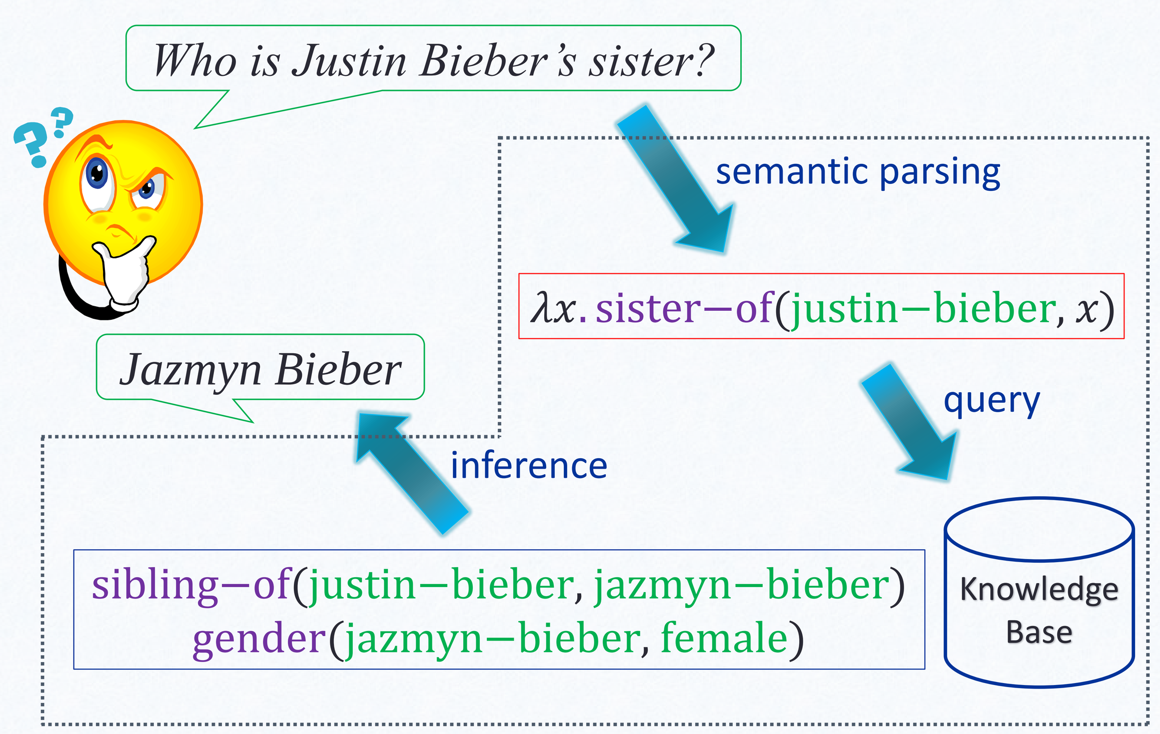
Semantic Parsing for Single-Relation Question Answering

Scott Wen-tau Yih, Xiaodong He & Christopher Meek
{scottyih, xiaohe, meek}@microsoft.com
Microsoft Research

ACL 2014
Baltimore, June 22-27

Capture Question Variations using Convolutional Neural Networks. 66% Error Reduction!

Open-Domain QA using Knowledge Base



Single-Relation Questions

- Most common questions in the search query logs
  - *"How old is Kirk Douglas, the actor?"*
  - *"What county is St. Elizabeth MO in?"*
  - *"What year was the 8 track invented?"*
  - *"Who owns the Texas Rangers?"*
- Foundation for answering complicated questions
  - *"Name a director of movies starred by Tom Hanks."*
  - CKY parsing that chains answers of single-relation questions [Bao et al., 2014]
- Challenge: lots of ways to ask the same question
  - *"What was the date that Minnesota became a state?"*
  - *"Minnesota became a state on?"*
  - *"When was the state Minnesota created?"*
  - *"Minnesota's date it entered the union?"*
  - *"When was Minnesota established as a state?"*
  - *"What day did Minnesota officially become a state?"*
  - …

Key Ideas & Related Work

- Simple Context-Free Grammar
  - Separate a question into a relation pattern and an entity mention
- Paraphrase detection using convolutional neural net
- Inspired by Paralex [Fader et al. 2013]
  - 35M question paraphrase pairs from WikiAnswers
  - Learn weighted lexical matching rules

Task & Problem Definition

Input
- A KB as a collection of triples (r, e_1, e_2)
- A single-relation question, describing a relation and one of its entity arguments
  "When were DVD players invented?"

Output
- An entity that has the relation with the given entity

High-level Approach: Semantic Parsing

$Q =$ *"When were DVD players invented?"*

$Q \rightarrow P \wedge M$
$P \rightarrow$ *when were X invented*
$M \rightarrow$ *DVD players*
when were X invented $\rightarrow$ be−invent−in$_2$
DVD players $\rightarrow$ dvd−player

$\lambda x.\,$be−invent−in(dvd−player, x)

Procedure: Enumerate All Hypotheses

$Q =$ "*When were DVD players invented*?"

$P \rightarrow$ *when X players invented*
$M \rightarrow$ *were DVD*

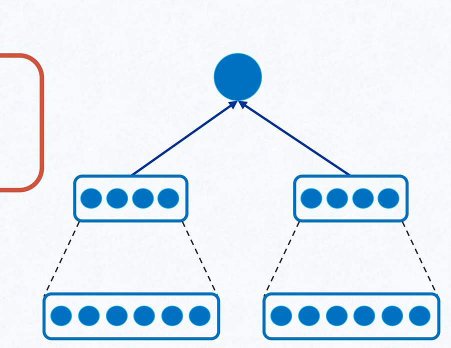
$Q =$ "*When were DVD players invented*?"

$P \rightarrow$ *when were X invented*
$M \rightarrow$ *DVD players*

$Prob($be−invent−in$_2 |$ *when were X invented*$) = 0.5$
$Prob($dvd−player$|$*DVD players*$) = 0.7$

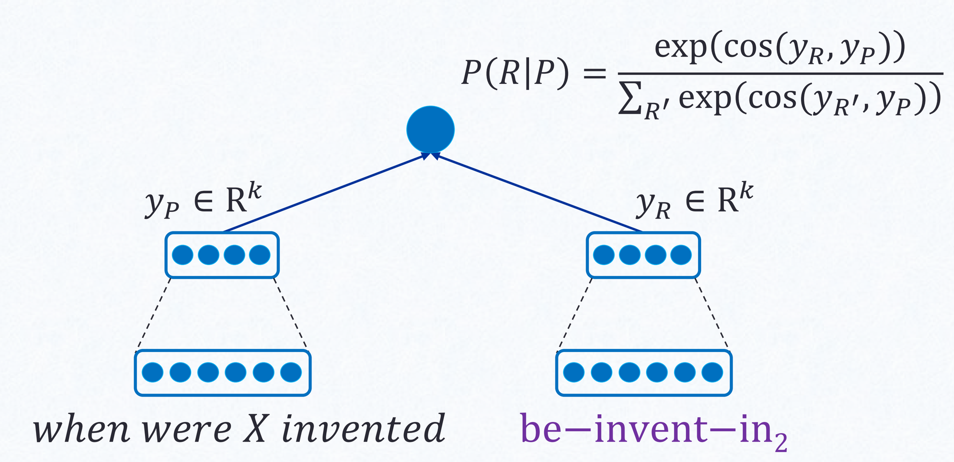
$Prob(\lambda x.\,$be−invent−in(dvd−player, $x)|Q) = 0.35$

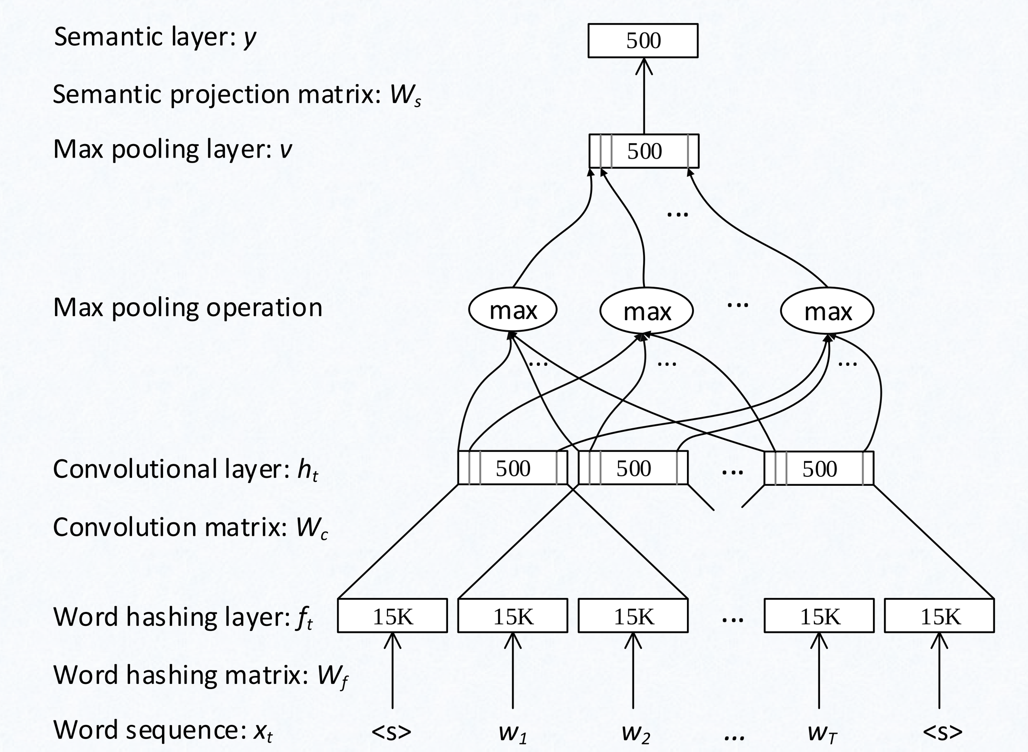
Paraphrase Detection via Siamese Neural Networks!



Convolutional Deep Semantic Similarity Model [Shen et al., 2014]

Siamese neural networks
- Input is mapped to two k-dimensional vectors
- Probability is determined by softmax of their cosine similarity





Experiments: Data & Task

Knowledge base: ReVerb [Fader et al., 2011]

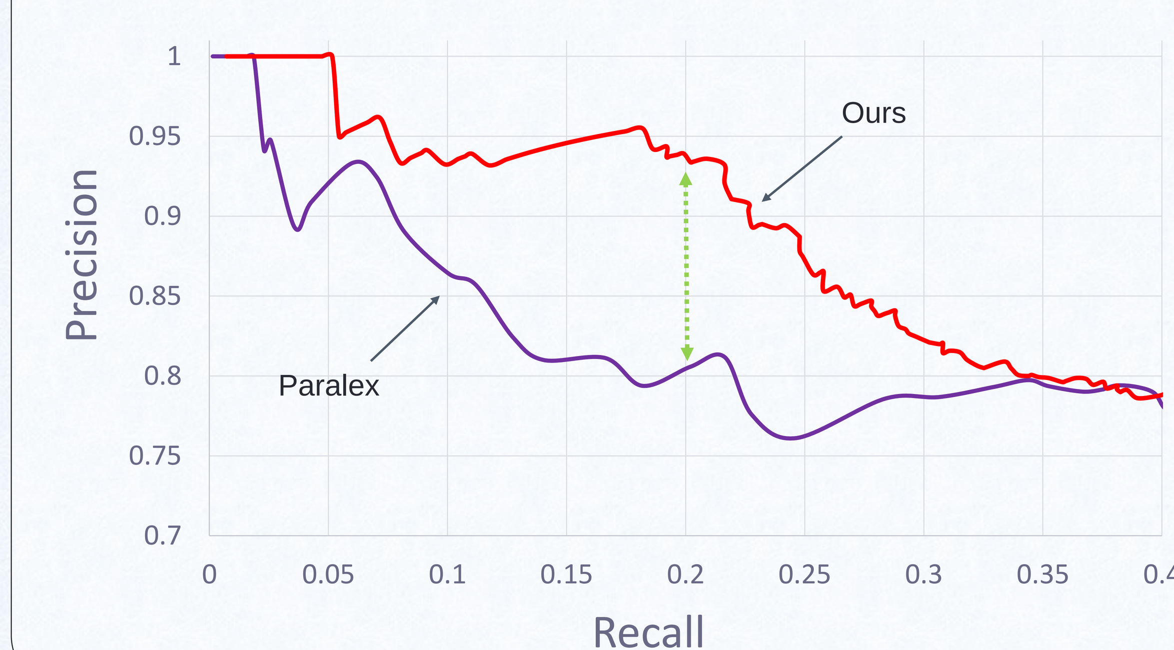
Relation	Entity Argument #1	Entity Argument #2
be-official-language	chinese-and-english	hong-kong
be-second-largest-city-in	arequipa	peru
be-tallest-mountain-in	ararat	armenia
have-population-of	city-of-vancouver	587,891
provide	microsoft	office-software
use-for	laser	lasik
…	…	…

Paralex dataset [Fader et al., 2013]
- 1.8M (question, single-relation queries)
  - *When were DVD players invented?*
  - $\lambda x.\,$be−invent−in(dvd−player, x)
- 1.2M (relation pattern, relation)
  - *When were X invented?*
  - be−invent−in$_2$
- 160k (mention, entity)
  - *Saint Patrick day*
  - st−patrick−day

Task: Question Answering
- *What language do people in Hong Kong use?*
  be−speak−in(english, hong−kong)
  be−predominant−language−in(cantonese, hong−kong)
- *Where do you find Mt Ararat?*
  be−highest−mountain−in(ararat, turkey)
  be−mountain−in(ararat, armenia)

Experiments: Results

- Same test questions in the Paralex dataset
- 698 questions from 37 clusters



Conclusions

- A new semantic parsing framework for single-relation questions
  - Use a semantic similarity function to match patterns and relations, as well as mentions and entities
  - Semantic similarity model – Convolutional neural networks with letter-trigram vector input
  - Go beyond bag-of-words and handle OOV better
  - Outperform previous work using lexical matching rules
- Future work
  - Apply this approach to more structured KB (Freebase)
  - Extend this work to handle multi-relation questions