Extracting Commonsense Properties from Embeddings with Limited Human Guidance

Yiben Yang, Larry Birnbaum, Ji-Ping Wang, Doug Downey

ACL, 2018

Barry Plunkett (eplu@sas.upenn.edu)

March 25, 2019

Penn Engineering

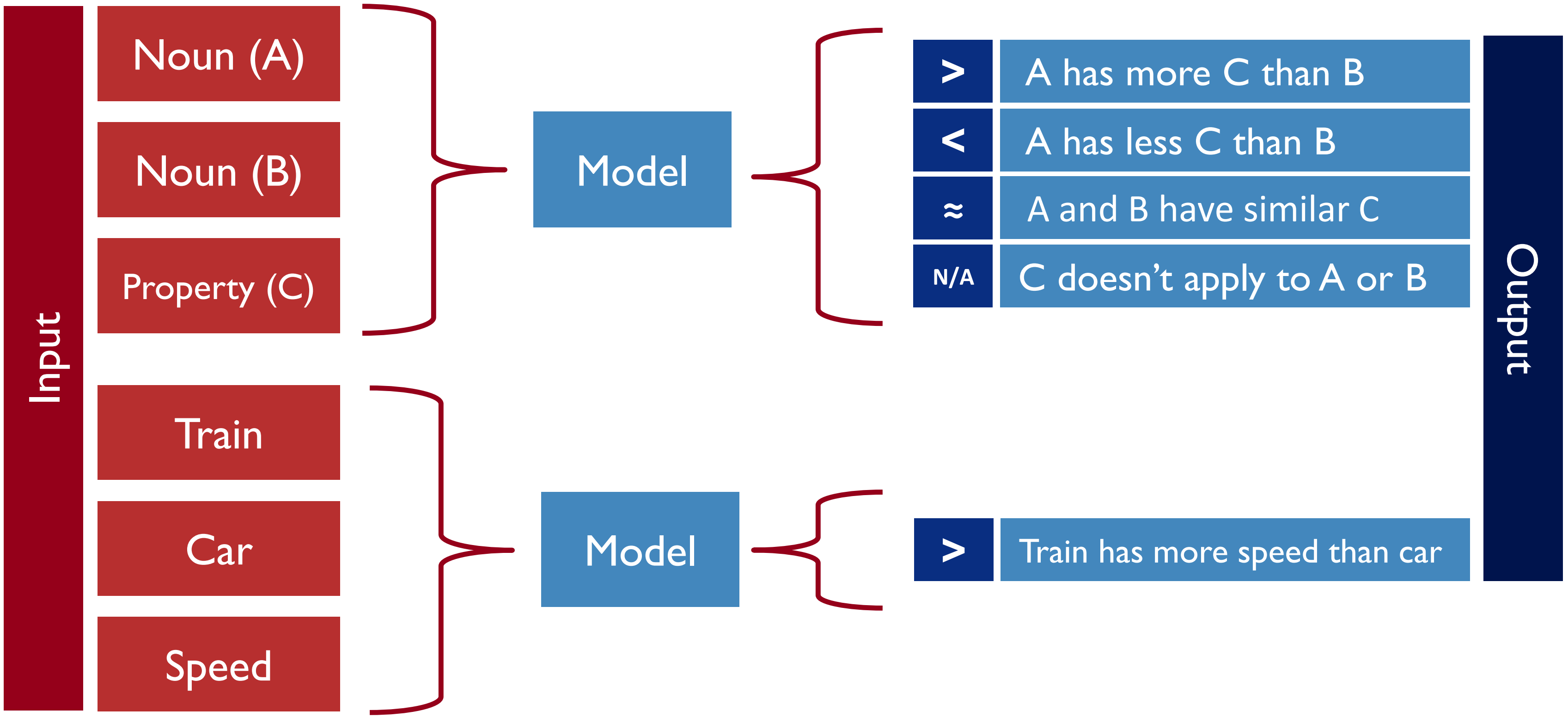
Problem & Motivation

Automatic extraction of binary comparisons from text

Is a train faster than a car?

Is wood more durable than steel?

Problem & Motivation



Problem and Motivation – Reporting Bias

"John picked up his briefcase and left for work."

Size

John > briefcase

"The earthquake toppled the wood buildings, but the steel ones survived."

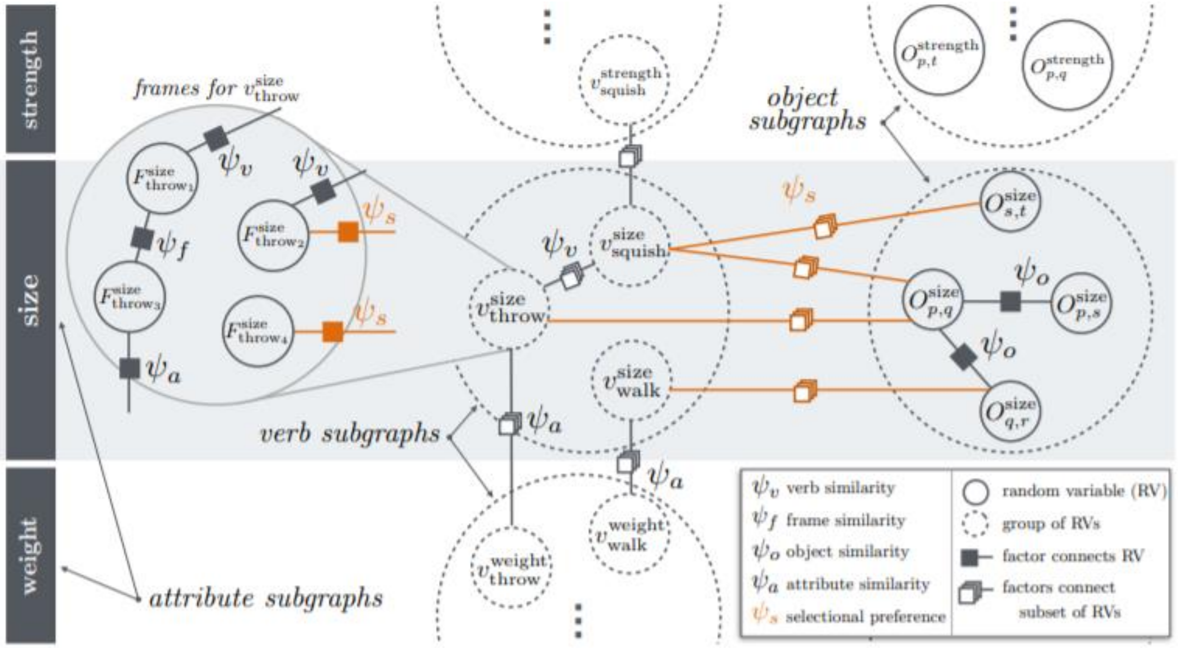
Durable

wood < steel

Contents:

- Previous approaches
- Contributions
  - SotA Performance on standard task
  - New task and baseline
- Conclusions
- Shortcomings and extensions

Previous approaches (SoTA)



Forbes & Choi (2017) introduce VERB PHYSICS comparison dataset and establish baseline model

Contributions of this work

- Achieve state-of-the-art performance on task introduced by F&C
  - Requires no additional annotation beyond training set
  - Model allows zero-shot learning on unseen properties and nouns
- Introduces alternative formulation of task
  - Provides baseline and dataset for this task

Details of Contributions – Setup

Formalization – Supervised multiclass classification

$$P(\mathbf{L}|\mathbf{O_1}, \mathbf{O_2}, \mathbf{Property}), \mathbf{L} \in \{\boxed{<}, \boxed{>}, \boxed{\approx}\}$$

	Description	Examples	
O_1	First object in comparison	Train	Wood
O_2	Second object in comparison	Car	Steel
Property	Property being compared for O_1, O_2	Speed	Durable
L	Label of comparison (>, <, ≈)	>	<

Penn Engineering

Details of Contributions – Setup

Assumptions

- Objects with similar word embeddings have similar properties

Skyscraper	Hut
Highrise	Cabin

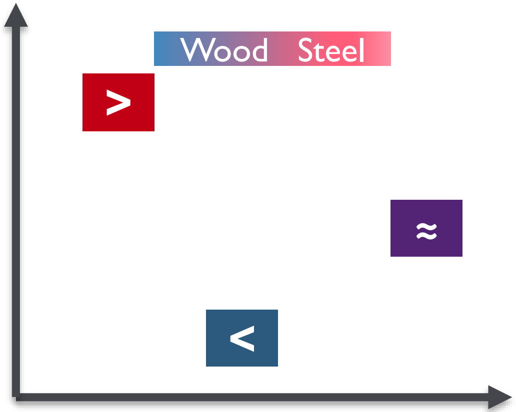
- A property can be represented as a set of pole adjectives

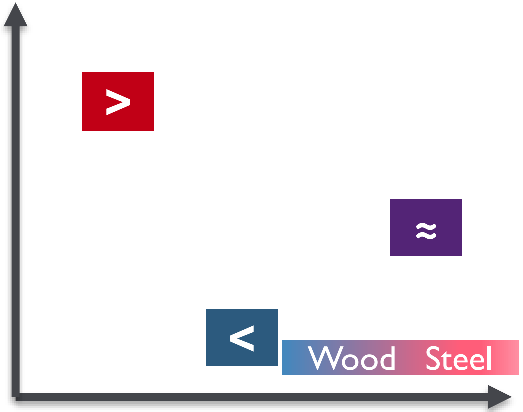
	>	≈	<
Durable	Durable	Similar	Fragile
Speed	Fast	Similar	Slow

Details of Contributions – Goals

- Learn projection for embeddings pair of nouns into vector space containing embeddings for pole adjectives
- Predict closest pole
- Use labeled comparisons to train projections of pairs to be near correct pole

Durable	
>	**Durable**
≈	Similar
<	Fragile





Details of Contributions – Model

Model Inputs (Q)	Nouns	X	Embedding of first comparison object (O_1)	Wood
		Y	Embedding of first comparison object (O_2)	Steel
	Pole Adjectives	$R_>$	Embedding for > adjective	Durable
		$R_\approx$	Embedding for “similar”	Similar
		$R_<$	Embedding for < adjective	Fragile

Penn Engineering

Details of Contributions – Model

- Model Inputs (Q)
  - Nouns
    - X
    - Y
  - Pole Adjectives
    - $R_>$
    - $R_\approx$
    - $R_<$

Property Commonsense Embedding (PCE) Model

Model Function – Softmax over "cosine similarity" between projection of object pair and each comparison label

$$P(\mathbf{L} = s|Q) = softmax(R_s \cdot \sigma((X \oplus Y)W))$$

Arbitrary Pole Embedding

Activation Function

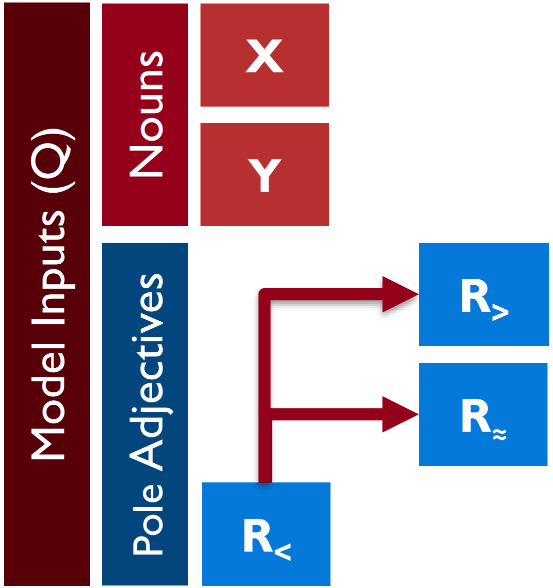
Object Embedding Concatenation

Projection Matrix

$$s \in \{<, >, \approx\}$$

Penn Engineering

Details of Contributions – One Pole Model



Model receives only one adjective embedding input

- Receives only “<“ adjective embedding
- Learns a projection for each of the other pole embeddings

$$R_{\approx} = \sigma(R_{<} W_2)$$

Activation Function

Projection Matrix

$$R_{>} = \sigma(R_{<} W_3)$$

Details of Contribution - Comparison

PCE Network (Yang)	Probabilistic Factor Graph (F&C)
• No frames • Word embeddings • Standard multiclass train scheme • Prediction scheme identical for seen and unseen objects • Can predict unseen attributes	• Dependency parsing to identify frames, verbs, objects • Word embeddings • Two-pass training scheme • Message passing for predicting unseen pairs • Cannot predict unseen attributes

Penn Engineering

Details of Contributions - Four-way Model

Identical to standard formulation with additional class

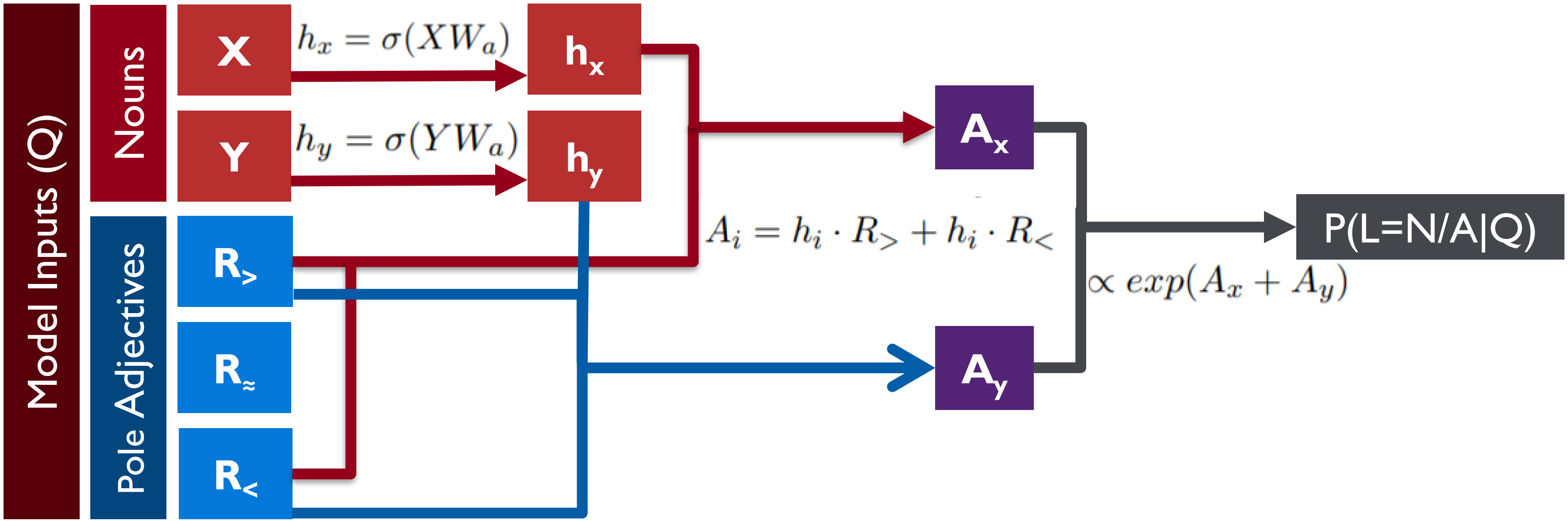
$$P(\mathbf{L}|\mathbf{O_1}, \mathbf{O_2}, \mathbf{Property}), \mathbf{L} \in \{\boxed{<}, \boxed{>}, \boxed{\approx}, \boxed{\text{N/A}}\}$$

Property does not apply

Is paper more intelligent than plastic?

Details of Contributions - Four-way Model

Identical three-way model with additional output for N/A in softmax layer



Details of Contributions – New Dataset

A shrimp is less (<) furry than a lion.

PROPERTY COMMONSENSE

- New dataset for four-way classification task
- 32 properties, 689 objects including proper nouns
- Properties and objects randomly permuted to generate samples
- Hand-labeled by 1-labeler (.64 Cohen's Kappa)

Experimental Setup

- Set-up for all models and experiments
  - Identity activation function for all σ
  - Full-batch gradient descent using ADAM with no tuning
  - Dropout before output (p=.5)
  - gLoVe (300d), word2vec (300d) , LSTM (1024d) embeddings
  - Accuracy for evaluation

Experiments

- **Zero-shot learning**: Make predictions on unseen class or attribute
  - For this task, zero-shot learning means making predictions on unseen property

Train Set	Test Set
Size Speed Rigidity Strength	Weight

Penn Engineering

Experiments

- Three-way classification task (VERB PHYSICS)
- Hold-one-out zero-shot learning on properties (VERB PHYSICS)
- Four-way classification task (PROPERTY COMMONSENSE)

	VERB PHYSICS	PROPERTY
Train	594	1819
Test	6000	1489
Total	6594	3308

Results and Analysis

- PCE and PCE(one-pole) achieve SotA performance on 3-way task

Model	Overall Accuracy
Majority Baseline	0.51
F&C	0.70
PCE(one-pole)	**0.75**
PCE(LSTM)	**0.76**

- Establishes strong baseline for 4-way task

Model	Overall Accuracy
Majority Baseline	0.51
PCE(gLoVe)	0.63
PCE(Word2Vec) and PCE(LSTM)	**0.67**

Results and Analysis

Establishes a strong baseline for zero-shot learning task

Model	Weight	Size	Strength	rigidity	speed
Emb-Similarity	0.37	0.53	0.48	0.43	0.35
PCE(one-pole)	0.73	0.71	0.67	0.53	0.34
PCE	**0.74**	**0.73**	**0.70**	**0.62**	**0.58**

Penn Engineering

Conclusions

- Embeddings may help address reporting bias for other commonsense knowledge tasks
- Learning projection of pairs of noun embeddings into vector space containing adjective embeddings enables generalization among properties and among nouns
- PROPERTY COMMON SENSE provides a harder (more sparse) framework

Shortcomings

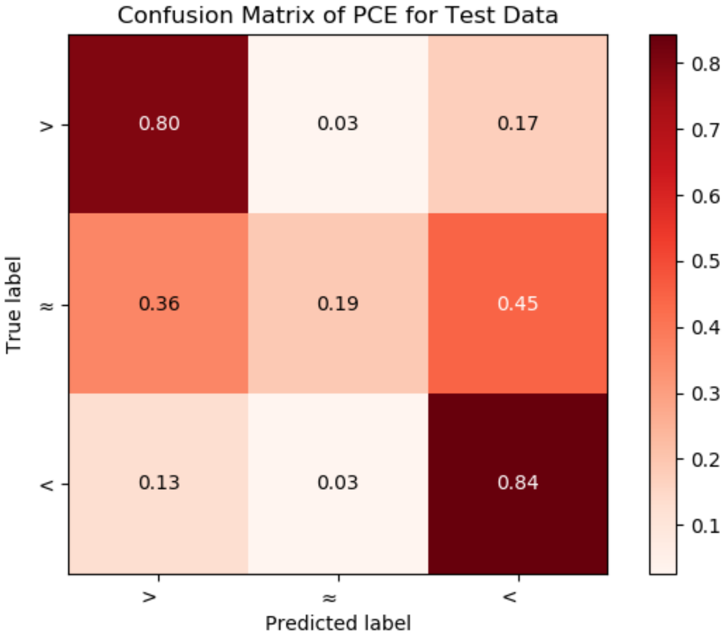
- Difficult to identify ceiling of approach
- Loss uniform when prediction not equal to truth
- Datasets are biased
  - Imbalanced with respect to objects and true labels

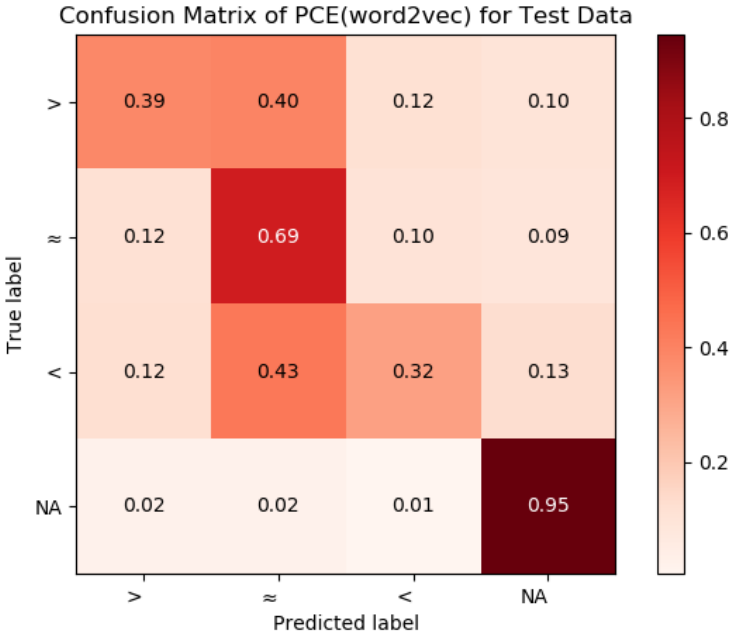
Dataset	$>$	$\approx$	$<$	N/A	Total
VERB PHYSICS	0.56	0.10	.34	0.00	6594
PROPERTY COMMON SENSE	0.18	0.25	0.18	0.39	3308

Dataset	Component	Density (Top 20%)
VERB PHYSICS	Properties	0.22
	Nouns	0.54
PROPERTY COMMON SENSE	Properties	0.22
	Nouns	0.3

Shortcomings

- Models highly sensitive to class imbalance





Extensions and Improvements

- More extensive experiments on architecture
  - Non-linear activation functions
  - More feed-forward layers to learn projection
  - Hyperparameter tuning
  - Measure performance after up-sampling / down-sampling and using less data
- Cost sensitive Loss
  - Penalize > more than ≈ when ground truth is <
  - Fixed penalty for N/A
- Develop unbiased dataset with respect to nouns, properties, and true labels