RESIDE: IMPROVING DISTANTLY-SUPERVISED NEURAL RELATION EXTRACTION USING SIDE INFORMATION

Shikhar Vashishth[1] Rishabh Joshi[2] * Sai Suman Prayaga[1]

Chiranjib Bhattacharyya[1] Partha Talukdar[1]

1 Indian Institute of Science
2 Birla Institute of Technology and Science, Pilani
{shikhar,chiru,ppt}@iisc.ac.in
f2014102@pilani.bits-pilani.ac.in, suman.sai14@gmail.com

BACKGROUND

- Relation Extraction
- Distant Supervision (DS)
  - Multi-instance Multi-label (MIML)
- Neural Relation Extraction:
  - PCNN: Piecewise Convolution NN
  - PCNN + Attention
- Side Information in RE
- Graph Convolution Networks (GCN)

RELATION EXTRACTION

- Example: Google was founded in the state of California in 1998.
  - Founding-year (Google, 1998)
  - Founding-location(Google, California)
- Traditional RE
  - Hand-built patterns
  - Supervised approaches:
    - Relation detection
    - Relation classification

*Lack of annotated data

DISTANT SUPERVISION(DS)

- Alleviates the problem of lack of annotated data
- Assumption:
  - “If two entities have a relationship in a KB, then all sentences mentioning the entities express the same relation

*Noisy labelled data

MULTI-INSTANCE MULTI-LABEL (MIML)

- DS might lead to noisy labelled data
- MIML: Relaxed DS assumption
  - Allow multiple relations to hold between entities
  - If a relation holds between entities then at least one sentence must support it

NEURAL NETWORKS FOR DS

- **Piecewise Convolution NN**
  - **Adapt CNNs for extracting sentence features**
- **PCNN + Attention**
  - **Attention mechanism**

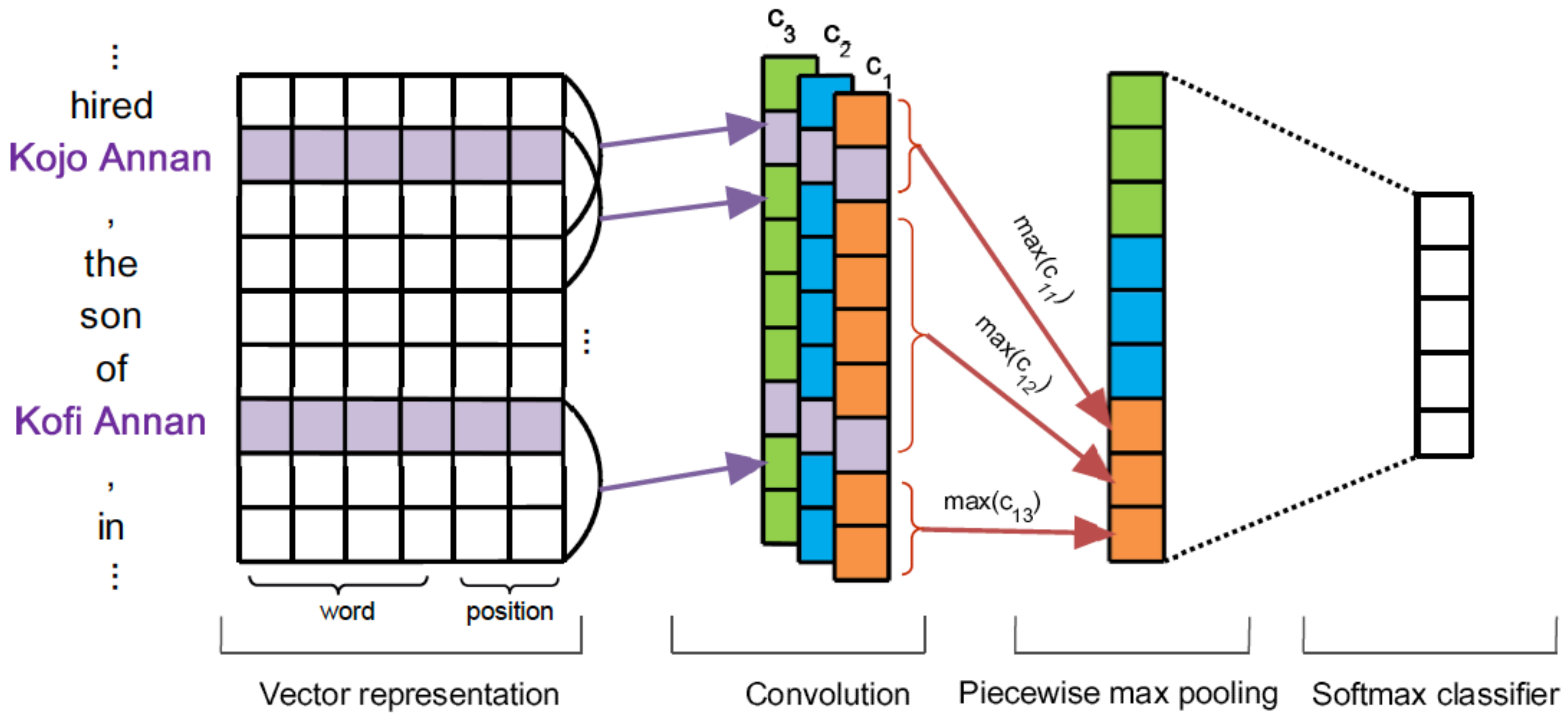


Figure 3: The architecture of PCNNs (better viewed in color) used for distant supervised relation extraction, illustrating the procedure for handling one instance of a bag and predicting the relation between *Kojo Annan* and *Kofi Annan*.

RESIDE

Given a bag of sentences (or instances) {s1, s2, ...sn} for a given entity pair

Task: predict the relation between them

- **Syntactic Sentence Encoding:**
  - Bi-GRU over the concatenated positional and word embedding
  - GCN over dependency tree, appended to the representation
  - attention over tokens is used to subdue irrelevant tokens and get an embedding for the entire sentence
- **Side Information Acquisition**
  - additional supervision from KBs
  - Open IE
- **Instance Set Aggregation**
  - sentence representation from syntactic sentence encoder is concatenated with the matched relation embedding

RESIDE OVERVIEW

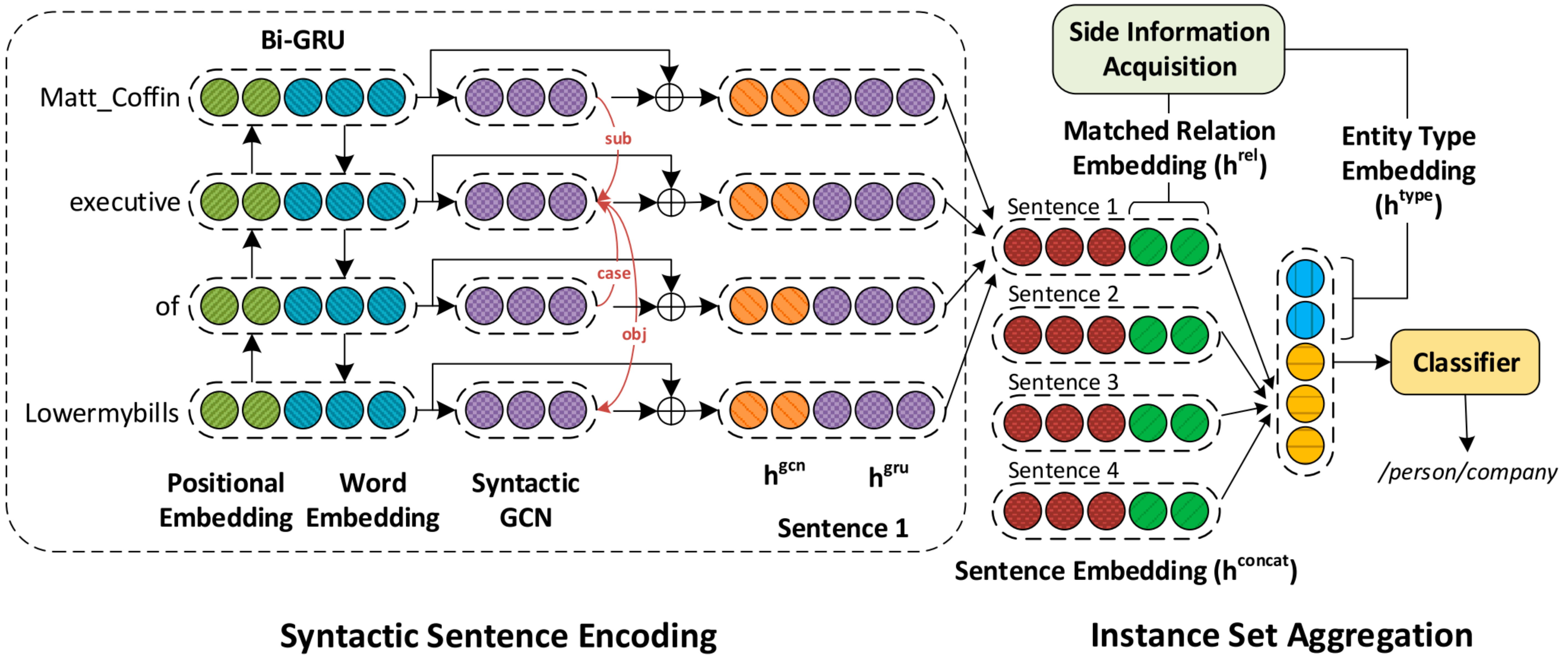


Figure 1: Overview of RESIDE. RESIDE first encodes each sentence in the bag by concatenating embeddings (denoted by ⊕) from Bi-GRU and Syntactic GCN for each token, followed by word attention. Then, sentence embedding is concatenated with relation alias information, which comes from the Side Information Acquisition Section (Figure 2), before computing attention over sentences. Finally, bag representation with entity type information is fed to a softmax classifier. Please see Section 5 for more details.

- **KGs contain information ->improve RE**
- **Dependency tree based features -> RE (GCN)**

GCN

Directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
$\mathcal{V}$: the set of vertices
$\mathcal{E}$: the set of edges
(u, v, l_{uv}): node u, node v, label l_{uv}

updated edge set : $\mathcal{E}'$
- inverse edges set: (v, u, l_{uv}^{-1})
- selfloops set: $(u, u, \top)$, self-loops: $\top$
- original edge set $\mathcal{E}$

For each node v in $\mathcal{G}$
initial representation: $x_v \in \mathbb{R}^d, \forall v \in \mathcal{V}$.
d -dimensional hidden representation: $h_v \in \mathbb{R}^d, \forall v \in \mathcal{V}$

label dependent model parameters: $W_{l_{uv}} \in \mathbb{R}^{d \times d}$ and $b_{l_{uv}} \in \mathbb{R}^d$
the set of neighbors of v based on $\mathcal{E}'$: $\mathcal{N}(v)$
non-linear activation function: f

$$h_v = f\left(\sum_{u \in \mathcal{N}(v)} \left(W_{l_{uv}} x_u + b_{l_{uv}} \right) \right)$$

Hidden representation of node v after k^{th} GCN layer:

$$h_v^{k+1} = f\left(\sum_{u \in \mathcal{N}(v)} \left(W_{l_{uv}}^k h_u^k + b_{l_{uv}}^k \right) \right)$$

GATED GCN

At k^{th} layer, the importance of an edge (u, v, l_{uv}) is computed as:

$$g_{uv}^k = \sigma\left(h_u^k \cdot \hat{w}_{l_{uv}}^k + \hat{b}_{l_{uv}}^k\right)$$

parameters: $\hat{w}_{l_{uv}}^k \in \mathbb{R}^m$, $\hat{b}_{l_{uv}}^k \in \mathbb{R}$
$\sigma(\cdot)$: sigmoid function.
With edgewise gating, the final GCN embedding for a node v after k^{th} layer:

$$h_v^{k+1} = f\left(\sum_{u \in \mathcal{N}(v)} g_{uv}^k \times \left(W_{l_{uv}}^k h_u^k + b_{l_{uv}}^k\right)\right)$$

SYNTACTIC SENTENCE ENCODING

For each sentence in the bag si with m tokens {w1, w2, ...wm}

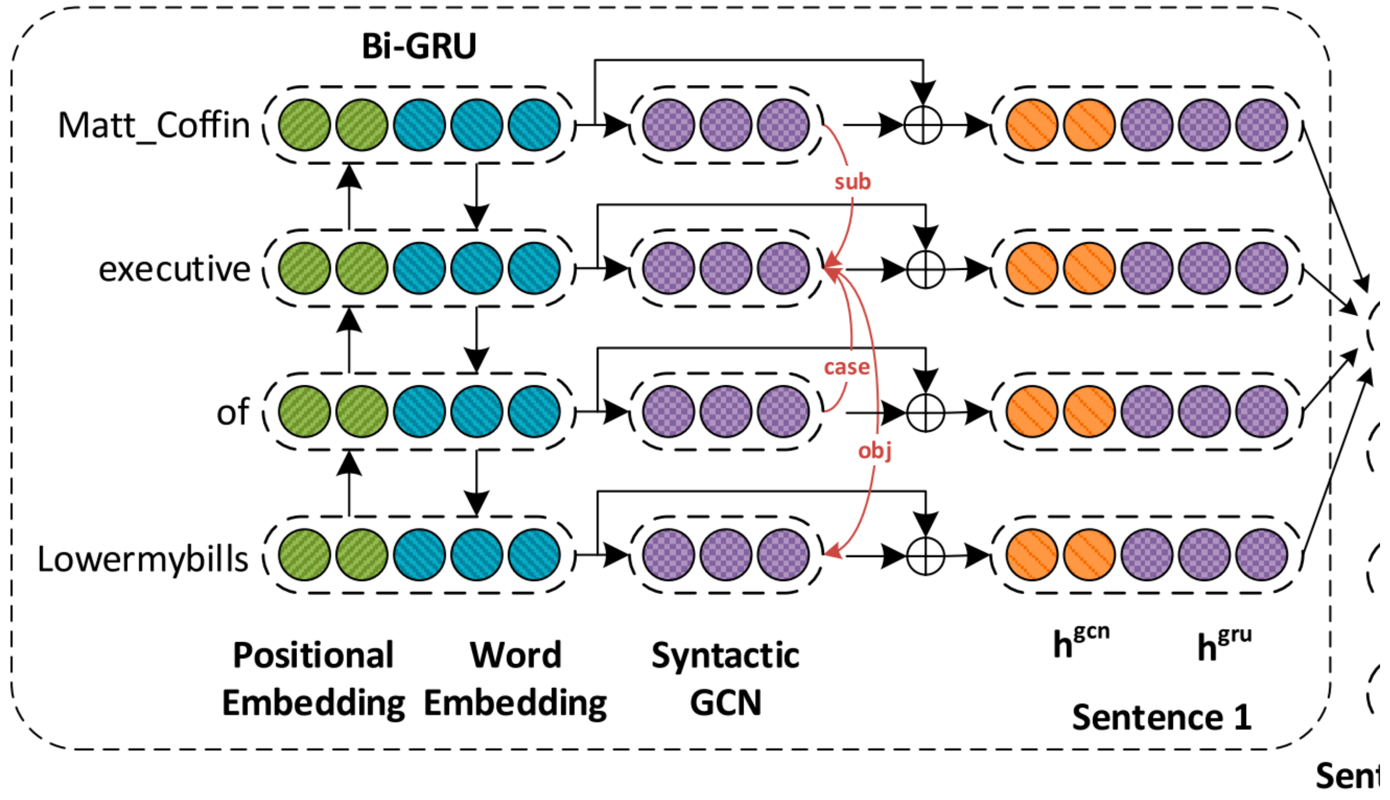
- represent each token: k-dimensional GloVe embedding
- relative position of tokens: p-dimensional position embeddings

Concatenate

sentence representation: $\mathcal{H} \in \mathbb{R}^{m \times (k+2p)}$

Bi-GRU

$\mathscr{H}^{gru} \in \mathbb{R}^{m \times d_{gru}}$



Syntactic Sentence Encoding

Dependency tree (Stanford CoreNLP) +GCN

edge: (u, v, l_{uv})
L_{uv} : edge label

$$L_{uv} = \begin{cases} \rightarrow & \text{if edge exists in dependency parse} \\ \leftarrow & \text{if edge is an inverse edge} \\ \top & \text{if edge is a self-loop} \end{cases}$$

For each token w_i, GCN embedding $h^{gcn}_{i_{k+1}} \in \mathbb{R}^{d_{gcn}}$ after k^{th} layer is defined as:

$$h^{gcn}_{i_{k+1}} = f\left(\sum_{u \in \mathcal{N}(i)} g^k_{iu} \times \left(W^k_{L_{iu}} h^{gcn}_{u_k} + b^k_{L_{iu}} \right) \right)$$

g^k_{iu}: edgewise gating
L_{iu}: edge label
f: ReLU
h^{concat}_i as $\left[h^{gru}_i ; h^{gcn}_{i^{k+1}} \right]$

SYNTACTIC SENTENCE ENCODING

For token w_i in the sentence, attention weight α_i: taking softmax over $\{\ u_i\}$

$$\alpha_i = \frac{\exp(u_i)}{\sum_{j=1}^{m} \exp(u_j)} \text{ where, } u_i = h_i^{\text{concat}} \cdot r$$

r: random query vector
u_i: relevance score assigned to each token
The representation of a sentence is given as a weighted sum of its tokens:

$$s = \sum_{j=1}^{m} \alpha_i h_i^{\text{concat}}$$

SIDE INFORMATION ACQUISITION

- **Relation Alias Side Information**
  - **Syntactic Context Extractor(Stanford Open IE and dependency parse):**

    $\mathcal{P}$: extracting relation phrases between target entities

    $|\mathcal{P}| > 1$: might get multiple matched reations -> take average

  - **Paraphrase Database (PPDB)**

    extended set of relation aliases $\mathcal{R}$

  - **matched relation embedding (Closest)**

    matched relation embedding (h^{rel})
    matching $\mathcal{P}$ with $\mathcal{R}$

    - d -dimensional space using GloVe embeddings
    - cosine distance
    - threshold on cosine distance to remove noisy aliases.

- **Entity Type Side Information**
  - **All relations are constrained by entity types**

    entity type embedding (h^{type})

  - eg. subject and object
  - (*person/place of birth* can only occur between *a person* and *a location)*
  - *KGs: Freebase, Wikidata*
  - *Not suitable as hard constraints*

SIDE INFORMATION ACQUISITION

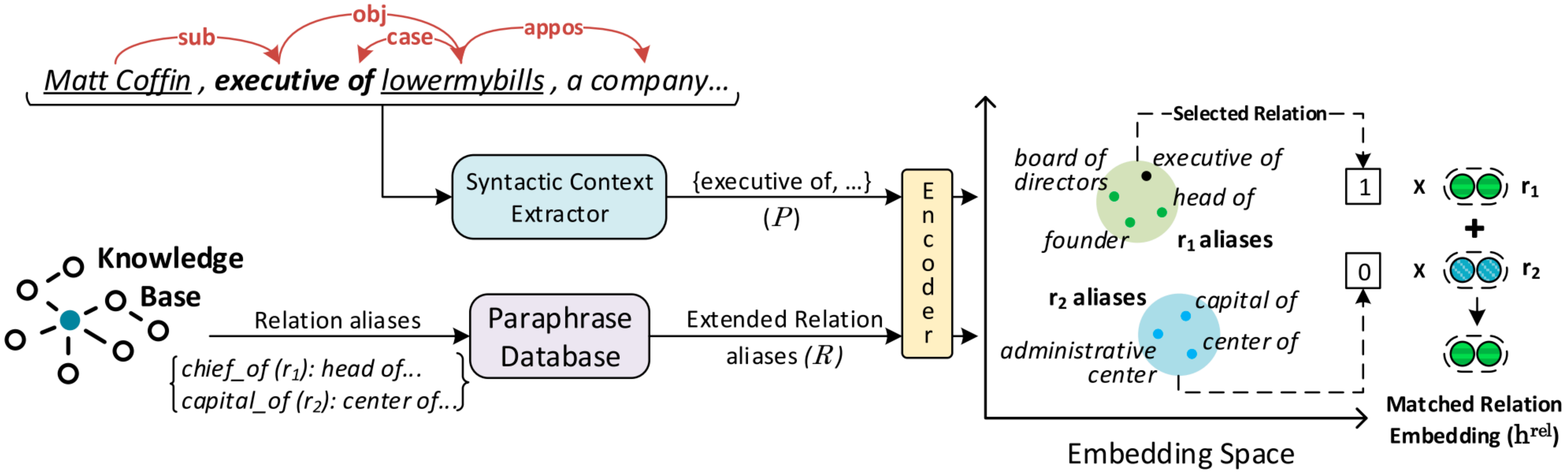
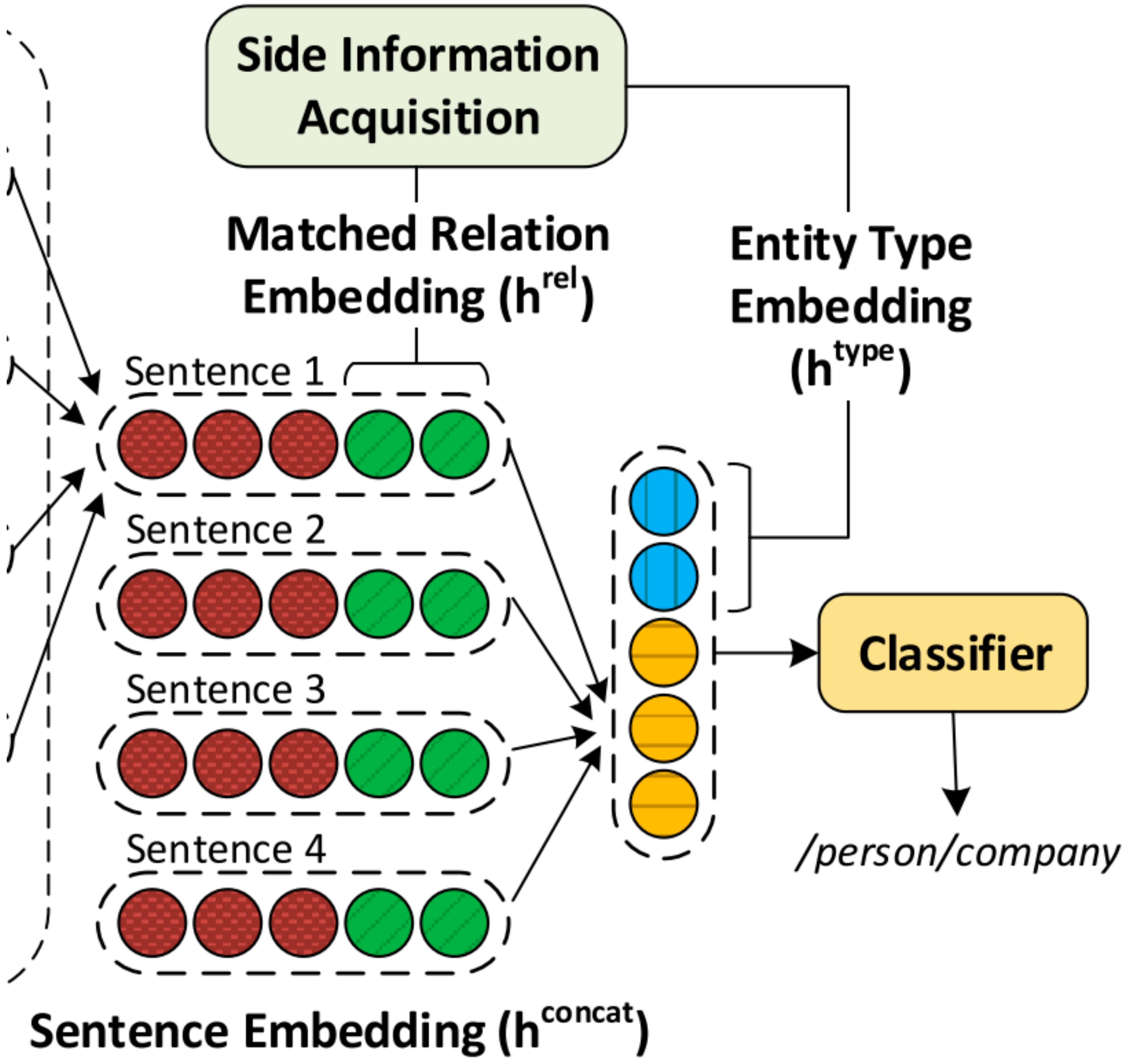


Figure 2: Relation alias side information extraction for a given sentence. First, Syntactic Context Extractor identifies relevant relation phrases $\mathcal{P}$ between target entities. They are then matched in the embedding space with the extended set of relation aliases $\mathcal{R}$ from KB. Finally, the relation embedding corresponding to the closest alias is taken as relation alias information. Please refer Section 5.2.

INSTANCE SET AGGREGATION



Instance Set Aggregation

sentence representation si

matched relation embedding $\left(h^{rel}\right)$

Concat

Sentence embedding $\left(h^{concat}\right)$

The attention score α_i for i^{th} sentence is formulated as:

$$\alpha_i = \frac{\exp(\hat{s}_i \cdot q)}{\sum_{j=1}^{n} \exp(\hat{s}_j \cdot q)} \text{ where, } \hat{s}_i = \left[s_i; h_i^{rel}\right].$$

Bag representation $\mathscr{B}$

$$\hat{\mathscr{B}} = \left[\mathscr{B}; h_{\text{sub}}^{\text{type}}; h_{obj}^{\text{type}}\right] \text{ where, } \mathscr{B} = \sum_{i=1}^{n} \alpha_i \hat{s}_i .$$

$$p(y) = \text{Softmax}(W \cdot \hat{\mathscr{B}} + b)$$

EXPERIMENTS DATASET

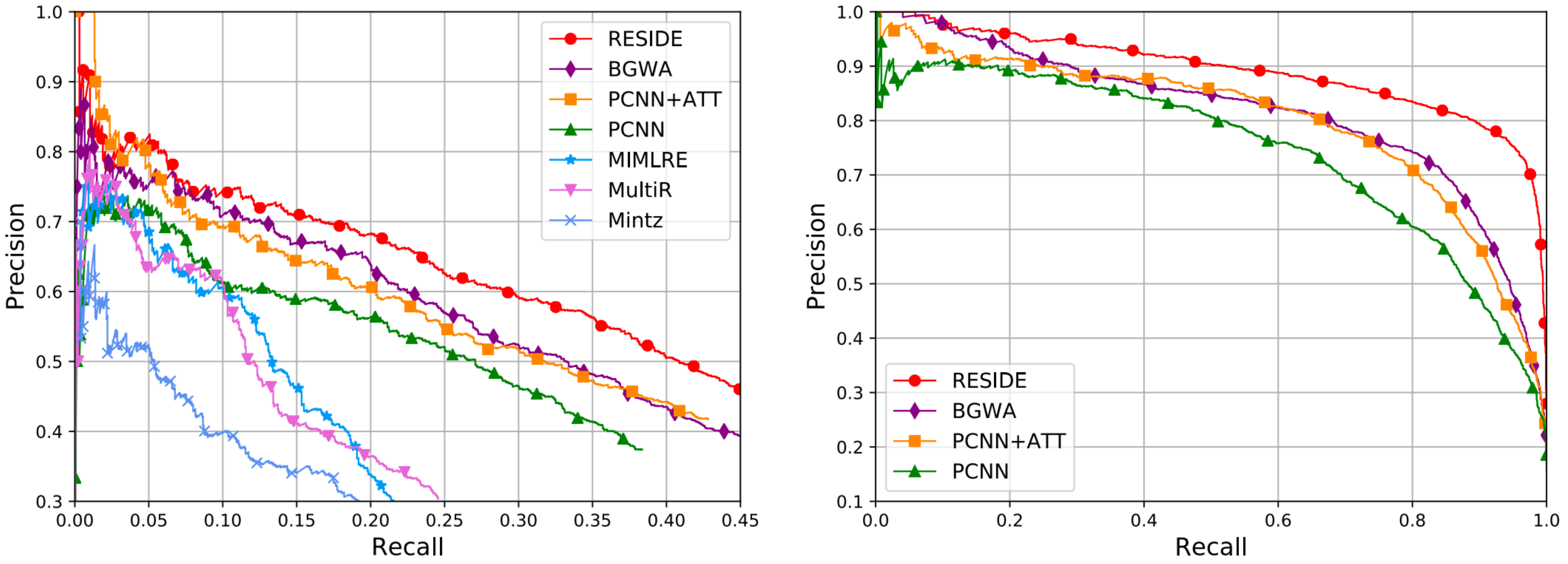
- Riedel
- GIDS

Datasets	Split	# Sentences	# Entity-pairs
Riedel (# Relations: 53)	Train	455,771	233,064
	Valid	114,317	58,635
	Test	172,448	96,678
GDS (# Relations: 5)	Train	11,297	6,498
	Valid	1,864	1,082
	Test	5,663	3,247

EXPERIMENT BASELINE

- Mintz: Multi-class logistic regression model proposed by (Mintz et al., 2009) for distant supervision paradigm.
- • MultiR: Probabilistic graphical model for multi instance learning by (Hoffmann et al., 2011)
- • MIMLRE: A graphical model which jointly models multiple instances and multiple labels. More details in (Surdeanu et al., 2012).
- • PCNN: A CNN based relation extraction model by (Zeng et al., 2015) which uses piecewise max-pooling for sentence representation.
- • PCNN+ATT: A piecewise max-pooling over CNN based model which is used by (Lin et al., 2016) to get sentence representation followed by attention over sentences.
- • BGWA: Bi-GRU based relation extraction model with word and sentence level attention (Jat et al., 2018).
- • RESIDE: The method proposed in this paper, please

EXPERIMENTS



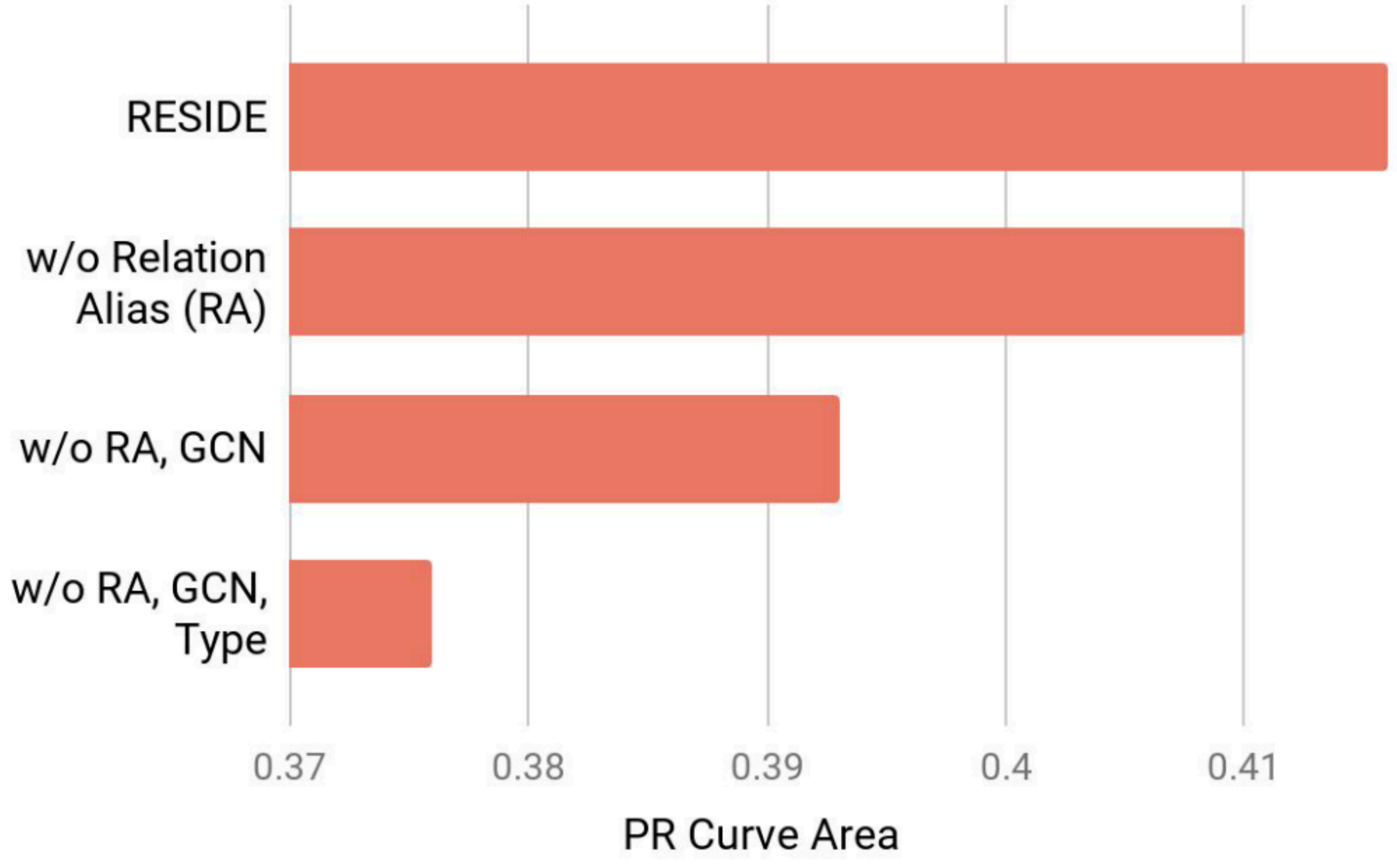
(a) Riedel dataset

(b) GIDS dataset

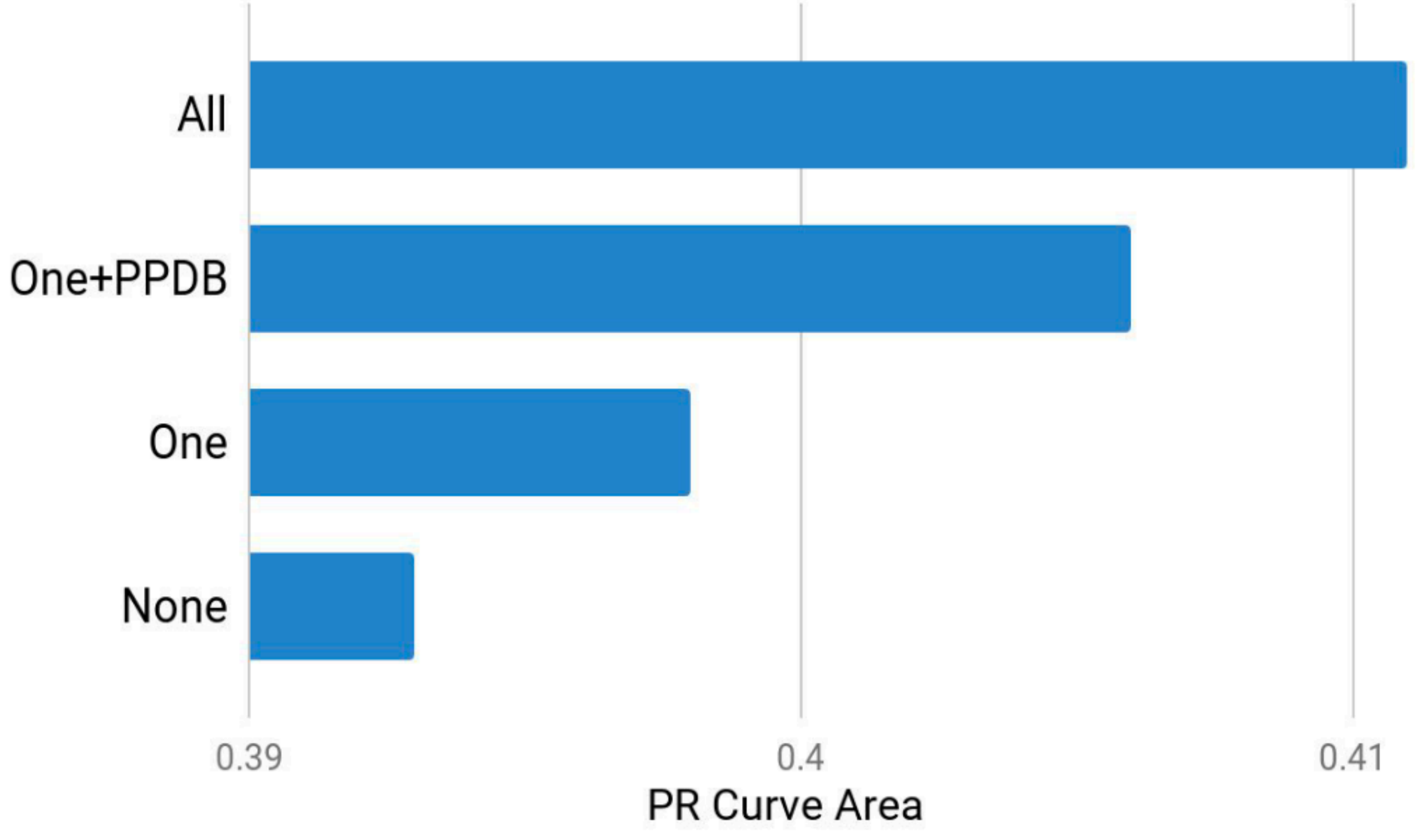
Figure 3: Comparison of Precision-recall curve. RESIDE achieves higher precision over the entire range of recall than all the baselines on both datasets. Please refer Section 7.1 for more details.

ABLATION RESULTS

Ablation Results



Effect of Relation Alias Side Information



REFERENCES

- *Mike Mintz, Steven Bills, Rion Snow, and Dan Juraf- sky. 2009. Distant supervision for relation extrac- tion without labeled data. In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Vol- ume 2-Volume 2, pages 1003–1011. Association for Computational Linguistics.*
- *Jeffrey Pennington, Richard Socher, and Christo- pher D. Manning. 2014. Glove: Global vectors for word representation. In Empirical Methods in Nat- ural Language Processing (EMNLP), pages 1532– 1543.*
- *Daojian Zeng, Kang Liu, Yubo Chen, and Jun Zhao. 2015. Distant supervision for relation extraction via piecewise convolutional neural networks. In Pro-ceedings ofthe 2015 Conference on Empirical Meth- ods in Natural Language Processing, pages 1753– 1762.*
- *Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. 2012. Multi-instance multi-label learning for relation extraction. In Pro-ceedings of the 2012 joint conference on empirical methods in natural language processing and compu- tational natural language learning, pages 455–465. Association for Computational Linguistics.*
- *Thomas N. Kipf and Max Welling. 2017. Semi- supervised classification with graph convolutional networks. In International Conference on Learning Representations (ICLR).*