<th>Dataset</th>
<th># of Entities</th>
<th># of Relations</th>
<th># of Triples</th>
<th># of Tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>FB15k-237</td>
<td>14,505</td>
<td>237</td>
<td>310,116</td>
<td>20</td>
</tr>
<tr>
<td>NELL-995</td>
<td>75,492</td>
<td>200</td>
<td>154,213</td>
<td>12</td>
</tr>
</tbody>
</table>

**FB15k-237**: Sampled from FB15k (Bordes et al., 2013), redundant relations removes

**NELL-995**: Sampled from the 995\textsuperscript{th} iteration of NELL system (Carlson et al., 2010b)

- **Dataset processing**
  - Remove useless relations: `haswikipediaurl`, `generalizations`, etc
  - Add inverse relation links to the knowledge graph
  - Remove the triples with task relations
Effect of Supervised Policy Learning

- **x-axis**: number of training epochs
- **y-axis**: success ratio (probability of reaching the target) on test set

-> Re-train the agent using reward functions
Inference Using Learned Paths

- Path as logical formula
  - **FilmCountry**: actionFilm\(^{-1}\) \(\rightarrow\) person\(\text{Nationality}\)
  - **PersonNationality**: placeOfBirth \(\rightarrow\) location\(\text{Contains}^{-1}\)
  - etc …

- Bi-directional path-constrained search
  - Check whether the formulas hold for entity pairs

Uni-directional search  bi-directional search
Link Prediction Result

<table>
<thead>
<tr>
<th>Tasks</th>
<th>PRA</th>
<th>Ours</th>
<th>TransE</th>
<th>TransR</th>
</tr>
</thead>
<tbody>
<tr>
<td>worksFor</td>
<td>0.681</td>
<td>0.711</td>
<td>0.677</td>
<td>0.692</td>
</tr>
<tr>
<td>athletePlaysForTeam</td>
<td><strong>0.987</strong></td>
<td>0.955</td>
<td>0.896</td>
<td>0.784</td>
</tr>
<tr>
<td>athletePlaysInLeague</td>
<td>0.841</td>
<td><strong>0.960</strong></td>
<td>0.773</td>
<td>0.912</td>
</tr>
<tr>
<td>athleteHomeStadium</td>
<td>0.859</td>
<td><strong>0.890</strong></td>
<td>0.718</td>
<td>0.722</td>
</tr>
<tr>
<td>teamPlaysSports</td>
<td>0.791</td>
<td>0.738</td>
<td>0.761</td>
<td><strong>0.814</strong></td>
</tr>
<tr>
<td>orgHirePerson</td>
<td>0.599</td>
<td><strong>0.742</strong></td>
<td>0.719</td>
<td>0.737</td>
</tr>
<tr>
<td>personLeadsOrg</td>
<td>0.700</td>
<td><strong>0.795</strong></td>
<td>0.751</td>
<td>0.772</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Overall</td>
<td>0.675</td>
<td><strong>0.796</strong></td>
<td>0.737</td>
<td>0.789</td>
</tr>
</tbody>
</table>

Mean average precision on NELL-995
Qualitative Analysis

Path length distributions

![Graph showing path length distributions](image)
Qualitative Analysis

Example Paths

**personNationality:**
- placeOfBirth -> locationContains\(^{-1}\)
- peoplePlaceLived -> locationContains\(^{-1}\)
- peopleMariage -> locationOfCeremony -> locationContains\(^{-1}\)

**tvProgramLanguage:**
- tvCountryOfOrigin -> countryOfficialLanguage
- tvCountryOfOrigin -> filmReleaseRegion\(-1\) -> filmLanguage
- tvCastActor -> personLanguage

**athletePlaysForTeam:**
- athleteHomeStadium -> teamHomeStadium\(^{-1}\)
- athletePlaysSports -> teamPlaysSports\(^{-1}\)
- athleteLedSportsTeam
Bridging Path-Finding and Reasoning w. Variational Inference (teaser): DIVA (Chen et al., NAACL 2018)
DIVA: Variational KB Reasoning (Chen et al., NAACL 2018)

- Inferring latent paths connecting entity nodes.

\[ p(r|e_s, e_d) \]

**Condition** \((e_s, e_d)\)

**Observed Variable** \(r\)

\[ \bar{p} = \text{argmax}_p \log p(r|e_s, e_d) \]
DIVA: Variational KB Reasoning (Chen et al., NAACL 2018)

- Inferring latent paths connecting entity nodes by parameterizing likelihood (path reasoning) and prior (path finding) with neural network modules.

\[
p = \arg\max_p p(r|e_s, e_d) = \arg\max_p \log \int_L p(r|L) p(L|e_s, e_d)
\]
DIVA: Variational KB Reasoning (Chen et al., NAACL 2018)

- Marginal likelihood \( \log \int_L p(r|L)p(L|e_s, e_d) \) is intractable
- We resort to Variational Bayes by introduce a posterior distribution \( q(L|e_s, e_d, r) \)

\[
\log p(r|e_s, e_d) \geq \text{ELBO} \geq \mathbb{E}_{q(L|e_s, e_d, r)}[\log p(r|L)] - KL(q(L|e_s, e_d, r)||p(L|e_s, e_d))
\]
Parameterization – Path-finder

- Approximate posterior $q_\varphi (L|e_s, e_d, r)$ and prior $p_\beta (L|e_s, e_d)$: parameterize with RNN

Transition Probability: $p(a_{\tau+1}, e_{\tau+1}|a_{1:\tau}, e_{1:\tau})$
Parameterization – Path Reasoner

- Likelihood $p_\theta (r|L)$: parameterize with CNN
DIVA: Variational KB Reasoning
(Chen et al., NAACL 2018)

- Training

\[
\mathbb{E}_{q_{\varphi}(L|e_s, e_d, r)} \left[ \log p_{\theta}(r|L) \right]
\]

\[
KL(q_{\varphi}(L|e_s, e_d, r) || p_{\beta}(L|e_s, e_d))
\]

posteriors: \(q_{\varphi}\), likelihood: \(p_{\theta}(r|L)\), prior: \(p_{\beta}(L|e_s, e_d)\)
DIVA: Variational KB Reasoning (Chen et al., NAACL 2018)

• Testing

posterior: $q_\phi$, likelihood: $p_\theta (r|L)$, prior: $p_\beta (L|e_s, e_d)$
Conclusions

• Embedding-based methods are very scalable and robust.
• Path-based methods are more interpretable.
• There are some recent efforts in unifying embedding and path-based approaches.
• DIVA integrates path-finding and reasoning in a principled variational inference framework.
• Natural Language Processing
  • Information Extraction: relation extraction, and distant supervision.
  • Summarization: abstractive summarization.
  • Social Media: non-standard English expressions.
  • Language & Vision: action/relation detection, and video captioning.
  • Spoken Language Processing: task-oriented neural dialogue systems.

• Machine Learning
  • Statistical Relational Learning: neural symbolic reasoning.
  • Deep Learning: sequence-to-sequence models.
  • Structure Learning: learning the structures for neural models.
  • Reinforcement Learning: efficient and effective methods for DRL and NLP.

• Artificial Intelligence
  • Knowledge Representation & Reasoning: beyond Freebase/OpenIE.
  • Knowledge Graphs: construction, completion, and reasoning.
Other Research Activities at UCSB’s NLP Group
Natural Language Generation
Reinforced Conditional Variational Autoencoder for Generating Emotional Sentences (Zhou and Wang, ACL 2018)

https://arxiv.org/abs/1711.04090

Omg you totally just made my day 😊

Replying to @user...
i want to be like u u are so confident a just beautiful and perfect in every way
### Controlling Emotions for RC-VAE Generated Sentences

<table>
<thead>
<tr>
<th>User’s Input</th>
<th>sorry guys, was gunna stream tonight but i ’m still feeling sick</th>
</tr>
</thead>
<tbody>
<tr>
<td>Designated Emojis</td>
<td>![emoji] ![emoji]</td>
</tr>
<tr>
<td>Generated by Seq2Seq Baseline</td>
<td>i ’m sorry you ’re going to be missed it i ’m sorry for your loss</td>
</tr>
<tr>
<td>Generated By MojiTalk</td>
<td>hope you are okay hun! hi jason, i ’ll be praying for you</td>
</tr>
</tbody>
</table>
Hierarchical Deep Reinforcement Learning for Video Captioning (Wang et al., CVPR 2018)

Caption #1: A woman offers her dog some food.
Caption #2: A woman is eating and sharing food with her dog.
Caption #3: A woman is sharing a snack with a dog.

Caption: A person sits on a bed and puts a laptop into a bag. The person stands up, puts the bag on one shoulder, and walks out of the room.
Deep Multimodal Video Captioning (Wang et al., NAACL 2018)

Ground Truth: A girl is singing.
A girl sings to a song.

Video Only: A woman is talking in a room.

Video + Audio: A girl is singing a song.
Adversarial Reward Learning for Visual Storytelling (Wang et al., ACL 2018)

<table>
<thead>
<tr>
<th>XE-ss</th>
<th>We took a trip to the mountains.</th>
<th>There were many different kinds of different kinds.</th>
<th>We had a great time.</th>
<th>He was a great time.</th>
<th>It was a beautiful day.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AREL</td>
<td>The family decided to take a trip to the countryside.</td>
<td>There were so many different kinds of things to see.</td>
<td>The family decided to go on a hike.</td>
<td>I had a great time.</td>
<td>At the end of the day, we were able to take a picture of the beautiful scenery.</td>
</tr>
<tr>
<td>Human-created Story</td>
<td>We went on a hike yesterday.</td>
<td>There were a lot of strange plants there.</td>
<td>I had a great time.</td>
<td>We drank a lot of water while we were hiking.</td>
<td>The view was spectacular.</td>
</tr>
</tbody>
</table>

Figure 6: Qualitative comparison example with XE-ss. The direct comparison votes (AREL:XЕ-ss:Tie) were 5:0:0 on Relevance, 4:0:1 on Expressiveness, and 5:0:0 on Concreteness.
Computational Social Science
Learning to Generate Slang Explanations (Ke Ni, IJCNLP 2017)

Walk it off: to stand up like a man and forget about it, or deal with it, like a man.

Depression just doesn't go away. You can't just walk it off. No I don't think I'll be alright. No I'm not fine. No it's not okay.
# Automatic Generation of Slang Words

(Kulkarni and Wang, NAACL 2018)

<table>
<thead>
<tr>
<th>Word</th>
<th>Derived From</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>dink</td>
<td>double income no kids</td>
<td>alphabetism</td>
</tr>
<tr>
<td>lambortini</td>
<td>lamborghini + martini</td>
<td>blend</td>
</tr>
<tr>
<td>diamat</td>
<td>dialectical + materialism</td>
<td>blend</td>
</tr>
<tr>
<td>tude</td>
<td>attitude</td>
<td>clipping (fore)</td>
</tr>
<tr>
<td>brill</td>
<td>brilliant</td>
<td>clipping (back)</td>
</tr>
<tr>
<td>teenie-weenie</td>
<td>teenie</td>
<td>reduplicative</td>
</tr>
<tr>
<td>yik-yak</td>
<td>yik</td>
<td>reduplicative</td>
</tr>
</tbody>
</table>
Leveraging Intra-Speaker and Inter-Speaker Representation Learning for Hate Speech Detection (Qian et al., NAACL 2018)
Deep Reinforcement Learning
Scheduled Policy Optimization (Xiong et al., IJCAI 2018)

Model Architecture
- Conv Layers
- Embedding Layers
- Previous Action
- RNN Layers

Training Algorithm
- Action Message
- Command
- Reward
- Block ID
- Environment
- Optimization Scheduler
- PPO Update
- LfD Update

Place the Pepsi block so that its top left corner touches the lower right corner of the McDonalds block.
Combining Model-Free and Model-Based Deep Reinforcement Learning (Wang et al., ECCV 2018)

Fig. 2: The overview of our method.
Structure Learning
Learning Activation Functions in Deep Neural Networks (Conner Vercellino, NIPS MetaLearning)
Acknowledgment

Sponsors: Adobe, Amazon, ByteDance, DARPA, Facebook, Google, IBM, LogMeIn, NVIDIA, Tencent.
Thanks! Questions?

nlp.cs.ucsb.edu

DeepPath Source code:
https://github.com/xwhan/DeepPath

KBGAN Source code:
https://github.com/cai-lw/KBGAN

Scheduled Policy Optimization:
https://github.com/xwhan/walk_the_blocks

ProPPR Source code:
https://github.com/TeamCohen/ProPPR

AREL Source code:
https://github.com/littlekobe/AREL