KG^2 : Learning to Reason Science Exam Questions with Contextual Knowledge Graph Embeddings

Y. Zhang, H. Dai, K. Toraman, L. Song

Georgia Institute of Technology
Korea Advanced Institute of Science and Technology

Presenter: Bill Zhang
https://qdata.github.io/deep2Read

Outline

Introduction

- Most questions in current QA datasets only require surface-level reasoning; does not reveal full complexity of QA problem
- AI2 Reasoning Challenge (ARC) has been proposed to address this; leading models in QA for SQuAD and SNLI cannot beat random baseline
- ARC includes natural science exam questions; includes challenge set which IR and word co-occurence cannot solve
- Propose neural reasoning engine KG^2 which reads question, generates hypotheses given answer choices, and finds supporting sentences to verify hypotheses

ARC Dataset

Easy Set

- Questions can be easily solved because of substantial word overlap or word co-occurence within data corpus
- Ex: Which property of air does a barometer measure? (A) speed **(B) pressure** (C) humidity (D) temperature
- Example can be easily answered from the sentence "Air pressure will be measured with a barometer" from corpus

- Questions which IR and word co-occurence cannot solve
- Ex: Which property of a mineral can be determined by looking at it? **(A) luster** (B) mass (C) weight (D) hardness
- No sentences in corpus similar to "A mineral's luster can be determined by looking at it"; "mineral" also frequently occurs with mass and hardness
- Need connection of "luster" to "brightness" to "look"
- IR models underperform random baseline using both provided ARC corpus and even entire web through Google Search

Related Work

- IR-based methods and Markov Logic Networks for science QA
- DGEM (Khot et al. 2018) is a neutral entailment model which uses Open IE to generate hypothesis graph; most similar to this work
- Graph embeddings

Task Definition

- ARC Challenge Set consists of questions $\mathcal{D} = \{q_i, (c_i^{(1)}, ..., c_i^{(m)}), a_i\}_{i=1}^n$ where q_i is the question stem, $c_i^{(j)}$ is the jth answer option for q_i, and a_i is the correct answer
- Goal is to find correct answer
- ARC corpus has 14M science-related sentences from the Web with knowledge relevant to ARC
- Use of ARC corpus is optional

Approach

Generating Hypotheses

- A hypothesis h combines a question q with an answer choice c
- Ex: $q =$ "Which of these occurs due to the rotation of the Earth?" and $c =$ "Day and Night" $\Rightarrow h =$ "Day and night occurs due to the rotation of the Earth."
- Identify "wh" word (where, what, why,...) and replace with answer option
- If no "wh" word, append answer behind question stem; some special cases like "which of these"
- Corner cases considered negligible

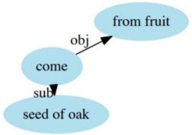
Approach

Searching Potential Supports

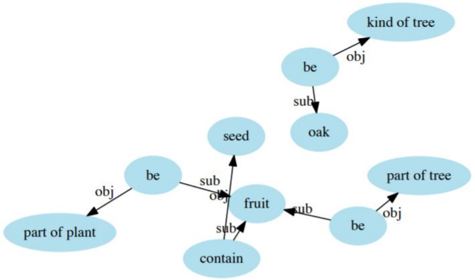
- Use hypothesis as query to search corpus
- ElasticSearch to quickly find
- Filter noisy sentences that contain negation words (i.e. not, except, etc.), unexpected characters, or are too long
- Pick top 20 sentences

- Use Open IE to extract relation triples from each sentence and use to construct contextual knowledge graph
- Each triple is $T(s, p, o_i)$ where s is the subject, p is predicate, and o_i is the ith object
- Construct graph by adding nodes s, p, and o_i and adding *subj*. and *obj*. directed edges
- Also edges with *time* and *loc*
- Similar graph is made with hypothesis and paired with knowledge graph

- Hypothesis "seed of oak comes from fruit"



(a) Knowledge graph for hypothesis

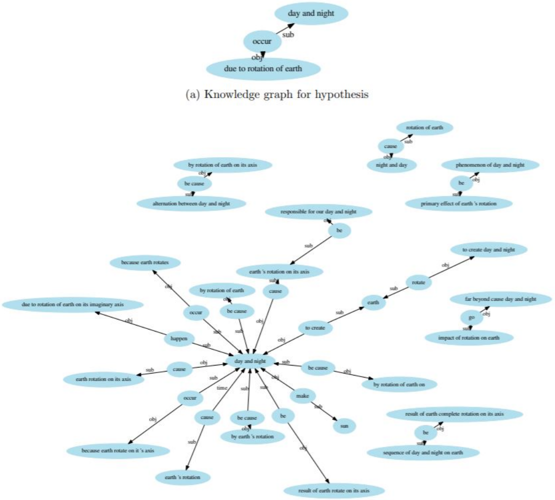


(b) Knowledge graph for supports

Figure 2: Example of knowledge graphs for paired hypothesis and supports.

Approach

Constructing Knowledge Graphs



(a) Knowledge graph for hypothesis

(b) Knowledge graph for supports

Figure 3: Another example of knowledge graphs for paired hypothesis and supports.

- With question q and candidate answer c, we now have hypothesis graph $G_{q,c}^{hypo}$ and supporting graph $G_{q,c}^{supp}$
- Choosing right answer becomes graph ranking problem
- Good $f : G^{hypo} x G^{supp} \rightarrow \mathbb{R}$ should assign highest score to correct hypothesis-supporting graph pair

- Let $G = (V, E)$ be a knowledge graph and $V_p \in V$ be set of predicate nodes
- Each node $v \in V$ has embedding vector μ_v capturing local information

$$\mu_v = h(\mathbf{x}_v, \mu_v^{(t-1)}, \{(\mu_u^{(t-1)}, e_{u,v})\}_{(u,v,e_{u,v}) \in E})$$

- $\mathbf{x}_v$ encodes text features of node generated by LSTM jointly trained with the supervision
- Edge type $e_{u,v}$ can be *time*, *loc*, etc.
- h is 2 layer neural network
- Run for T iterations

Approach

Graph Embeddings: Scoring Function

$$f(G^{hypo}, G^{supp}) = f(\{\mu_u\}_{u \in V_p^{hypo}}, \{\mu_v\}_{v \in V_p^{supp}}) = \sigma(\max_{u,v} \frac{\mu_u^T \mu_v}{\|\mu_u\| \|\mu_v\|} - 0.5)$$

- σ is sigmoid
- -0.5 shift used to center matching score at 0
- Max inner product search between all pairs of predicate node embeddings
- Mimics reasoning on most relevant hypothesis and corresponding supporting evidence, since embedding vector already captures information within T-hop neighborhood

- Use ARC Challenge Set for all experiments
- 2,590 questions from human exams
- For each question, QA system receives 1 point for correct answer and $1/k$ points for k way tie which includes correct answer
- ARC corpus used optionally for all models

- **Guess-all/Random**: Select all answers or select 1 random answer
- **IR-ARC**: Send question stem plus each option as query to search engine build on ARC corpus; choose option which yields sentence with highest search score
- **IR-Google**: Same as above, except with Google Search API to search entire Web
- **TableILP**: Table-based reasoning formatted as Integer Linear Program
- **TupleInterference**: Search for graph which best connects terms in question with answer via knowledge from Open IE

Experiments

Baselines

- **DecompAttn**: Neural entailment model adapted to multiple choice QA; Top SNLI performer
- **DGEM-OpenIE**: Neural model for sentence-level entailment, using Open IE to create structured representation of hypothesis; Top SciTail Performer
- **BiDAF**: Span prediction QA; Top SQuAD performer

- None of the baselines perform significantly better than random
- KG^2 achieves score of 31.70, improving on previous state-of-the-art by 17.5%

Method	Test Scores
IR-ARC	20.26
IR-Google	21.58
TupleInference	23.83
DecompAttn	24.34
Guess-all / Random	25.02
DGEM-OpenIE	26.41
BiDAF	26.54
TableILP	26.97
KG^2	**31.70**

- Still far from "passing" exam
- More than half the questions do not have enough support; even humans cannot solve given the supporting sentences
- Could be caused by limited coverage of corpus and importance of sentences with low word overlap for reasoning
- 12% lose key information in graphs due to Open IE
- Answering all learnable questions gives upper bound of 36.25 points

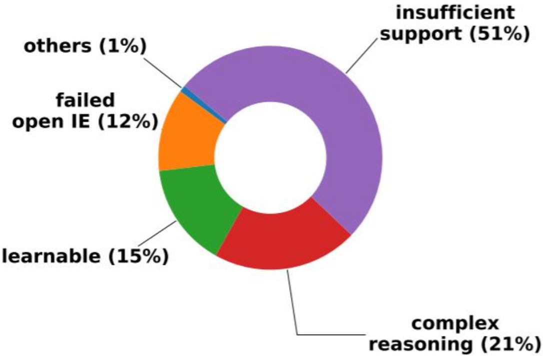


Figure 1: Distribution of various difficulties in solving the ARC Challenge Set.

Conclusion

- Present a neural reasoning engine for answering science exam questions which learns to reason over contextual knowledge graphs
- Method outperforms existing QA systems on ARC dataset
- Future work on how to exploit knowledge sources and trying to improve quality of open IE by sentence parsing