Relaxing Unanswerable Geographic Questions Using A Spatially Explicit Knowledge Graph Embedding Model

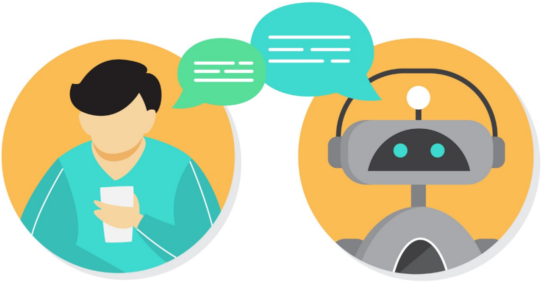
Gengchen Mai, Bo Yan, Krzysztof Janowicz, **Rui Zhu**

STKO Lab, Department of Geography
University of California, Santa Barbara

AGILE 2019: June 20, Limassol, Cyprus

Question Answering

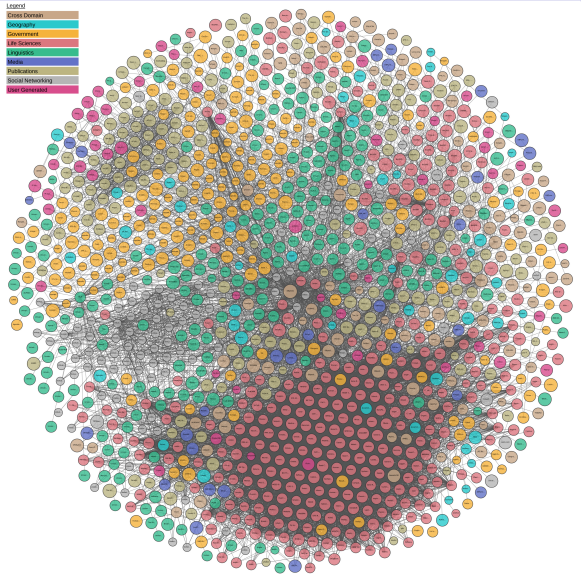
- **Question answering (QA)** refers to the methods, process, and systems which allow users to ask questions in the form of natural language sentences and receive one or more answers, often in the form of sentences.



QA Using Knowledge Graph

- "Knowledge graphs are **large networks** of entities, their semantic types, properties, and relationships between them" (M. Kroetsch and G. Weikum, 2016)
- The graph structure provides **rich contexts** for entities in a knowledge graph
- Most **state-of-the-art QA systems** are using knowledge graphs
  - E.g., Google Assistant, Apple Siri, Amazon Alexa

The Linked Data Knowledge Graph



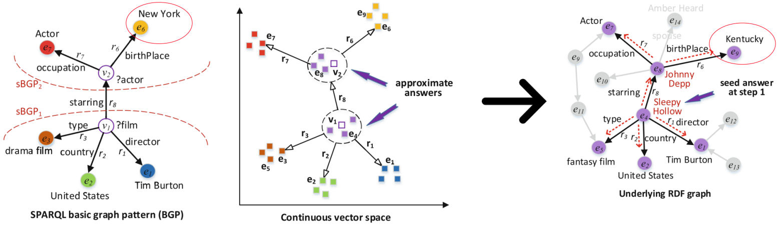
The Linked Open Data Cloud from lod-cloud.net

Unanswerable Questions

- Due to **missing information** and **logical inconsistency**, it is likely to receive **no answer** for questions given a knowledge graph
- This challenge is commonly handeled by **query relaxation/rewriting** based on **knowledge graph embedding**
- Examples:
  - What is the weather like in Agios Athanasios? (missing information)
    - After **relaxation**: What is the weather like in Limassol?
  - Which city spans Texas and Colorado? (logical inconsistency)
    - After **rewriting**: Which city locates in Texas?

Query Relaxation Based on Knowledge Graph Embeddings

- What is the American drama films directed by Tim Burton, one of whose star actors was born in New York?

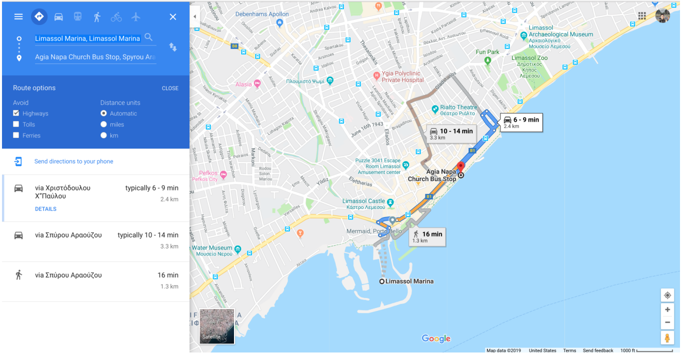


M. Wang et al., 2018

- Note: each **(head(h), relation(r), tail(t))** in the graph is a **triple**

Geographic Question Answering

- **Geographic question answering** refers to those questions that involve *geographic information*
  - Example: How long will it take to travel from Limassol Marina to the Ayia Napa church?



Spatial Is Special

Geographic question answering is **fundamentally different** from general question answering:

- Context-dependent
  - Find the nightclubs **near me** that is **open now** and is **18+**.
- Spatial operations
  - What is the **shortest path** from the Pefkos City hotel to Tassos Papadopoulos building of CUT?
- Vagueness and uncertainty
  - How many **lakes** are there in Cyprus?

Spatial Is Special

Even more fundamental research questions:

- How could we incorporate spatial information into the question answering systems?
- Will such spatial information help to improve the geographic question answering?

Workflow

Question

↓

Query

↓

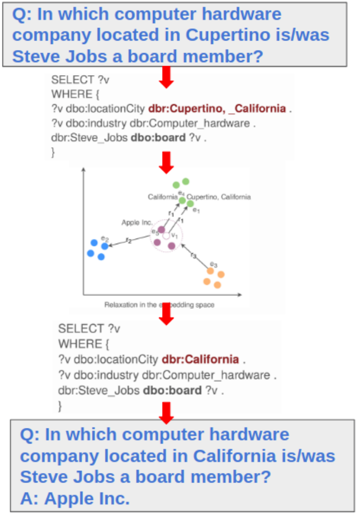
Query relaxation / rewriting

↓

Relaxed query

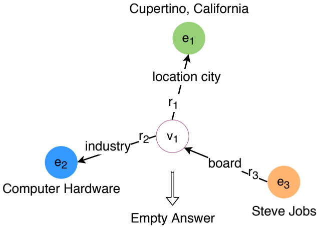
↓

Relaxed Question + Answer

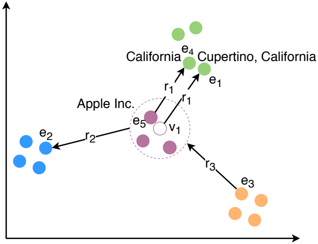


Core Ideas

- Transform the knowledge graph entities into an **embedding space** considering:
  - Graph structures
  - Domain knowledge (e.g., spatial information)
- **Relax/rewrite** the unanswerable query based on the similarity of entity embedding
- Example: In which computer hardware company located in Cupertino is/was Steve Jobs a board member?



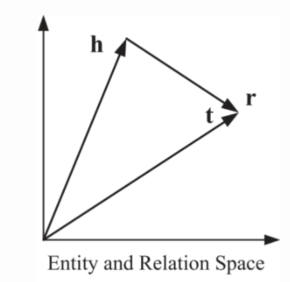
Incomplete knowledge graph yields an empty answer



Relaxation in the embedding space

Knowledge Graph Embedding

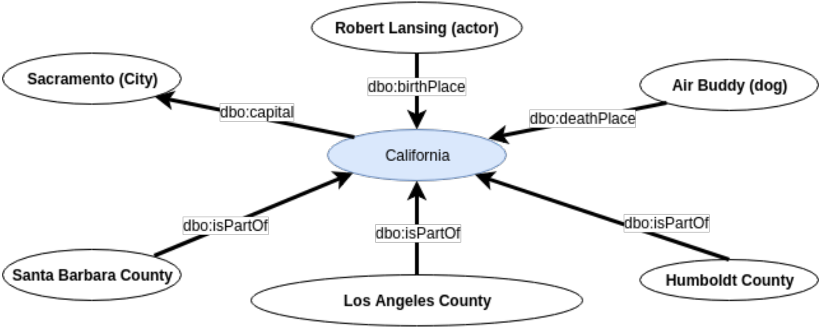
- **TransE**: a **translation-based** KG embedding model
- Entities are embedded into a **low-dimensional vector space**, while relations are treated as **translation operations in the same space**



- In a perfect situation, if $(h, r, t) \in G$, $\| \mathbf{h} + \mathbf{r} - \mathbf{t} \| = 0$
- Example: $(Limassol, is\_located, Cyprus) \in G$ $\| \mathbf{limassol} + \mathbf{is\_located} - \mathbf{cyprus} \| = 0$

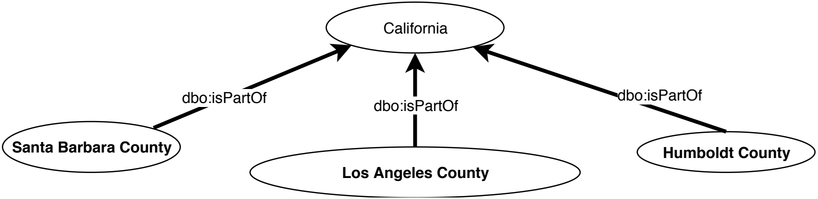
Contexts in Knowledge Graph

- **Entity context modeling**: all 1 degree neighbors of the target entity are considered **equally** as its context



Limitation of State-of-the-Art

- **Distance effect** is not considered
  - Example: Santa Barbara County is closer to Los Angeles County compared to Humboldt County



Spatially-explicit Knowledge Graph Embedding

- **TransGeo**: to assign **larger weights** to geographical triples in an entity context, and these weights are modeled using a **distance decay function**

Learning Weights for Triples

- Given a KG $G = \langle E, R \rangle$, a set of geographic entities $P \subseteq E$, and a triple $T_i = (h_i, r_i, t_i) \in G$

$$w(T_i) = \begin{cases} \max(\ln \frac{D}{dis(h_i,t_i)+\varepsilon}, l) & \text{if } h_i \in P \wedge t_i \in P \\ l & \text{otherwise} \end{cases}$$

  - $dis(h_i, t_i)$ is the geodesic distance between geographic entity h_i and t_i
  - l is the lowest edge weight we allow for each triple
  - D is the longest (simplified) earth surface distance
  - ε is a hyperparameter

Learning Weights for Entities

- The weight for each entity in the **edge-weighted** knowledge graph is modeled as:

$$w(e_i) = N \cdot \frac{\frac{1}{-\ln PR(e_i)}}{\sum_i \frac{1}{-\ln PR(e_i)}}$$

  - $PR(e_i)$ is the **edge weighted PageRank score** for each entity e_i, which is computed using the **weights of triples**
  - $PR(e_i)$ represents the **probability of a random walker to arrive at entity** e_i after n time steps
  - N is the number of entities in G
- $w(e_i)$ encodes the **structural information of the KG** and the **spatial information among geographic entities**.

TransGeo: Context Sampling & Loss Function

- For each entity e_i in G, we sample an entity context $C_{samp}(e_i) \subseteq C(e_i)$ where the **sampling probability** $P(r_{ci}, e_{ci})$ of each context item $(r_{ci}, e_{ci}) \in C(e_i)$ is based on the **entity weight** $w(e_{ci})$

$$P(r_{ci}, e_{ci}) = \frac{w(e_{ci})}{\sum_{(r_{cj}, e_{cj}) \in C(e_i)} w(e_{cj})}$$

$$\text{where } (e_i, r_{ci}, e_{ci}) \in G \vee (e_{ci}, r_{ci}, e_i) \in G$$

- An **incompatibility score** between $C_{samp}(e_i)$ and an arbitrary entity e_k can be computed as:

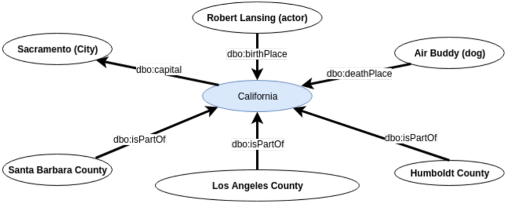
$$f(e_k, C_{samp}(e_i)) = \frac{1}{|C_{samp}(e_i)|} \cdot \sum_{(r_{cj}, e_{cj}) \in C_{samp}(e_i)} \phi(e_k, r_{cj}, e_{cj})$$

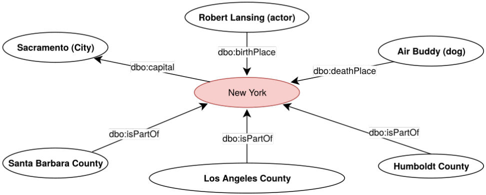
$$\phi(e_k, r_{cj}, e_{cj}) = \begin{cases} \|\mathbf{e_k} + \mathbf{r_{cj}} - \mathbf{e_{cj}}\| & \text{if } (e_i, r_{cj}, e_{cj}) \in G \\ \|\mathbf{e_{cj}} + \mathbf{r_{cj}} - \mathbf{e_k}\| & \text{if } (e_{cj}, r_{cj}, e_i) \in G \end{cases}$$

- **Pairwise ranking loss function**:

$$\mathcal{L} = \sum_{e_i \in G} \sum_{e_i' \in Neg(e_i)} max\Big(\gamma + f(e_i, C_{samp}(e_i)) - f(e_i', C_{samp}(e_i)), 0\Big)$$

Interpretation of the Incompatibility Score



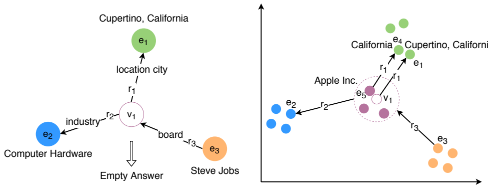


TransGeo

- After training the neural network using the proposed **context sampling** and **loss function**, we have the vector-based embedding for each entity
- This learned embedding encodes both the **graph structural information** and **spatial information**
- We then use the learned embedding to **relax/rewrite queries**

Query Relaxation/Rewriting

Recall: In which computer hardware company located in Cupertino is/was Steve Jobs a board member?



Incomplete knowledge graph yields an empty answer

Relaxation in the embedding space

- Use $\mathbf{v_i} = \mathbf{e_i} + \mathbf{r_i}$ to **predict** variable embedding $\mathbf{v_i}$ from each triple path
- Compute the final variable embedding $\mathbf{v}$ as **weighted average** of $\mathbf{v_i}$
- Use **nearest neighbor search** in entity embedding space to get the approximate answer
- Use the **approximate answer** to relax/rewrite the original

DB18 for Training

- We collected a new KG embedding training dataset, *DB18*[1], which is a subgraph of DBpedia.

Table: Summary statistic for *DB18*

DB18	Total	Training	Testing
# of triples	139155	138155	1000
# of entities	22061	-	-
# of relations	281	-	-
# of geographic entities	1681 (7.62%)	-	-

[1] `https://github.com/gengchenmai/TransGeo`

GeoUQ for Evaluation

- We constructed an evaluation dataset, GeoUQ, which is composed of **20 unanswerable geographic questions** based on *DB18*.
- These queries satisfy 2 conditions:
  - each query yields **empty answer set** when executing it on the **training KG**;
  - each query returns **only one answer** when executing it on the **whole KG**

Evaluation

- **Link prediction**: Given h, r, to predict the correct t
- **Answer prediction by relaxation/rewriting**: The rank of the correct answer in the queried answer ranking list

Table: Two evaluation tasks for different KG embedding models

	Link Prediction				Query Relaxation	
	MRR		HIT@10		MRR	HIT@10
	Raw	Filter	Raw	Filter		
TransE Model	**0.122**	0.149	30.00%	34.00%	0.008	5% (1 out of 20)
Wang et al. (2018)	0.113	0.154	27.20%	30.50%	0.000	0% (0 out of 20)
$TransGeo_{regular}$	0.094	0.129	28.50%	33.40%	0.098	25% (5 out of 20)
$TransGeo_{unweighted}$	0.108	0.152	30.80%	37.80%	0.043	15% (3 out of 20)
TransGeo	0.104	**0.159**	**32.40%**	**42.10%**	**0.109**	**30% (6 out of 20)**

Example

Recall: In which computer hardware company located in Cupertino is/was Steve Jobs a board member?

Original SPARQL Query:

Query:

```
SELECT ?v
WHERE {
?v dbo:locationCity dbr:Cupertino,_California .
?v dbo:industry dbr:Computer_hardware .
dbr:Steve_Jobs dbo:board ?v .
}
```

Answer: dbr:Apple_Inc

Relaxed Query by TransGeo:

Query:

```
SELECT ?v
WHERE {
?v dbo:locationCity dbr:California .
?v dbo:industry dbr:Computer_hardware .
dbr:Steve_Jobs dbo:board ?v .
}
```

Answer: dbr:Apple_Inc

Conclusion

- We propose a **spatially explicit KG embedding models, TransGeo**, which incorporates the **spatial information** into the KG embedding
- We show the use of TransGeo to **relax/rewrite geographic queries**
- Our spatially-explicit model outperforms other baseline models in both **link prediction** and **query relaxation/rewriting**
- Our code and collected data are **open sourced** for other researchers to **reproduce** our experiments

Future Work

- In the future, **complex geometries** and **topology** will be considered
- We will explore ways to design an **end-to-end** model for query answering prediction
- More **complex spatial interactions** other than distance decay can be incorporated into the model
- We plan to investigate the encoding of **temporal information** into KG embedding models