Graph-Based Reasoning over Heterogeneous External Knowledge for Commonsense Question Answering

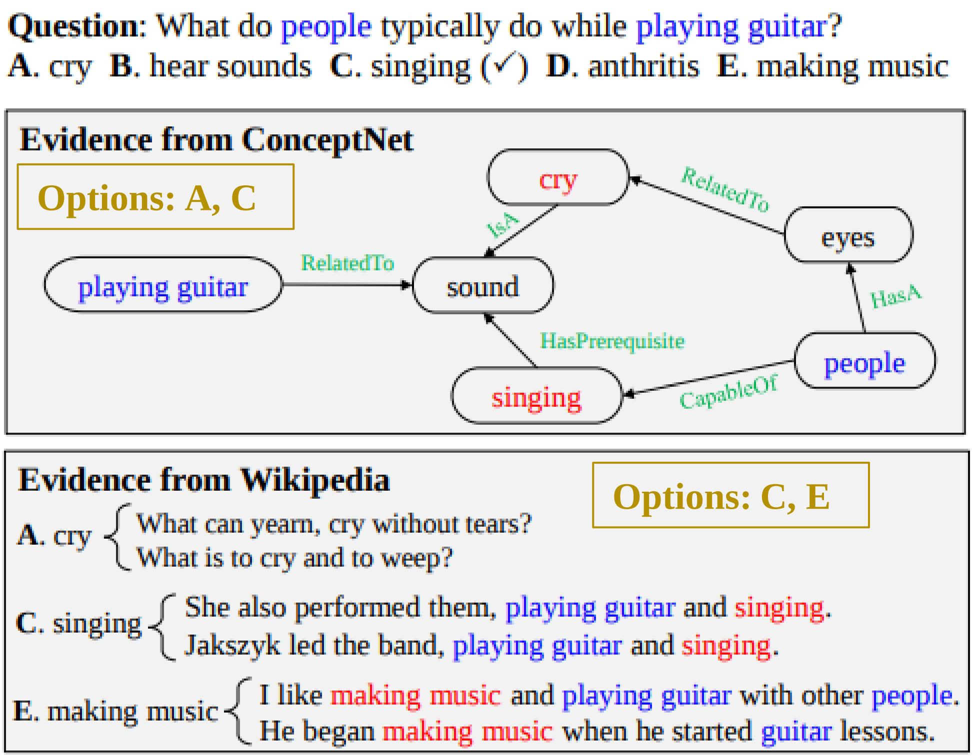
Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Songlin Hu

Published: AAAI 2020

Presenter: Anushree Hede

April 20, 2020

Motivation



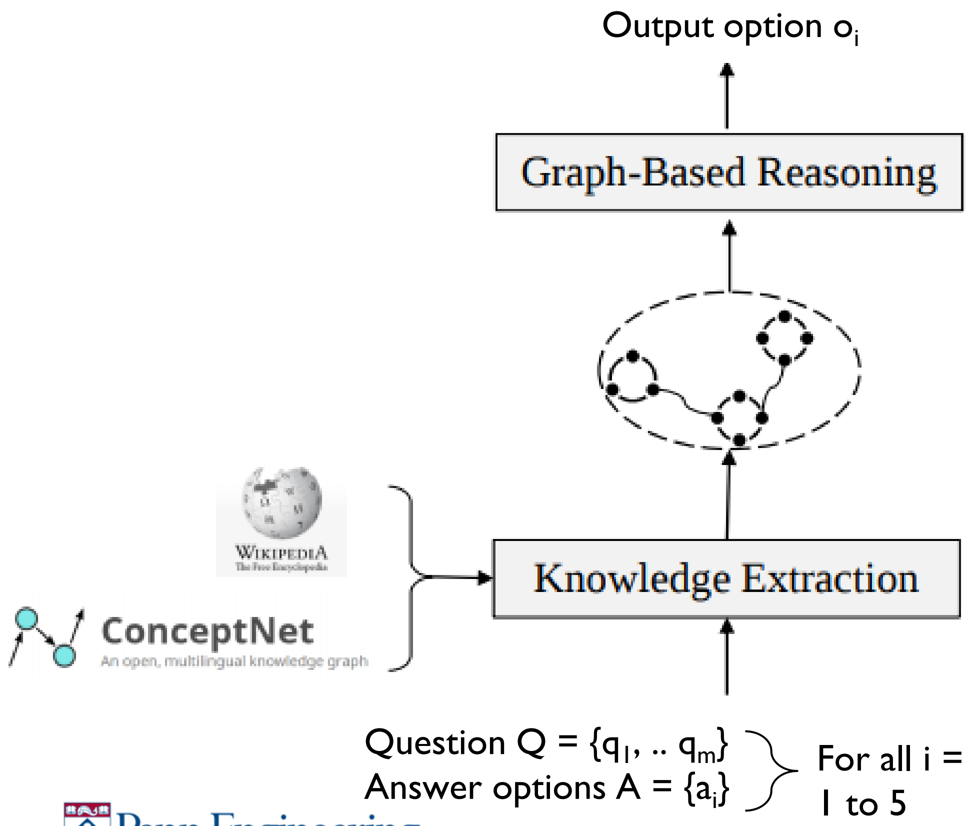
- Combining evidence from ConceptNet & Wikipedia gives the option C
- Commonsense QA
  - Collect background knowledge and reason over it
- Structured KBs: relations beneficial for reasoning
  - But low coverage is an issue
- Unstructured text: abundant coverage

Contributions

- Main: Combine heterogenous knowledge sources together into the same representation space
- Graph modules to leverage structure for reasoning
  - Context representation learning module
  - Inference module
- New state-of-the-art performance: 75.3%

Contents

Problem: Overview

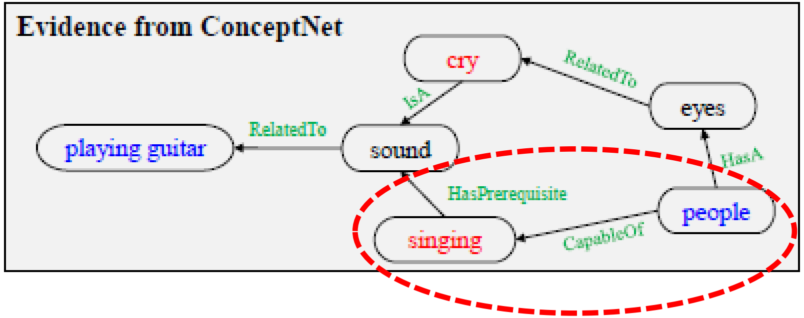


- Dataset:
  - CommonsenseQA [1]
  - Questions lack evidence, rely on background knowledge
- Evaluation:
  - Accuracy
  - Ablation Study
  - Error Analysis

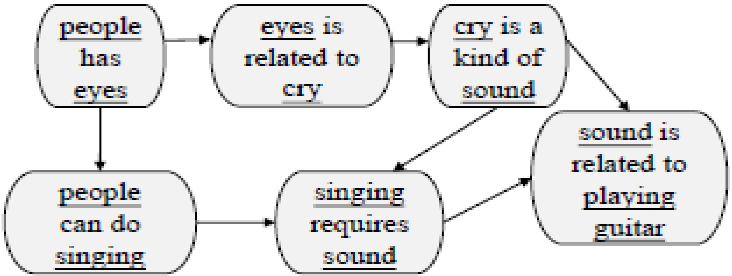
Knowledge Extraction ConceptNet → Concept-Graph

- Commonsense Knowledge Base
- Locate and search for path from question entities → answer choice entities (< 3 hops)
- Merge triples as nodes in graph
  - Edge from s_i to s_j if they contain same entity
- Convert triples to natural language sentences

Question: What do people typically do while playing guitar?
A. cry B. hear sounds C. singing (✓) D. anthritis E. making music



ConceptNet triple



Knowledge Extraction Wikipedia → Wiki-Graph

- Top 10 Wiki sentences from Elastic Search for (question + choices)
- Semantic Role Labeling: Nodes are **subject, predicate, object**
- Edges:
  - (subject, predicate)
  - (predicate, object)
  - Node A is contained in node B and the #words (A) > 3 - - - - - - ▸
  - Node A and node B only have one different word and #words(A) and #words(B) > 3

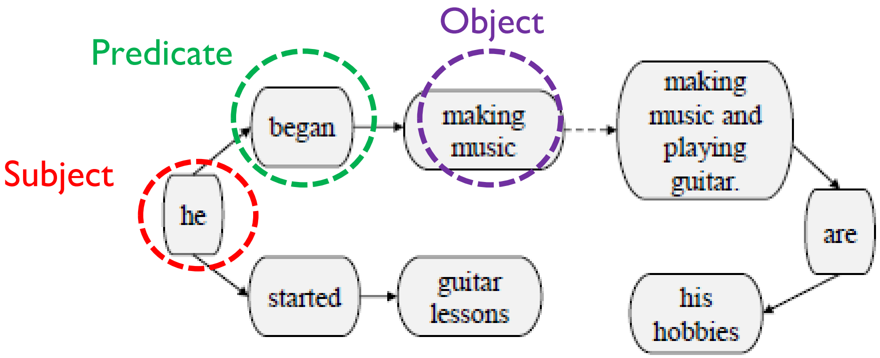
Question: What do people typically do while playing guitar?
A. cry **B**. hear sounds **C**. singing (✓) **D**. anthritis **E**. making music

Evidence from Wikipedia

A. cry { What can yearn, cry without tears? / What is to cry and to weep?

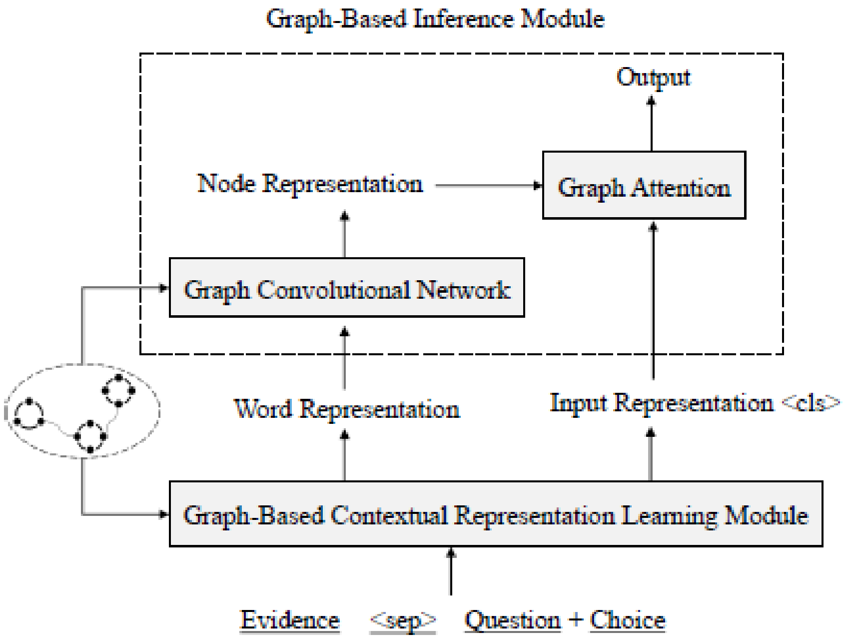
C. singing { She also performed them, playing guitar and singing. / Jakszyk led the band, playing guitar and singing.

E. making music { Making music and playing guitar are his hobbies. / He began making music when he started guitar lessons.

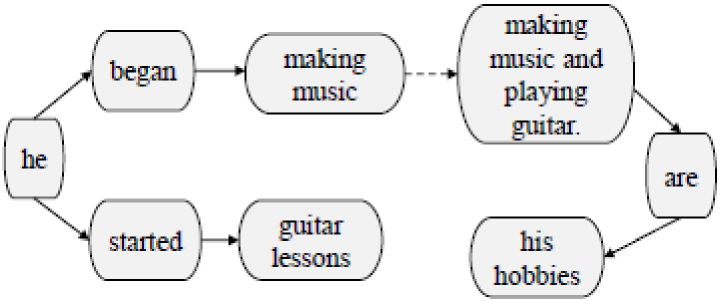


Graph-Based Reasoning

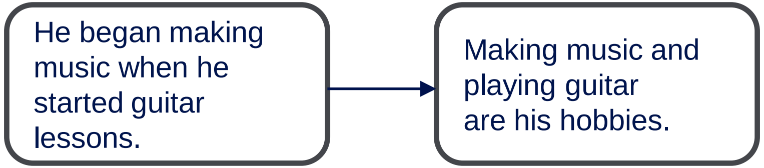
- Evidence
  - Concept-Graph
  - Wiki-Graph
- Context Representation Learning
- Inference
  - Graph Convolutional Network
  - Graph Attention
- Output



Contextual Representation Learning Module



Wiki-Graph



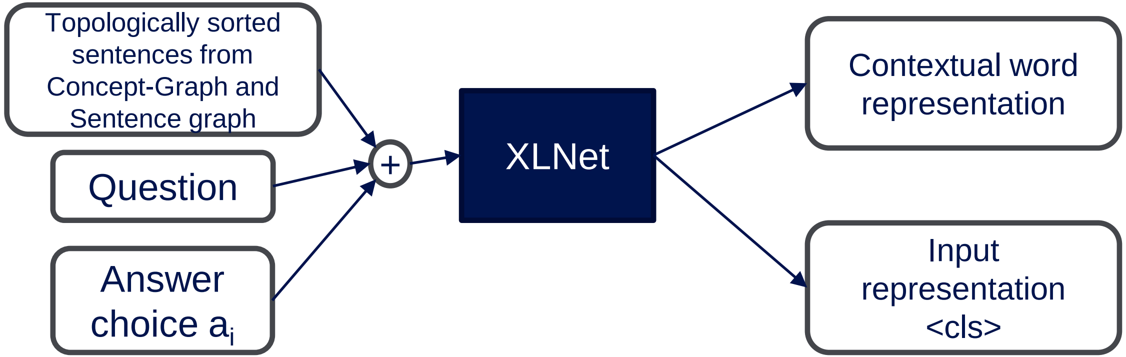
Sentence Graph

- If $p \in s_i$, $q \in s_j$ and (p,q) is an edge in Wiki-Graph, then (s_i, s_j) in an edge in sentence

- Topological sort on Concept-Graph & sentence graph
- Goal: Shorten distance between semantically similar nodes

Contextual Representation Learning Module

- XLNet: captures long term dependencies



- Goal:
  - Obtain better contextual word representations
  - Fuse two knowledge sources in same representation space

Topology Sort Algorithm (for reference)

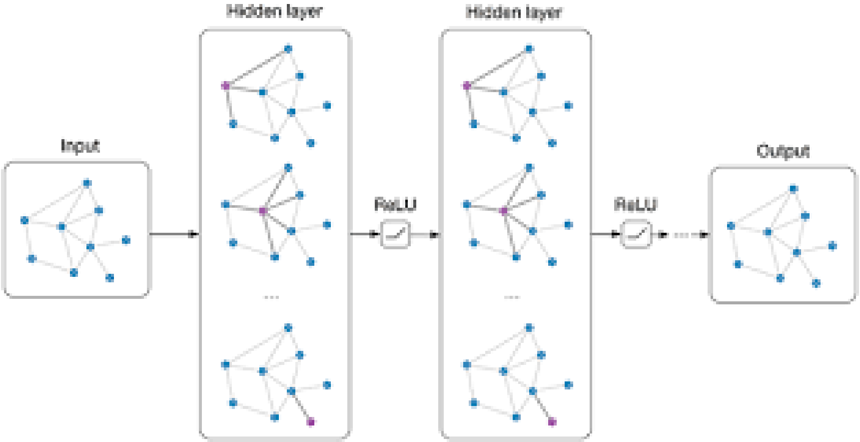
Algorithm 1 Topology Sort Algorithm.

Require: A sequence of nodes $S = \{s_i, s_2, \cdots, s_n\}$; A set of relations $R = \{r_1, r_2, \cdots, r_m\}$.

function DFS(node, visited, sorted_sequence)
    for each child s_c in node's children **do**
        if s_c has no incident edges and visited[s_c]==0 **then**
            visited[s_c]=1
            sorted_sequence.append(0, s_c)
            Remove the incident edges of s_c
            DFS(s_c, visited, sorted_sequence)
        end if
    end for
end function
sorted_sequence = []
visited = [0 for i in range(n)]
S,R = to_acyclic_graph(S,R)
for each node s_i in S **do**
    if s_i has no incident edges and visited[i] == 0 **then**
        visited[i] = 1
        sorted_sequence.append(s_i)
        DFS(s_i, visited, sorted_sequence)
    end if
end for
return sorted_sequence

Inference Module

- Graph Convolutional Networks (GCNs)
  - Use Concept-Graph and Wiki-Graph
  - Update graph node representations using features of neighboring nodes
- The i^{th} node representation in layer 0



$$h_i^0 = \sigma(W \sum_{w_j \in s_i} \frac{1}{|s_i|} h_{w_j}) , \quad (1)$$

XLNet representation for token w_j

Evidence sentence

- Subsequent layers

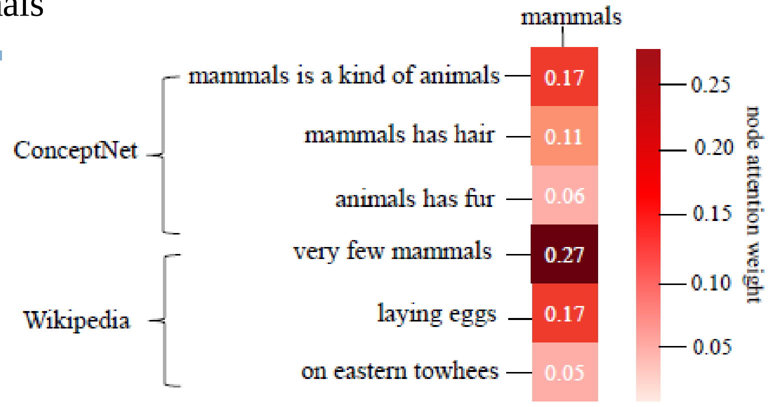
$$z_i^l = \sum_{j \in N_i} \frac{1}{|N_i|} V^l h_j^l , \quad (2)$$

Aggregated neighbor information for i^{th} node at layer l

$$h_i^{l+1} = \sigma(W^l h_i^l + z_i^l) . \quad (3)$$

Inference Module

Q: Animals who have hair and don't lay eggs are what?
A: Mammals



- Graph Attention (multiplicative)
  - Attention function: alignment score between <cls> and final GCN representation of i^{th} node
  - Aggregate over all nodes of graph
  - Obtain normalized score, compare across options

input representation <cls>

i^{th} node representation in last layer of GCN

Importance of node i

$$\alpha_i = \frac{h^c \sigma(W_1 h_i^L)}{\sum_{j \in N} h^c \sigma(W_1 h_j^L)}, \quad (4)$$

Graph representation

$$h^g = \sum_{j \in N} \alpha_j^L h_j^L . \quad (5)$$

Normalized scoring

$$score(q,a) = MLP(h_g . h_c)$$

$$p(q,a) = \frac{e^{score(q,a)}}{\sum_{a' \in A} e^{score(q,a')}} .$$

Correct option = $\operatorname{argmax}_{a \in A} p(q,a)$

Experiments

Models without descriptions

Models without extracted knowledge

Models without extracted structured knowledge

Models without extracted unstructured knowledge

Group	Model	Dev Acc	Test Acc
Group 1	SGN-lite	-	57.1
	BECON (single)	-	57.9
	BECON (ensemble)	-	59.6
	CSR-KG	-	61.8
	CSR-KG (AI2 IR)	-	65.3
Group 2	BERT-large	-	56.7
	XLNet-large	-	62.9
	RoBERTa(single)	78.5	72.1
	RoBERTa(ensemble)	-	72.5
Group 3	KagNet	-	58.9
	BERT + AMS	-	62.2
	RoBERTa + CSPT	76.2	69.6
Group 4	Cos-E	-	58.2
	BERT + OMCS	68.8	62.5
	HyKAS	-	62.5
	AristoBERTv7	-	64.6
	DREAM	73.0	66.9
	RoBERT + KE	77.5	68.4
	RoBERTa + CSPT	76.2	69.6
	RoBERTa + IR	78.9	72.1
	Our Model	**79.3**	**75.3**

Ablation Studies

Components of graph-based reasoning

Model	Dev Acc
XLNet + E	75.8
XLNet + E + Topology Sort	77.7
XLNet + E + Graph Inference	77.2
XLNet + E + Topology Sort + Graph Inference	**79.3**

- Topology sort change the relative position between words for better contextual word representation
- GCN and graph attention can aggregate both word and node representations to infer answers
- Both together: complementary

Heterogenous knowledge sources

Knowledge Sources	Dev Acc
None	68.9
ConceptNet	75.3
Wikipedia	73.5
ConceptNet + Wikipedia	**79.3**

- None: XLNet large model
- Both sources individually bring about improvement
- Combining both: much larger benefit

Conclusion

- Knowledge Extraction into graphs
  - ConceptNet (structured)
  - Wikipedia (unstructured)

  Heterogeneous background knowledge
- Graph-based reasoning
  - Contextual word representation learning module (Top. Sort + XLNet)
  - Inference module (GCN + Attention)
- State-of-the-art performance: 75.3%
- Graph structure of evidence sentences: basis for reasoning in commonsense question answering task

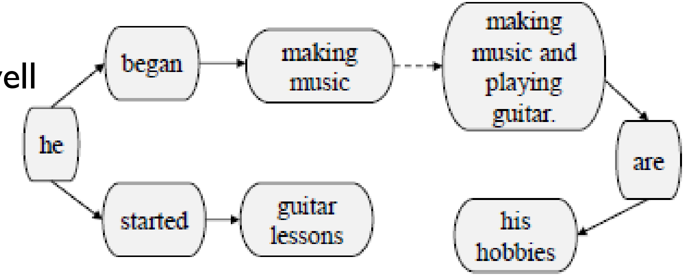
Issues

- Opening example in paper:

  Question: What do people typically do while playing guitar?
  A. cry **B**. hear sounds **C**. singing (✓) **D**. anthritis **E**. making music

  – Claim: "Dataset built in a way that answer choices share the same relation with question concept" ✘
- Semantic Role Labeling: typing errors
  – "Subjective" refers to → "subject"
  – "Objective" refers to → "object"

  Does not reflect well
- Wiki-Graph example
  – "Node A is contained in node B and the #words (A) > 3"



- Uses **only** entities in question to extract knowledge
  – Replacing "typically" with "never" would not change Concept-Graph, rely only on Wiki-Graph ✘
- Removal of stopwords during Wikipedia (Elastic Search)
  – Words like "not" would be skipped, this would give opposite results
  – BERT-Large baseline can't deal with negation either [1]
- Robustness: case studies of failed examples absent

Discussion

- Error Analysis (in paper): extracted evidence lack answer; two options too similar
- Limitations (opinion) for other graph-based reasoning (not commonsense)
    - ➢ Question Answering via Integer Programming over Semi-Structured Knowledge [3]
    - ➢ Question Answering as Global Reasoning over Semantic Abstractions [4]
  - This paper and [4] use SRL. [3] uses table schema, WordNet-based entailment score.
    - Support graph mathematically rigorous than Concept/Wiki graphs
  - Both use structure of graph to formulate ILP problem
  - XLNet representations vs. ILP
    - Pre-trained models perhaps perform better, but representations/constraints not explainable
  - Using SRL for unstructured → structured knowledge: important advantage
  - Does it address limitations of those papers?
    - Reasoning fails to exploit requisite knowledge from graph ✘
    - Natural language modules fail to represent the underlying phenomena of context ✘

References

- [1] Talmor, A.; Herzig, J.; Lourie, N.; and Berant, J. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In Proc. of NAACL, 4149–4158.
- [2] Ben Bogin, Matt Gardner, and Jonathan Berant. 2019. Global reasoning over database structures for textto-sql parsing. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLPIJCNLP), pages 3650–3655.
- [3] Daniel Khashabi, Tushar Khot, Ashish Sabharwal, Peter Clark, Oren Etzioni, and Dan Roth. 2016. Question answering via integer programming over semi-structured knowledge. arXiv preprint arXiv:1604.06076 .
- [4] D. Khashabi, T. Khot, A. Sabharwal, and D. Roth. Question answering as global reasoning over semantic abstractions. In AAAI'18, 2018.