Hybrid Attention-Based Prototypical Networks for Noisy Few-Shot Relation Classification

Tianyu Gao, Xu Han, Zhiyuan Liu, Maosong Sun

Tsinghua University

Background

- Most existing methods for **relation classification** (RC) rely on distant supervision, which suffers from false positive, wrong labelling and data sparsity.
- Inspired by how people grasp new knowledge from few instances, we formalize RC as a **few-shot** learning problem.
- Yet current few-shot models mainly focus on low-noise vision tasks. There are few models for dealing with noisy data and for language tasks.

Few-Shot Relation Classification

Relation Classification

Given one sentence and two entities in it, relation classification is to extract the type of relation between those two entities within a set of pre-defined relations.

Few Shot

In a N-way K-shot few-shot task, there will be N classes, which have K supporting samples for each, and a query instance. Few-shot models are required to classify the query instance into one of N classes.

Pipeline for Relation Classification

- Word embedding + position embedding
- CNN / RNN
- Max pooling + non-linear
- Linear

CNN Sentence Encoder

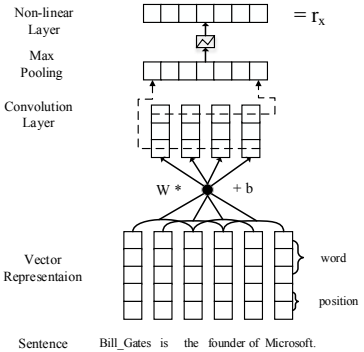


Figure: CNN sentence encoder (from Lin et al. 2016).

Approaches for Few-Shot Learning

- Meta Networks
- Neural Attentive Learner (SNAIL)
- Graph Neural Networks
- Prototypical Networks

Prototypical Networks

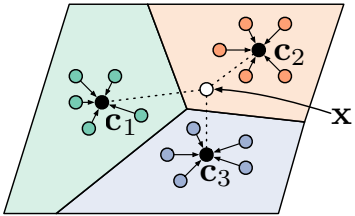


Figure: Prototypical Networks for few-shot learning.

Prototypical Networks

- Use the encoder to get the instance features
- Calculate *prototype* for each class
  - The original paper uses **average** of instance features for each class
- Calculate **distances** between instance features and prototypes
  - The original paper uses **Euclidean Distance** (Each dimension is equal)
- Classify instances by distances

Not All Instances Are Equal

When a human tries to classify a sentence into one of several classes...

Support Set of Relation **Date of Birth**	Query Instance
(A) On March 14, 1879, Einstein was born.	We are preparing for her birthday on Monday.
(B) "My birthday is January 13", she said.	
(C) The baby came to the world on November.	

Not All Instances Are Equal

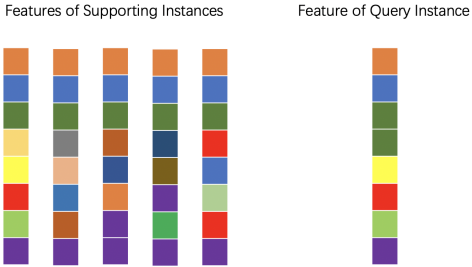
When a human tries to classify a sentence into one of several classes...

Support Set of Relation **Date of Birth**	Query Instance
(A) On March 14, 1879, Einstein was born.	We are preparing for her birthday on Monday.
(B) "My birthday is January 13", she said.	
(C) The baby came to the world on November.	

For each query instance, give those in the support set that closer to it higher weights would be better when calculating *prototype*.

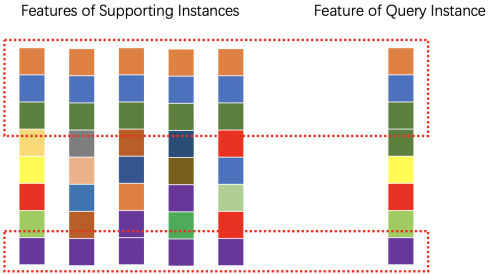
Not All Features Are Equal

When a human tries to classify a sentence into one of several classes...



Not All Features Are Equal

When a human tries to classify a sentence into one of several classes...



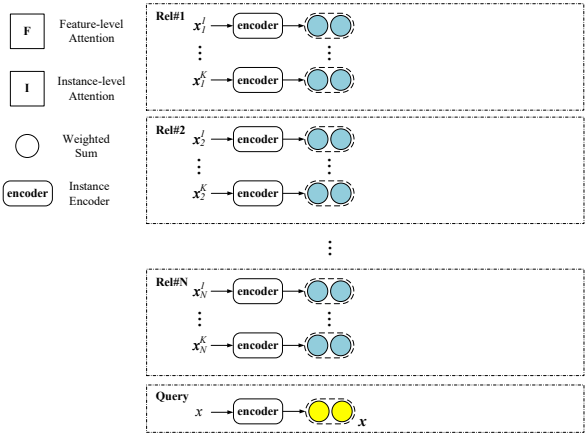
Seek the common parts inside one class.

Hybrid Attention-Based Prototypical Networks

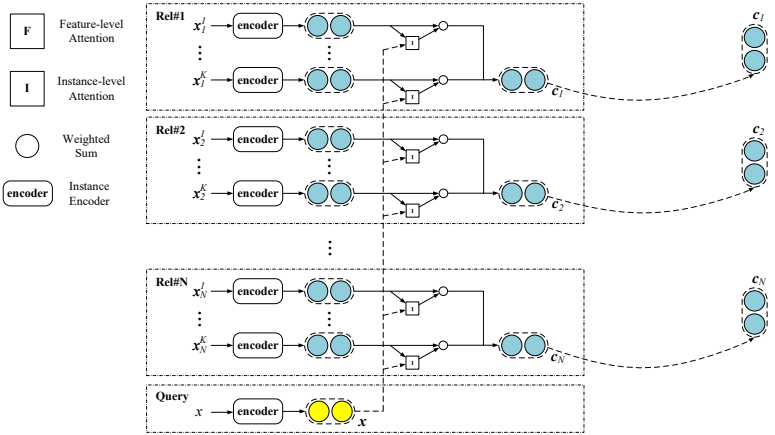
We propose Hybrid Attention-based Prototypical Networks for few-shot relation classification. It adds two components on the original Prototypical Networks:

- Instance-Level Attention
  - Give **supporting instances** that are closer to the query instance higher scores
- Feature-Level Attention
  - Give **dimensions of features** that are more discriminative higher scores

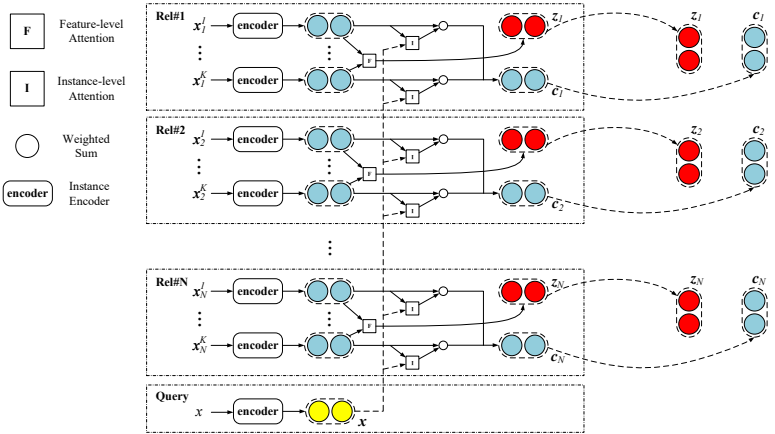
Model Structure - Encoder



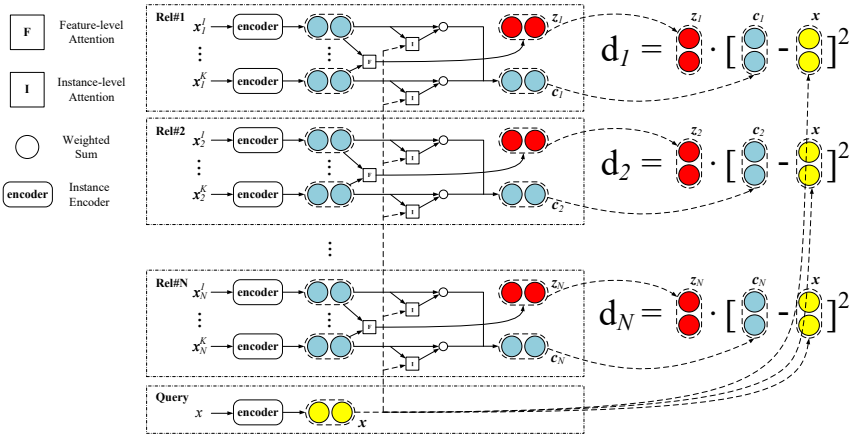
Model Structure - Instance-Level Attention



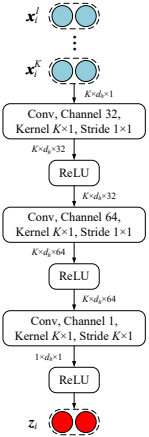
Model Structure - Feature-Level Attention



Model Structure - Calculate Distances



Feature-Level Attention Extractor



Model Detail - Instance-Level Attention

The original Prototypical Networks calculates *prototype* as follows,

$$\mathbf{c}_i = \sum_{j=1}^{n_i} \mathbf{x}_i^j, \tag{1}$$

where $\mathbf{x}_i^j$ is the feature for jth instance of class i ,and $\mathbf{c}_i$ is the *prototype* of class i.

Model Detail - Instance-Level Attention

Instance-Level Attention is actually weighted average of instance features,

$$\mathbf{c}_i = \sum_{j=1}^{n_i} \alpha_j \mathbf{x}_i^j. \tag{2}$$

α_j is defined as follows,

$$\alpha_j = \frac{\exp(e_j)}{\sum_{k=1}^{n_i} \exp(e_k)}, \tag{3}$$

Model Detail - Instance-Level Attention

where

$$e_j = \text{sum}\Big\{\sigma\big(g(\mathbf{x}_i^j) \odot g(\mathbf{x})\big)\Big\}, \tag{4}$$

where $g(\cdot)$ is a linear layer, $\odot$ is element-wise production, $\sigma(\cdot)$ is an activation function and $\text{sum}\{\cdot\}$ means the sum of all elements of the vector. In this paper, we choose tanh for $\sigma(\cdot)$ to produce results among $[-1, 1]$.

Model Detail - Feature-Level Attention

The original Prototypical Networks use **Euclidean Distance** as the metric,

$$d(\mathbf{s}_1, \mathbf{s}_2) = (\mathbf{s}_1 - \mathbf{s}_2)^2, \tag{5}$$

where $\mathbf{s}_1$ and $\mathbf{s}_2$ represent features of two instances and $d(\mathbf{s}_1, \mathbf{s}_2)$ indicates the **distance** between them.

Model Detail - Feature-Level Attention

Feature-Level Attention adds different weights to different dimensions,

$$d(\mathbf{s}_1, \mathbf{s}_2) = \mathbf{z}_i \cdot (\mathbf{s}_1 - \mathbf{s}_2)^2, \tag{6}$$

where z_i is calculated by **Feature-Level Attention Extractor** shown above.

Experiments - Dataset

We train and evaluate our model on FewRel[1], a large-scale few-shot relation classification dataset.

- It has 64 relations for training, 16 relations for validation and 20 relations for test
- There are no overlapping relations between training and test set
- Each relation has 700 instances in FewRel

[1] https://thunlp.github.io/fewrel.html

Experiments - Overall Results

Noise Rate	Model	5 Way 5 Shot	5 Way 10 Shot	10 Way 5 Shot	10 Way 10 Shot
0%	Proto	89.05 ± 0.09	90.79 ± 0.08	81.46 ± 0.13	84.01 ± 0.13
	Proto-HATT	$\mathbf{90.12 \pm 0.04}$	$\mathbf{92.06 \pm 0.06}$	$\mathbf{83.05 \pm 0.05}$	$\mathbf{85.97 \pm 0.08}$
10%	Proto	87.63 ± 0.10	90.15 ± 0.08	79.39 ± 0.14	83.05 ± 0.12
	Proto-HATT	$\mathbf{88.74 \pm 0.06}$	$\mathbf{91.45 \pm 0.05}$	$\mathbf{81.09 \pm 0.08}$	$\mathbf{85.08 \pm 0.07}$
30%	Proto	82.45 ± 0.09	87.64 ± 0.07	72.43 ± 0.12	79.31 ± 0.11
	Proto-HATT	$\mathbf{84.71 \pm 0.07}$	$\mathbf{89.59 \pm 0.05}$	$\mathbf{75.68 \pm 0.11}$	$\mathbf{82.43 \pm 0.07}$
50%	Proto	72.91 ± 0.15	81.71 ± 0.10	61.11 ± 0.17	71.29 ± 0.14
	Proto-HATT	$\mathbf{76.57 \pm 0.07}$	$\mathbf{85.17 \pm 0.09}$	$\mathbf{65.97 \pm 0.11}$	$\mathbf{76.42 \pm 0.13}$

Experiments - Convergence Speed

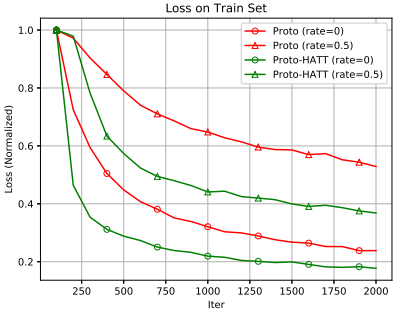


Figure: Loss of different models on the training set.

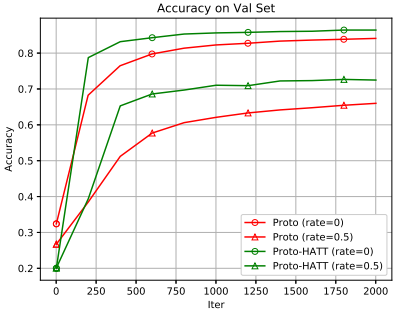


Figure: Acc of different models on the validation set.

Experiments - Case Study of Instance-Level Attention

- **(A) facet of**
  - (1) In 2001, he also published the "Khaki Shadows" that recounted the *military history* of *Pakistan* during the cold war.
  - (2) However, critics have questioned the universal applicability of this model outside of *Singapore*'s communitarian political system and coordinated *urban planning program*.
- **(B) series**
  - **(Highest score)** "*Crying Out Loud*" is the 23rd episode of the 6th season of the American sitcom "*Modern Family*", and the series' 143rd episode overall.
  - (2) The novel is the 4th in Moorcock's four book *The History of the Runestaff* series, and the narrative follows on immediately from the preceding novel "*the Sword of the Dawn*".
- **Query Instance**: The song appeared on the 1st episode of the *4th season* of the American adult animated sitcom "*American Dad!*".

Experiments - Effect of Feature-Level Attention

Comparison between features with different feature-level attention scores. Points with different classes in the right figure are more separable than in the left figure.

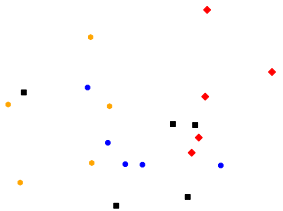


Figure: Features with lower attention score.

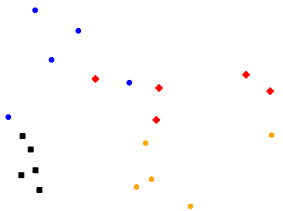


Figure: Features with higher attention score.

Experiments - Hybrid Attention Improves Encoder Capacity

Comparison between instance embeddings trained with or without hybrid attention. Points in the right figure are easier to classify while those in left just lump together.

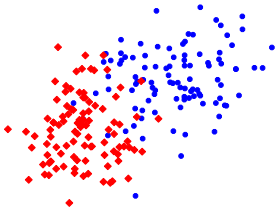


Figure: Embeddings trained without hybrid attention.

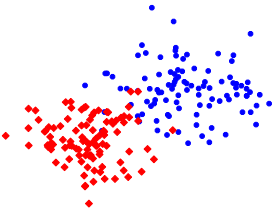


Figure: Embeddings trained with hybrid attention.

Conclusion

- We propose hybrid attention-based Prototypical Networks for noisy few-shot relation classification task.
- Hybrid attention contains two modules
  - Instance-level attention, which highlights those query-related instances
  - Feature-level attention, which alleviates the problem of feature sparsity
- It not only achieves the state-of-the-art results and performs better in noisy data, but also converges a lot faster when training.

Code and Dataset

- Our code: https://github.com/thunlp/fewrel
- FewRel Dataset: https://thunlp.github.io/fewrel.html

Thanks!