Factoid Question Answering

Roy Aslan (ra2752@Columbia.edu)

A Neural Network for Factoid Question Answering over Paragraphs

Mohit Iyyer, Jordan Boyd-Graber, Leonardo Claudino, Richard Socher, and Hal Daumé III

Task and Setting

- Factoid question answer
- Quiz Bowl dataset
  - Multi sentence "question" mapped to entity as the "answer"
  - Questions exhibit pyramidality: initial sentences are more subtle (e.g., few named entities)

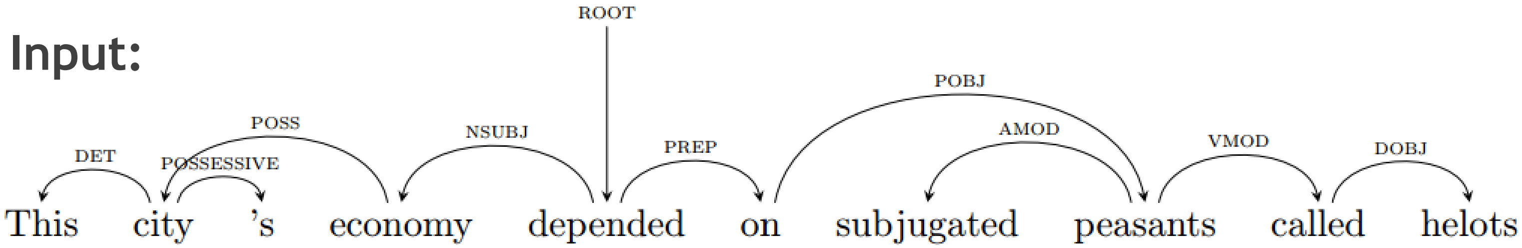
QUESTION:
He left unfinished a novel whose title character forges his father's signature to get out of school and avoids the draft by feigning desire to join. A more famous work by this author tells of the rise and fall of the composer Adrian Leverkühn. Another of his novels features the jesuit Naptha and his opponent Settembrini, while his most famous work depicts the aging writer Gustav von Aschenbach. Name this German author of The Magic Mountain and Death in Venice.
ANSWER: Thomas Mann

Contributions

- Bag of words representation relies on indicative named entities
- Paragraph (versus sentence) length inputs
- Proposed dependency-tree recursive NN (DT-RNN) model exploits semantic/compositional information
  - Previous work used DT-RNN to map text descriptions to images
  - Here question/answer representations can be learned in same vector space
  - Robust to varying syntax (same question can be asked in a variety of ways)

Model illustration in next slide

Input:



Leaf hidden vector: $h_{\text{helots}} = f(W_v \cdot x_{\text{helots}} + b)$

tanh

x to h mtrx

Word2vec (mikolov)

Internal node hidden vector:

$$h_{\text{called}} = f(W_{\text{DOBJ}} \cdot h_{\text{helots}} + W_v \cdot x_{\text{called}} + b).$$

Dependency relation mtrx

1 for each 46 relations

Root node:

$$h_{\text{depended}} = f(W_{\text{NSUBJ}} \cdot h_{\text{economy}} + W_{\text{PREP}} \cdot h_{\text{on}} + W_v \cdot x_{\text{depended}} + b). \qquad (3)$$

General formula for node: $f(W_v \cdot x_w + b + \sum_{k \in K(n)} W_{R(n,k)} \cdot h_k)$

Dependents

Training

- Questions and answers trained in same vector space
- Want question sentences near answers and far from incorrect answers
- Given a question sentence and correct answer pair, select j incorrect answers

Training

Correct answer

Node in DT

Random set of (100) wrong answers

Sentence error: $$C(S,\theta) = \sum_{s\in S}\sum_{z\in Z} L(rank(c,s,Z))\max(0, 1 - x_c \cdot h_s + x_z \cdot h_s),$$

parameters

Rank estimator: $(|Z|-1)/K$ (sample K till violation)

Rank loss: $$L(r) = \sum_{i=1}^{r} 1/i$$

Error over all T sentences and N nodes: $$J(\theta) = \frac{1}{N}\sum_{t\in T} C(t,\theta).$$

Back propagation: $$\frac{\partial C}{\partial \theta} = \frac{1}{N}\sum_{t\in T}\frac{\partial J(t)}{\partial \theta}$$

Experiments

- About 10k quiz bowl question mapped to about 1k answers
- About a dozen training examples per answer (minimum 6)
- Number of random wrong answers set to 100
- All parameters randomly initialized (except preprocessed word2vec vectors)
- Trans sentential averaging
  - Concatenate and average node representations to form sentence representation
  - Average representations of all sentences in question (paragraph)
- Question representation is fed into logistic regression classifier for answer prediction

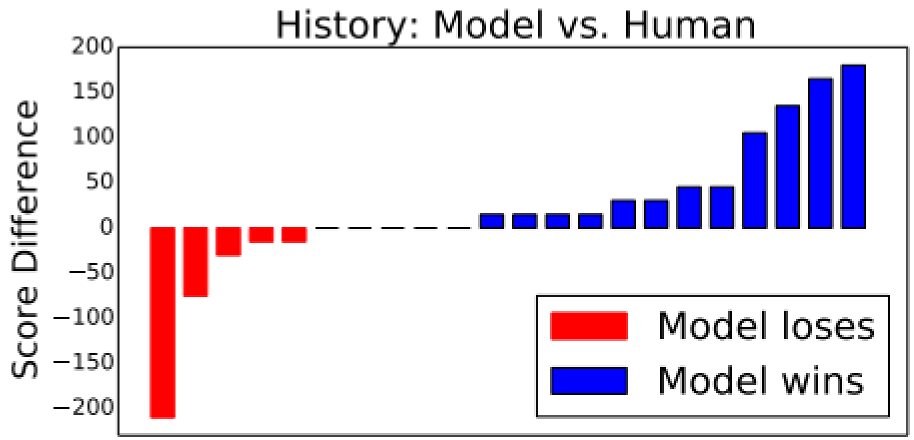
Results – vs baselines

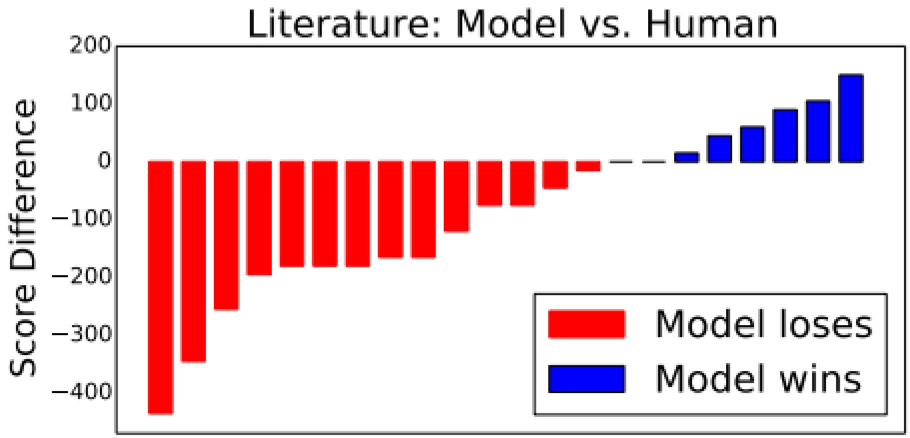
Pos 1 and Pos 2 means at first/second sentence position within question

Model	History			Literature		
	Pos 1	Pos 2	Full	Pos 1	Pos 2	Full
BOW	27.5	51.3	53.1	19.3	43.4	46.7
BOW-DT	35.4	57.7	60.2	24.4	51.8	55.7
IR-QB	37.5	65.9	71.4	27.4	54.0	61.9
FIXED-QANTA	38.3	64.4	66.2	28.9	57.7	62.3
QANTA	**47.1**	**72.1**	**73.7**	**36.4**	**68.2**	**69.1**
IR-WIKI	53.7	76.6	77.5	41.8	74.0	73.3
QANTA+IR-WIKI	**59.8**	**81.8**	**82.3**	**44.7**	**78.7**	**76.6**

Results – vs human

Each bar represents individual human player





Semantic Parsing for Single-Relation Question Answering

Wen-tau Yih, Xiaodong He, and Christopher Meek

Task and Setting

- Answering single relation factual questions
  - “Who is the CEO of Tesla?”
  - “Who founded Paypal?”
- Multi relation questions are out of scope
  - “When was the child of the former Secretary of State in Obama’s administration born?”

Contribution

- Novel dual semantic similarity model using CNN
  - Map entity mention to entity in KB
  - Map relation pattern to relation
- "When were DVD players invented?"
  - Entity mentioned: dvd-players
  - Relation: be-invent-in

$$Q \rightarrow RP \wedge M \qquad (1)$$

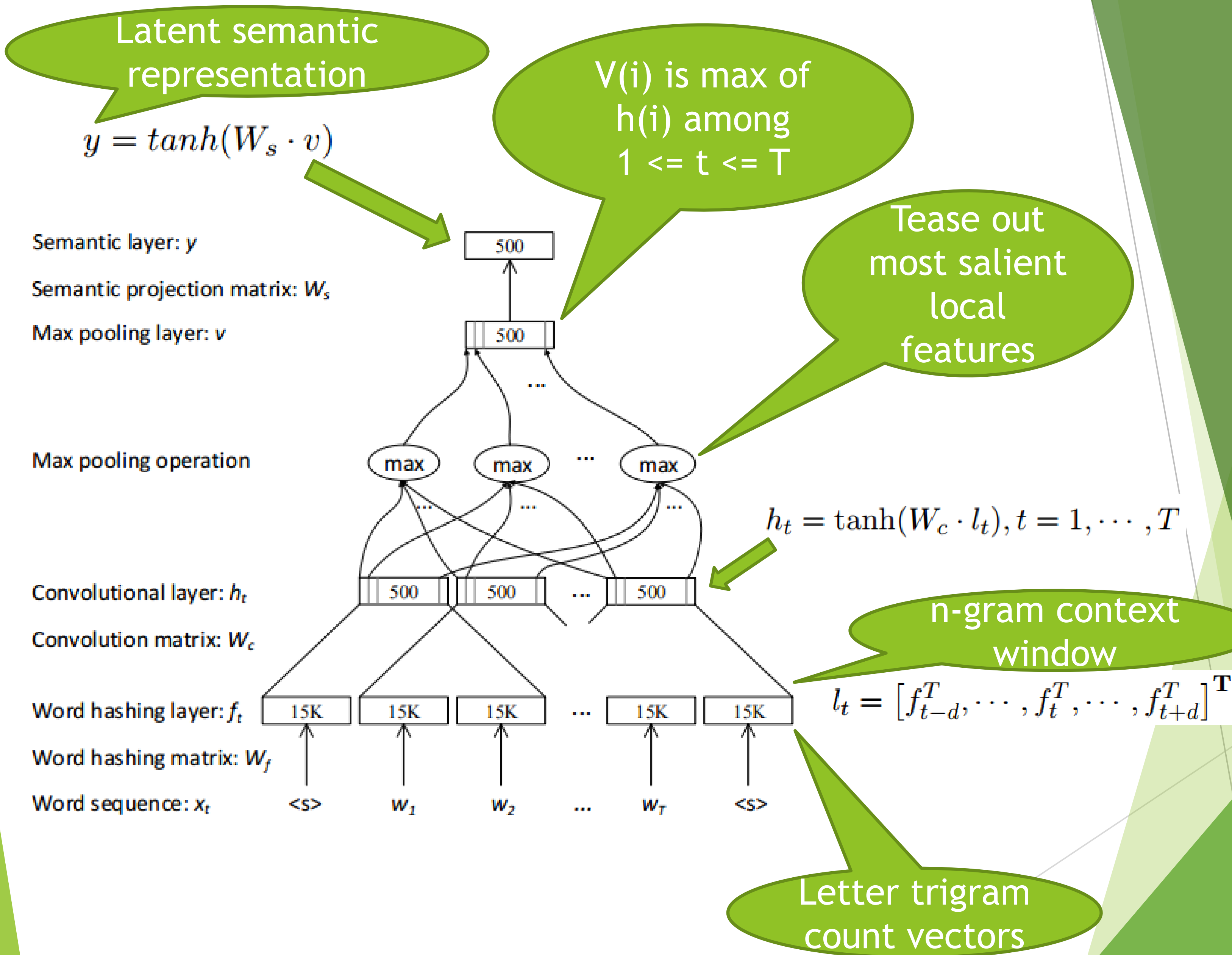
$$RP \rightarrow \textit{when were X invented} \qquad (2)$$

$$M \rightarrow \textit{dvd players} \qquad (3)$$

$$\textit{when were X invented} \rightarrow \texttt{be-invent-in} \qquad (4)$$

$$\textit{dvd players} \rightarrow \texttt{dvd-player} \qquad (5)$$

Model in Next Slide



Latent semantic representation
$y = tanh(W_s \cdot v)$
V(i) is max of h(i) among 1 <= t <= T
Tease out most salient local features
$h_t = \tanh(W_c \cdot l_t), t = 1, \cdots, T$
n-gram context window
$l_t = [f_{t-d}^T, \cdots, f_t^T, \cdots, f_{t+d}^T]^{\mathbf{T}}$
Letter trigram count vectors
Semantic layer: y
Semantic projection matrix: W_s
Max pooling layer: v
Max pooling operation
Convolutional layer: h_t
Convolution matrix: W_c
Word hashing layer: f_t
Word hashing matrix: W_f
Word sequence: x_t
500
500
max
max
max
500
500
500
15K
15K
15K
15K
15K
<s>
w_1
w_2
w_T
<s>

Training

- Two models are trained from
  - Pattern-relation pairs
  - Mention-entity pairs
- 100 randomly selected negative examples
- Softmax based on cosine similarity used for calculating probability of correct relation given an input

$$P(R^+|Q) = \frac{\exp(\gamma \cdot \cos(y_{R^+}, y_Q))}{\sum_{R'} \exp(\gamma \cdot \cos(y_{R'}, y_Q))}$$

- Maximize log probability using SGD

Experiments

- PARALEX dataset
  - Derived 1.2M patterns-relation pairs with argument position for answer
  - 160K mention-entity pairs
- Context windows size set to 3
- Question evaluation:
  - Compute top 150 relation candidates for pattern (based on similarity score)
  - For each candidate, compute mention and argument entity similarity (among KB triplets with this relation)
  - Product of the pattern-relation and mention-argument probabilities (softmax based on cosine) is used as final ranking
  - Predefined threshold to establish precision-recall trade-off

Results

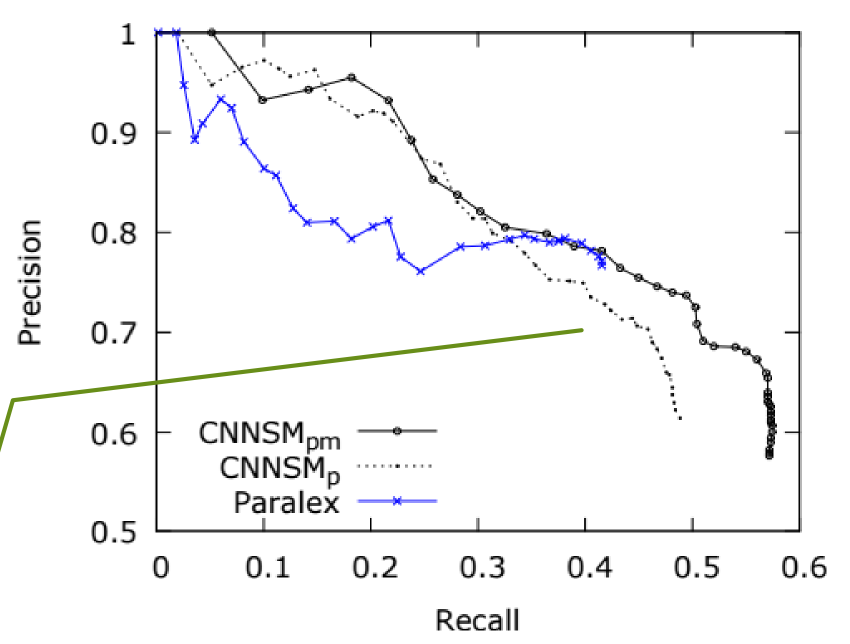
$$F_1 = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}.$$

	F_1	Precision	Recall	MAP
CNNSM_{pm}	**0.57**	0.58	**0.57**	**0.28**
CNNSM_{p}	0.54	0.61	0.49	0.20
PARALEX	0.54	**0.77**	0.42	0.22

Full model

Surface level mention-entity similarity

Baseline



Gap at higher recall

Results - examples

- What is the national anthem in the France?
  PARALEX: be-currency-in.r euro.e france.e
  CNNSM: be-national-anthem-of.r la-marseillaise.e france.e

- What is the title of france national anthem?
  PARALEX: be-national-dog-of.r poodles.e france.e
  CNNSM: be-national-anthem-of.r la-marseillaise.e france.e

- What is the name of the national anthem of France?
  PARALEX: be-national-language-in.r french.e france.e
  CNNSM: be-national-anthem-of.r la-marseillaise.e france.e

Questions? Go back to beginning for first paper