KEML: A Knowledge-Enriched Meta-Learning Framework for Lexical Relation Classification

Chengyu Wang[1], Minghui Qiu[1], Jun Huang[1], Xiaofeng He[2]

[1] Alibaba Group [2] East China Normal University

Introduction

✓Lexical Relation Classification

- Task: Classifying a word pair into a finite set of relation types (e.g., synonymy, antonymy)

Relation	Tag	Template	Example
Synonymy	SYN	W2 can be used with the same meaning as W1	*candy-sweet, apartment-flat*
Antonymy	ANT	W2 can be used as the opposite of W1	*clean-dirty, add-take*
Hypernymy	HYPER	W1 is a kind of W2	*cannabis-plant, actress-human*
Part-whole meronymy	PART_OF	W1 is a part of W2	*calf-leg, aisle-store*
Random word	RANDOM	None of the above relations apply	*accident-fish, actor-mild*

CogALex-V shared task

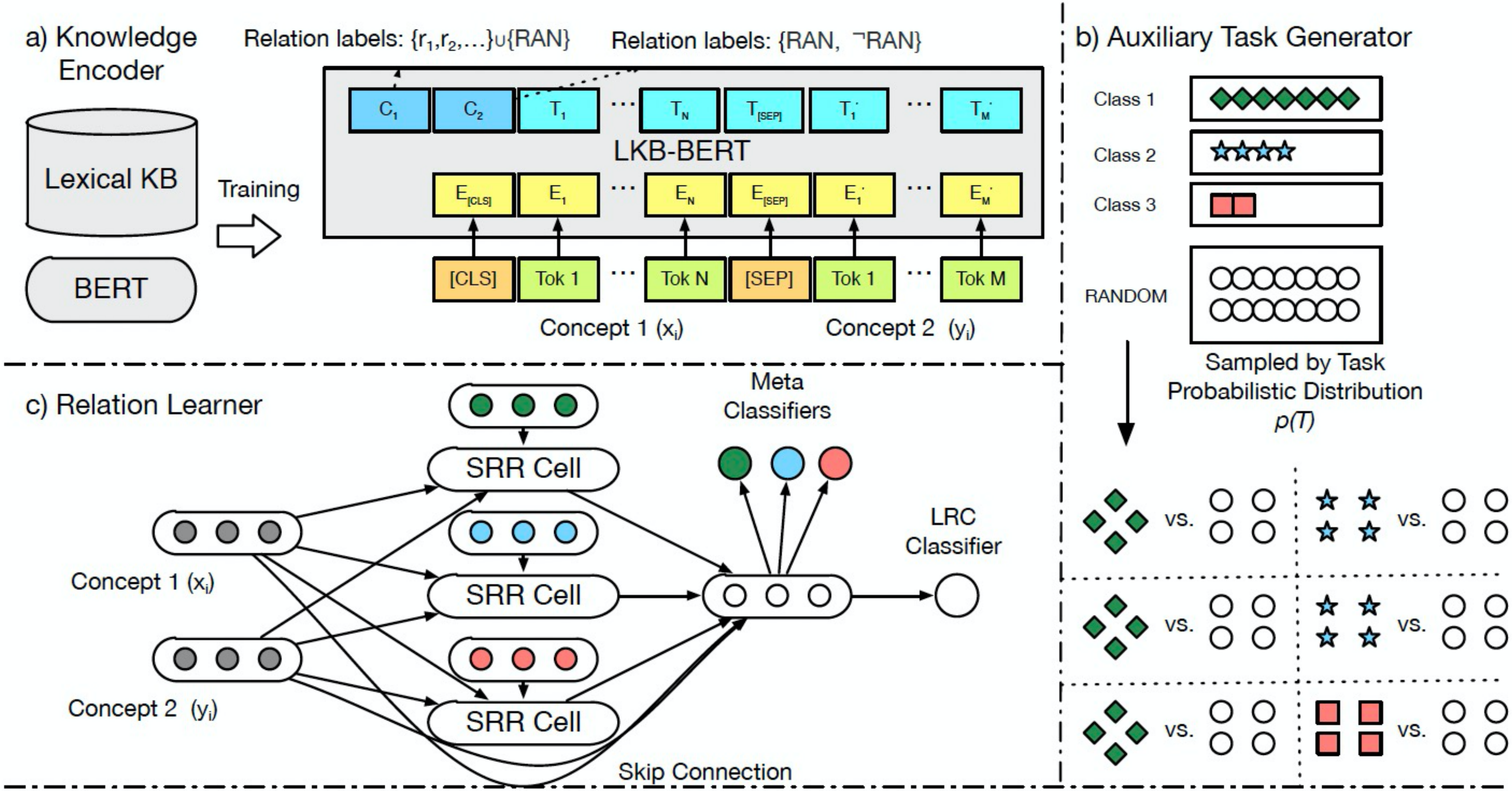
Introduction

✓Existing Approaches

- Path-based approaches: use dependency paths connecting two terms to infer lexical relations
  - “Low coverage” problem

- Distributional approaches: consider the global contexts of terms to predict lexical relations using word embeddings
  - “Lexical memorization” problem

KEML: Our Solution



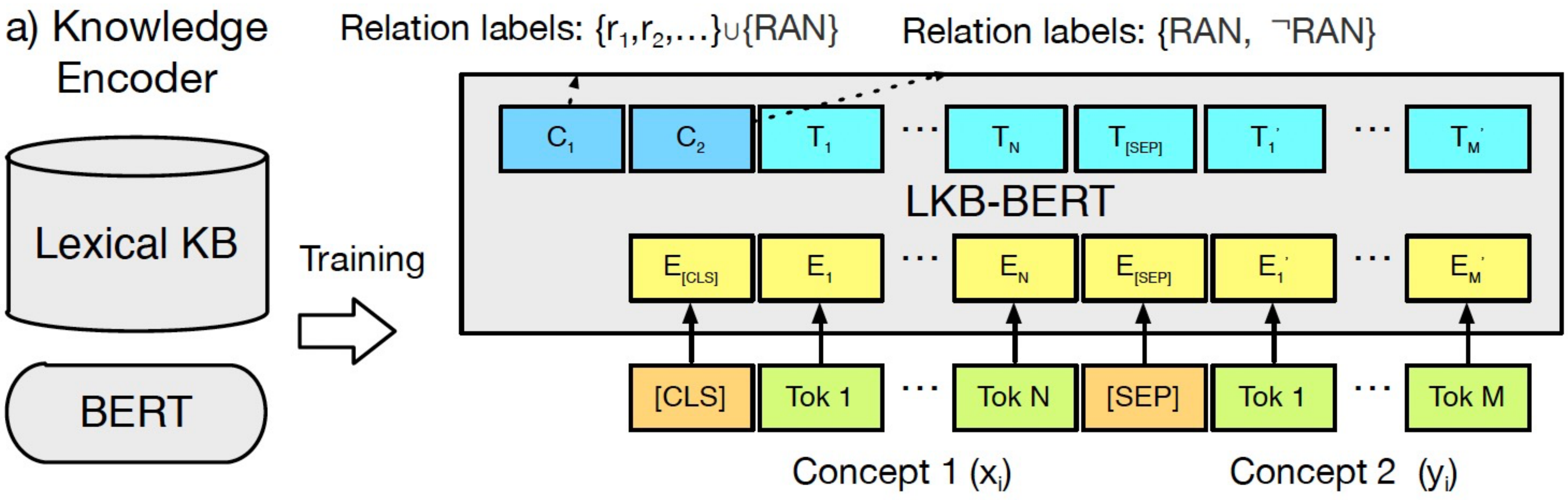
Abbreviations

KEML: Knowledge-Enriched Meta-Learning
Lexical KB: Lexical Knowledge Base
LKB-BERT: Lexical KB-BERT
SRR Cell: Single Relation Recognition Cell

Knowledge Encoder

✓Integrating lexical relations into BERT

- Task 1: Classifying concept pairs into multiple relation types.
- Task 2: Classifying concept pairs into RANDOM or non-RANDOM.



LKB-BERT

Auxiliary Task Generator

✓Goal

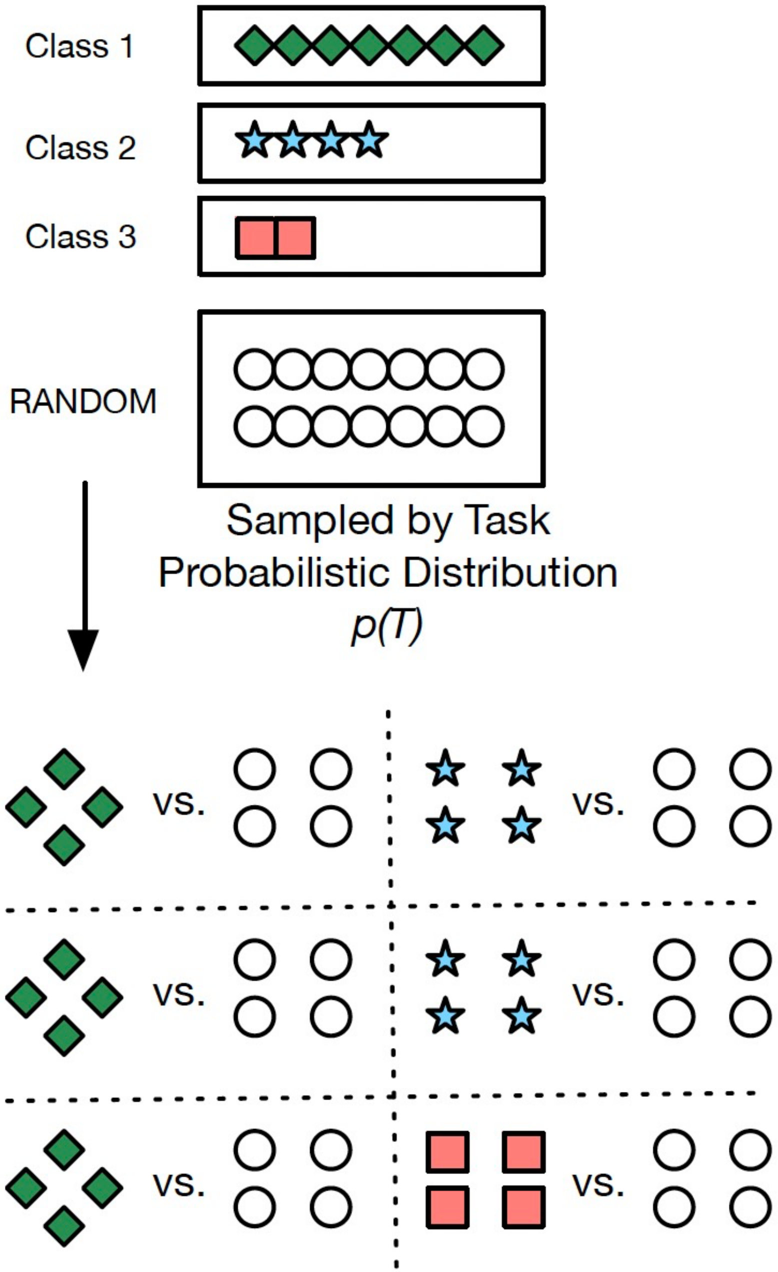
- Enabling the neural network to recognize a specific type of lexical relation

✓Learning Objective

$$\mathcal{L}(\mathcal{T}_r) = -\sum_{(x_i,y_i,r_i)\in\mathcal{S}_r\cup\mathcal{S}_{\mathbf{RAN}}} (\mathbf{1}(r_i = r) \cdot \log q_r(x_i, y_i) + \mathbf{1}(r_i = \mathbf{RAN}) \cdot \log q_{\mathbf{RAN}}(x_i, y_i))$$

✓Task Distribution

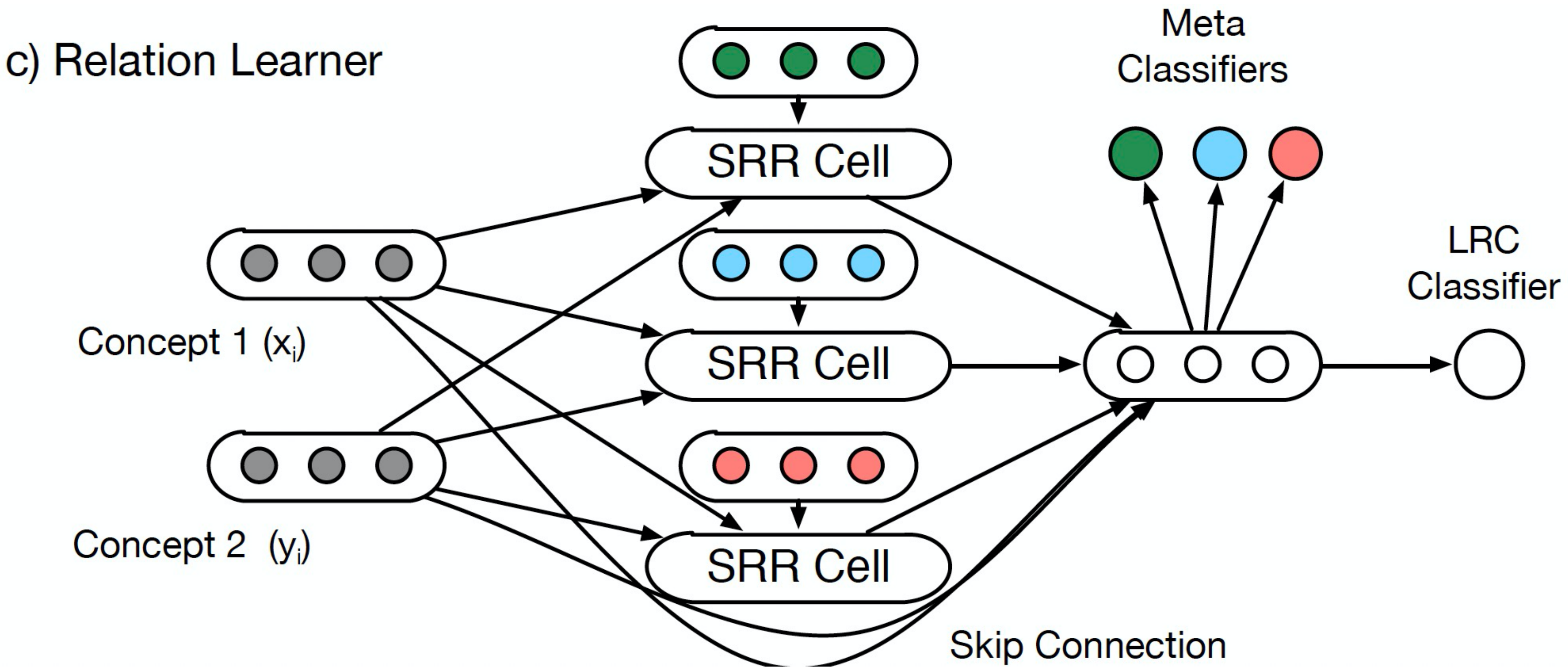
$$p(\mathcal{T}_r) = \frac{\ln|\mathcal{D}_r| + \gamma}{\sum_{r'\in\mathcal{R}\setminus\{\mathbf{RAN}\}}(\ln|\mathcal{D}_{r'}| + \gamma)}$$



Relation Leaner

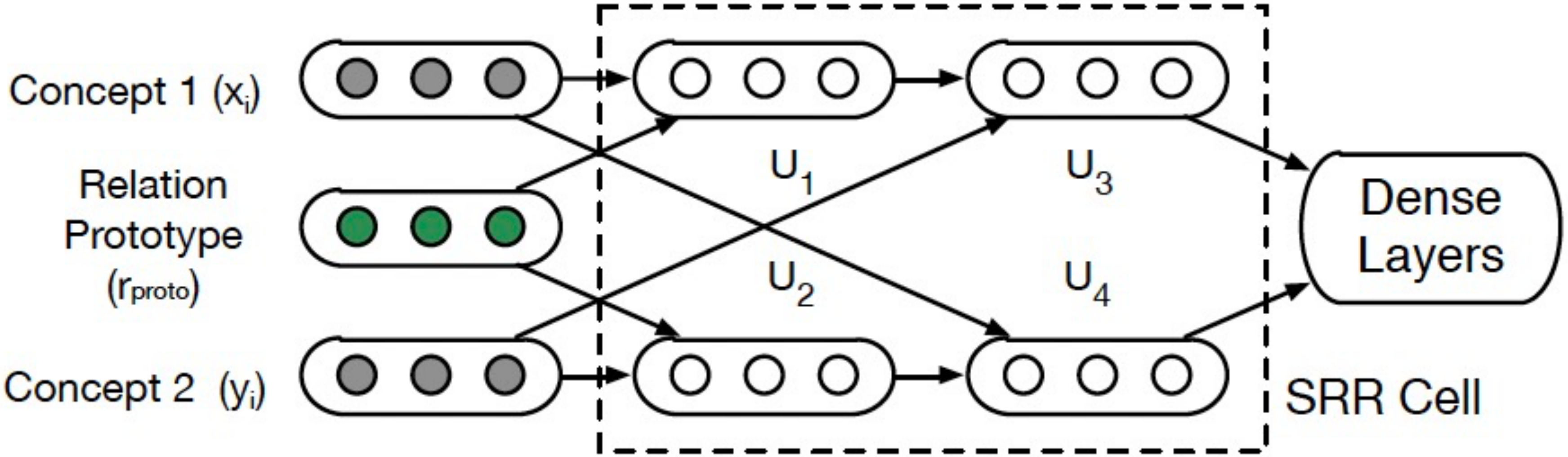
✓Design of the Neural Network

- For each type of lexical relation, use an SRR Cell to recognize such relations



Relation Leaner

✓Design of the SRR (Single Relation Recognition) Cell



- U1, U2: Inferring the embeddings of relation objects or subjects
- U3, U4: Predicting the existence of the lexical relation

$$\vec{U}_1 = \tanh((\vec{x}_i \oplus \vec{r}_{proto}) \cdot \mathbf{W}_1 + \vec{b}_1) \qquad \vec{U}_3 = \tanh((\vec{U}_1 - \vec{y}_i) \cdot \mathbf{W}_3 + \vec{b}_3)$$

$$\vec{U}_2 = \tanh((\vec{y}_i \oplus \vec{r}_{proto}) \cdot \mathbf{W}_2 + \vec{b}_2) \qquad \vec{U}_4 = \tanh((\vec{U}_2 - \vec{x}_i) \cdot \mathbf{W}_4 + \vec{b}_4)$$

Relation Leaner

✓Meta-learning Algorithm for LRC

Algorithm 1 Meta-Learning Algorithm for LRC

Initialize model parameters θ;
while not converge **do**
    Sample N auxiliary tasks $\mathcal{T}_{r_1}, \mathcal{T}_{r_2}, \cdots, \mathcal{T}_{r_N}$ from the task distribution $p(\mathcal{T})$;
    for each auxiliary task $\mathcal{T}_r$ **do**
        Sample a batch (positive samples $\mathcal{S}_r$ and negative samples $\mathcal{S}_{\text{RAN}}$) from the training set $\mathcal{D}$;
        Update adapted parameters: $\theta_r \leftarrow \theta - \alpha \nabla \mathcal{L}(\mathcal{T}_r)$ based on $\mathcal{S}_r$ and $\mathcal{S}_{\text{RAN}}$;
    end for
    Update meta-parameters: $\theta \leftarrow \theta - \epsilon \nabla \sum_{\mathcal{T}_r} \mathcal{L}(\mathcal{T}_r)$;
end while
Fine-tune θ over $\mathcal{D}$ by standard supervised learning LRC;

Experiments

✓LRC results over four benchmark datasets

- Pre-trained model: BERT
- Lexical KB: Subset of WordNet

Method	K&H+N			BLESS			ROOT09			EVALution		
	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1	Pre	Rec	F1
Concat	0.909	0.906	0.904	0.811	0.812	0.811	0.636	0.675	0.646	0.531	0.544	0.525
Diff	0.888	0.886	0.885	0.801	0.803	0.802	0.627	0.655	0.638	0.521	0.531	0.528
NPB	0.713	0.604	0.55	0.759	0.756	0.755	0.788	0.789	0.788	0.53	0.537	0.503
NPB+Aug	-	-	0.897	-	-	0.842	-	-	0.778	-	-	0.489
LexNET	0.985	0.986	0.985	0.894	0.893	0.893	0.813	0.814	0.813	0.601	0.607	0.6
LexNET+Aug	-	-	0.970	-	-	0.927	-	-	0.806	-	-	0.545
SphereRE	0.990	0.989	0.990	0.938	0.938	0.938	0.860	0.862	0.861	0.62	0.621	0.62
LKB-BERT	0.981	0.982	0.981	**0.939**	0.936	0.937	**0.863**	**0.864**	**0.863**	**0.638**	**0.645**	**0.639**
KEML-S	0.984	0.983	0.984	**0.942**	**0.940**	**0.941**	**0.877**	**0.871**	**0.873**	**0.649**	**0.651**	**0.644**
KEML	**0.993**	**0.993**	**0.993**	**0.944**	**0.943**	**0.944**	**0.878**	**0.877**	**0.878**	**0.663**	**0.660**	**0.660**

Experiments

✓How Lexical KB Helps the Learning Process?

- Binary: only binary classification
- Multi: only lexical relation classification
- Full: full implementation

Dataset	Binary	Multi	Full
K&H+N	0.964	0.972	**0.983**
BLESS	0.921	0.929	**0.939**
ROOT09	0.854	0.861	**0.863**
EVALution	0.630	0.632	**0.641**
CogALex-V	0.464	0.467	**0.472**

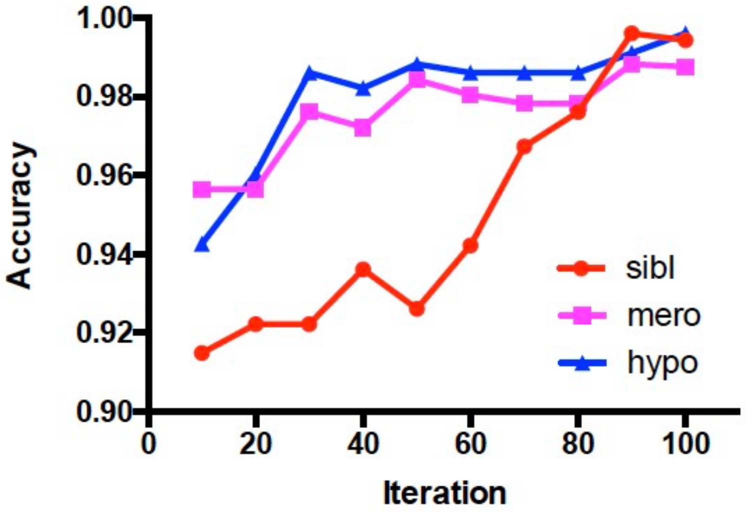
✓How KEML Deals with Each Type of Relations?

- CogALex-V shared task

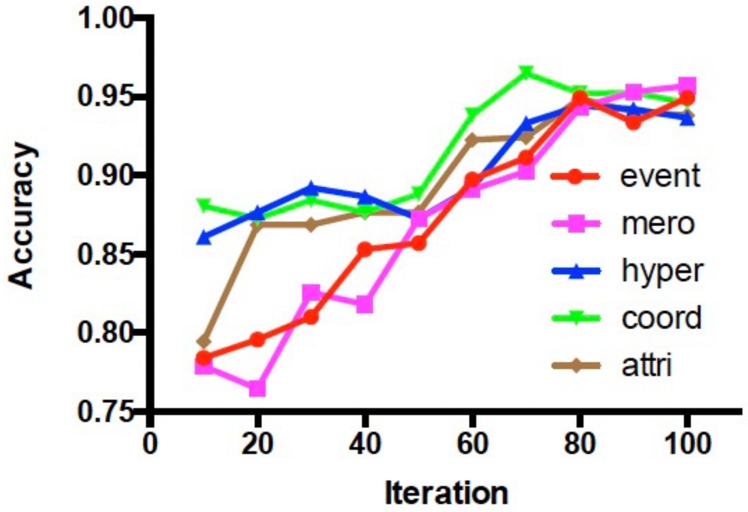
Method	SYN	ANT	HYP	MER	All
GHHH	0.204	0.448	0.491	0.497	0.423
LexNET	0.297	0.425	0.526	0.493	0.445
STM	0.221	**0.504**	0.498	0.504	0.453
SphereRE	0.286	0.479	0.538	0.539	0.471
LKB-BERT	0.281	0.470	0.532	0.530	0.464
KEML-S	0.276	0.470	**0.542**	**0.631**	**0.485**
KEML	**0.292**	0.492	**0.547**	**0.652**	**0.500**

Experiments

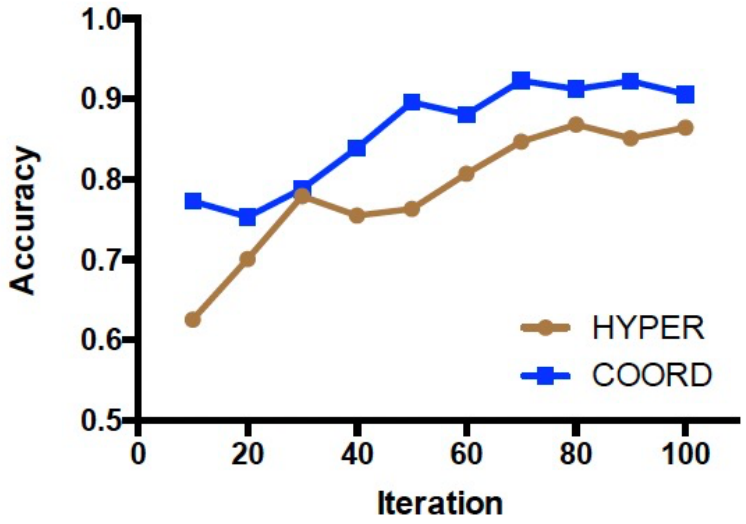
✓How the Meta-learning Process Helps the Learning Process?



(a) Dataset: K&H+N



(b) Dataset: BLESS



(c) Dataset: ROOT09

Experiments

✓Error Analysis: it is still difficult to distinguish some “blurry” lexical relations.

Dataset: ROOT09

Prediction↓ True→	Co-hyponym	Hypernym	Random
Co-hyponym	*83.8%*	**8.2%**	**7.2%**
Hypernym	10.2%	*86.5%*	2.4%
Random	**6.0%**	5.3%	*90.4%*

Dataset: K+H&N

Prediction↓ True→	Co-hyponym	Hypernym	Meronym	Random
Co-hyponym	*99.4%*	**1.8%**	**1.0%**	0.2%
Hypernym	0.2%	*97.5%*	0.3%	0.1%
Meronym	0.1%	0.2%	*96.5%*	0.1%
Random	0.3%	0.5%	**2.2%**	*99.6%*

Conclusion

✓We present the KEML framework for lexical relation classification.

✓Experiments show that KEML achieves SOTA results.

✓Future work includes:

- Improving relation representation learning with deep neural language models
- Integrating richer linguistic and commonsense knowledge into KEML
- Applying KEML to downstream tasks such as taxonomy learning

THANKS

--------- Q&A Section --------