The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Mention and Entity Description Co-Attention for Entity Disambiguation

Feng Nie,[*1] **Yunbo Cao,**[*2] **Jinpeng Wang,**[3] **Chin-Yew Lin,**[3] **Rong Pan**[*1]
[1]Sun-Yat-Sen University [2]Tencent Corporation, Beijing, China [3]Microsoft Research Asia
fengniesysu@gmail.com, yunbocao@tencent.com, {jinpwa, cyl}@microsoft.com, panr@sysu.edu.cn

Abstract

For the task of entity disambiguation, mention contexts and entity descriptions both contain various kinds of information content while only a subset of them are helpful for disambiguation. In this paper, we propose a type-aware co-attention model for entity disambiguation, which tries to identify the most discriminative words from mention contexts and most relevant sentences from corresponding entity descriptions simultaneously. To bridge the semantic gap between mention contexts and entity descriptions, we further incorporate entity type information to enhance the co-attention mechanism. Our evaluation shows that the proposed model outperforms the state-of-the-arts on three public datasets. Further analysis also confirms that both the co-attention mechanism and the type-aware mechanism are effective.

Introduction

Entity disambiguation is the task of mapping textual mentions of entities in unstructured text to the corresponding entities in a knowledge base. For example, the entity mention 'Hendrix' shown in Figure 1 may refer to an American rock guitarist (e_a) or a town in Bryan County, Oklahoma (e_b), depending on the contexts.

Several aspects of evidence are useful for entity disambiguation, such as semantic similarity, global consistency (Han, Sun, and Zhao 2011) and entity popularity (Cheng and Roth 2013; Durrett and Klein 2014; Huang et al. 2014; Heinzerling, Strube, and Lin 2017). Among all these aspects, it has been widely accepted that *semantic similarity* between the context of an entity mention and its target entity candidates is the key to resolve the ambiguity (Bunescu and Pasca 2006; Cucerzan 2007; Ji and Grishman 2011; Shen, Wang, and Han 2015). In this paper, we propose a **Type**-Aware **Co**-**Att**ention model (`TypeCoAtt`) for entity disambiguation, which introduces a **co-attention** mechanism to estimate the semantic similarity. This model is motivated as follows.

Identifying discriminative words from mention contexts and relevant sentences from corresponding entity descriptions can be effective for disambiguation. Take a specified query Q_1 in Figure 1 as an example, in order to determine the meaning of 'Hendrix' ('Jimi Hendirx' vs. 'Hendrix, Oklahoma'), it is sufficient to examine the context words *song* and *guitar*. Meanwhile, in the description of entity e_a (with the name 'Jimi Hendrix') sentences **S1** and **S2** contain more information relevant to the context of Q_1 than **S3**. To take advantage of this observation, our proposal introduces a **co-attention** mechanism which chooses information crucial for disambiguation in both mention contexts and entity descriptions simultaneously. Specifically, it uses an entity representation to guide context attention and uses a context representation to guide description attention. Previous studies (Lazic et al. 2015) on entity disambiguation also try to pick up discriminative content from mention contexts or entity descriptions separately. However, for entity disambiguation, the mention contexts and entity descriptions both contain various kinds of information content while only a subset of them are helpful for disambiguation. This insight motivates the idea of the co-attention mechanism that aligns content of mention contexts and entity descriptions, and picks up discriminative content from mention contexts and entity descriptions simultaneously.

Moreover, entity type information for candidate entity can help to bridge the semantic gap between mention contexts and entity descriptions. For example, *vocal* and *style* are most helpful words for mention 'Hendrix' in Q_2, however, these words are absent from its answer e_a but appear in the descriptions of e_1, ..., e_n sharing the same entity type 'artist.musician' as e_a. In light of that, we borrow the information from the type of an entity to enhance the co-attention mechanism.

We conduct extensive experiments with three public datasets. The results show that the proposed type-aware co-attention model outperforms previous neural methods. When combined with other evidence (e.g., similarity between entity mention and title of entity description, popularity), the proposal can outperform the state-of-the-arts as well. The detailed analysis shows that our proposal performs well particularly over hard queries and the co-attention mechanism is crucial for the success of our proposal. The main contributions of this paper are summarized as:

- To the best of our knowledge, this is the first effort of studying both context attention and (entity) description at-

[*]Correspondence author is Rong Pan. This work was done when the first author was an intern and second author was an employee at Microsoft Research Asia.

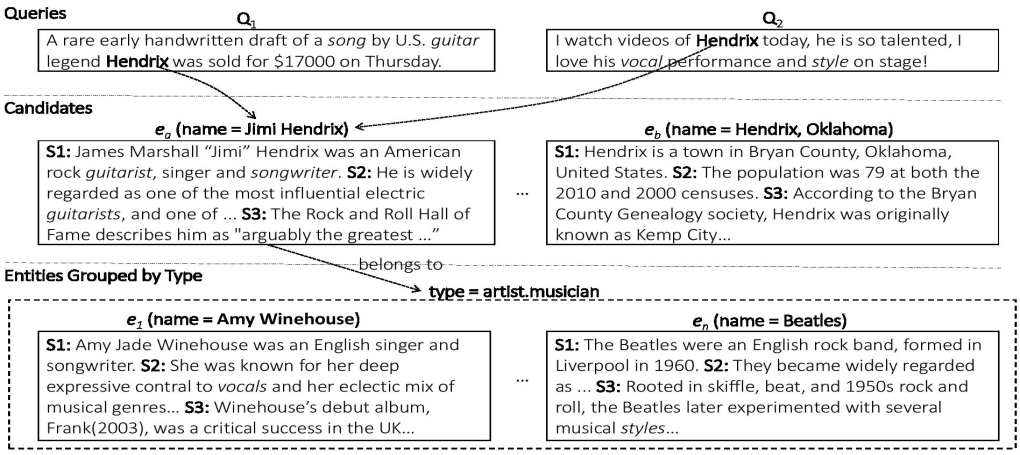

Figure 1: Two example queries Q_1 and Q_2 with 'Hendrix' as a mention and e_a and e_b as candidate entities. Furthermore, e_a shares the same entity type 'artist.musician' with e_1, ..., e_n.

tention for entity disambiguation in a single framework.

- To further enhance the co-attention mechanism, We propose a type-aware mechanism which incorporate entity type information.
- Experimental studies on three publicly available datasets show that our proposed framework outperforms the current state-of-the-art results, suggesting the effectiveness of our approach that jointly considers mention contexts, candidate descriptions and entity type information.

Task and Notations

Assume that we are given a query q and a set of candidate entities $\{e_n\}_{n=1}^N$. A query q consists of a pair (m, ctx), where m denotes an entity mention and ctx denotes context of the mention, i.e., a piece of text surrounding m. ctx is a sequence of words $[w_1, w_2, \cdots, w_l]$. Each candidate entity e consists of a pair ($desc$, t), where $desc$ denotes the description of e in a knowledge base (e.g., the article defining e) and t denotes the entity type of e. A description $desc$ consists of a sequence of sentences, i.e., $desc = [s_1, s_2, \cdots, s_m]$. The task of entity disambiguation requires us to choose an entity e^* from $\{e_n\}_{n=1}^N$ (meaning that e^* is referred to by m) if the set includes the answer, or nothing otherwise.

Note that the task of entity disambiguation usually involves a sub-task, namely constructing the set of candidates $\{e_n\}_{n=1}^N$ for a given query q. In this paper, we assume that this sub-task is well solved by previous work (Fang et al. 2016; Francis-Landau, Durrett, and Klein 2016)

Our Proposed Model

Model Overview

Figure 2 presents an overview of our proposed model for entity disambiguation. We name it as '**T**ype-**A**ware **C**o-**A**ttention model' (`TypeCoAtt`). Generally, `TypeCoAtt` is a neural network for calculating the semantic similarity

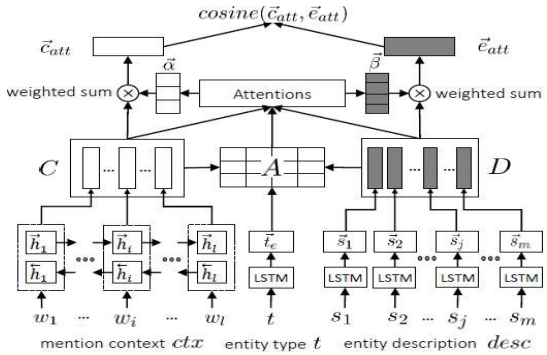

Figure 2: The **T**ype-**A**ware **C**o-**A**ttention Model.

between the context of mention entity and each candidate $e = (desc, t)$. Therefore its input consists of three parts, namely the mention context ctx from the specified query, the entity description $desc$ and the entity type t.

`TypeCoAtt` first encodes mention context ctx and entity description $desc$ independently using a bidirectional LSTM and stack the encoding vectors into matrices C and D, as the representations of ctx and $desc$ respectively. `TypeCoAtt` then forces the model to focus on discriminative information in both mention contexts and entity descriptions by a *co-attention* mechanism, using attention weights determined by an affinity matrix A. To bridge the semantic gap between mention contexts and entity descriptions, we further introduce entity type information to help the co-attention mechanism. `TypeCoAtt` finally provides a semantic similarity score between ctx and e with the adjusted representations ($\overrightarrow{c}_{att}$ and $\overrightarrow{e}_{att}$) modulated by the attention.

We first introduce the encoding methods for context and entity information, and then describe our designed co-attention mechanism and type-aware mechanism. Finally,

we introduce a linear combination model to integrate other contextual features and sparse features that are useful for entity disambiguation.

Context Encoding

The context ctx of a mention m is a sequence of words $w_1, ..., w_l$. To encode ctx, we first embed each word in ctx into a d dimensional vector by looking up a word embedding matrix $E \in \mathbb{R}^{d \times |V|}$ (V denotes the vocabulary and $|V|$ its size), which yields a sequence of word vectors $v_1, ..., v_l$.

Then we feed the set of word vectors into a bi-directional LSTM (bi-LSTM). In the architecture, one LSTM (Hochreiter and Schmidhuber 1997) processes the input from left to right while the other processes it from right to left.

$$\overrightarrow{h}_i = \text{LSTM}(\overrightarrow{h}_{i-1}, v_i), i = 1, \cdots, l \quad (1)$$

$$\overleftarrow{h}_i = \text{LSTM}(\overrightarrow{h}_{i+1}, v_i), i = l, \cdots, 1 \quad (2)$$

The context embedding $C = [\overrightarrow{c}_1, \overrightarrow{c}_2, \cdots, \overrightarrow{c}_l]$, with $C \in \mathbb{R}^{l \times k}$, is obtained by concatenating the forward and backward hidden states: $\overrightarrow{c_i} = [\overrightarrow{h}_i; \overleftarrow{h}_i]$. In this way, contextual vector $\overrightarrow{c}_i$ encodes information about the i-th word with respect to all other surrounding words in context ctx.

Entity Encoding

In this subsection, we encode entity description $desc$ and entity type t to represent entity e.

Entity descriptions can be encoded in several levels of granularity, i.e., word, sentence and paragraph. In this paper, we choose sentence as basic semantic unit since it covers an aspect of the described entity. Accordingly, a description is then a sequence of sentences $s_1, s_2, \cdots, s_m$.

Each sentence s_j is a sequence of words, i.e., $s_j = \{w_1^j, ..., w_{z_j}^j\}$, where z_j is the length of s_j. Like what we do for context encoding (c.f., Eq. 1 and 2), we first encode each sentence s_j in $desc$ with bi-LSTM. Then we concatenate the last forward hidden state $\overrightarrow{h}^j_{z_j}$ and the first backward hidden state $\overleftarrow{h}^j_1$ into a single vector: $\overrightarrow{s}_j = [\overrightarrow{h}^j_{z_j}; \overleftarrow{h}^j_1]$, where $\overrightarrow{s}_j \in \mathbb{R}^k$. And the entity embedding matrix is $D = [\overrightarrow{s}_1, \overrightarrow{s}_2, \cdots, \overrightarrow{s}_m], D \in \mathbb{R}^{m \times k}$.

The representation for entity type t is also a k-dimensional vector, denoted as $\overrightarrow{t}_e$. It can be treated as a segment of text the name of t and then applying bi-LSTM to it. For example, we encode the entity type 'location/city' by treating it as a sentence with two words 'location' and 'city'

Context-Entity Co-Attention

For the task of entity disambiguation, the mention contexts and entity descriptions both contain various kinds of information content while only a subset of them are helpful for disambiguation. This insight motivates the idea of the co-attention mechanism that aligns content of mention contexts and entity descriptions. In this section, we propose a co-attention mechanism that attends to words in ctx and sentences in $desc$ simultaneously. Our model is similar to (Lu et al. 2016) in research of visual question answering, however, this is the first work which considers co-attention for entity disambiguation.

We connect mention context ctx and entity e by calculating the similarity between every contextual vector in C and every sentence vector in D. An affinity matrix $A \in \mathbb{R}^{l \times m}$ is calculated as

$$A = CW_aD^T \quad (3)$$

$W_a \in \mathbb{R}^{k \times k}$ is used in a bilinear term, which allows us to compute a similarity between C and G more flexibly than just a dot product. And T denotes matrix transpose.

We then normalize the affinity matrix A row-wisely to produce attention relatedness L^C across all sentences in $desc$ for every word in ctx. Similarly, we normalize the affinity matrix A column-wisely to produce attention relatedness L^D across all words in ctx for every sentence in $desc$.

$$L^C = \text{softmax}(A),\ L^D = \text{softmax}(A^T) \quad (4)$$

Similar to (Lu et al. 2016), we compute the attention probabilities for ctx and $desc$ using the attention relatedness matrix L^C and L^D respectively.

$$H^c = \tanh(W_cC^T + (W_dD^T)L^C) \quad (5)$$

$$H^d = \tanh(W_dD^T + (W_cC^T)L^D) \quad (6)$$

$$\overrightarrow{\alpha} = \text{softmax}(w_{hc}^T H^c) \quad (7)$$

$$\overrightarrow{\beta} = \text{softmax}(w_{hd}^T H^d) \quad (8)$$

where $W_c, W_d \in \mathbb{R}^{k \times k}$, $w_{hc}, w_{hd} \in \mathbb{R}^k$ are weight parameters. $\overrightarrow{\alpha} \in \mathbb{R}^l$ and $\overrightarrow{\beta} \in \mathbb{R}^m$ are attention probabilities of each contextual vector $\overrightarrow{c}_i$ in ctx and sentence vector $\overrightarrow{s}_j$ in e respectively.

We get the updated representations $\overrightarrow{c}_{att}$ and $\overrightarrow{e}_{att}$ for ctx and e respectively by the weighted sum of the contextual vectors and sentence vectors

$$\overrightarrow{c}_{att} = \sum_{i=1}^{l} \alpha_i * \overrightarrow{c_i},\ \overrightarrow{e}_{att} = \sum_{j=1}^{m} \beta_j * \overrightarrow{s_j}, \quad (9)$$

and cosine similarity are employed to measure the similarity between $\overrightarrow{c}_{att}$ and $\overrightarrow{e}_{att}$

$$\text{sim}(\overrightarrow{c}_{att}, \overrightarrow{e}_{att}) = cosine(\overrightarrow{c}_{att}, \overrightarrow{e}_{att}) \quad (10)$$

Type-Aware Co-Attention

Entity type information can enrich the representation of candidate entity by providing a topical information to bridge the semantic gap between mention contexts and entity descriptions, which further helps identify discriminative content in the mention context. Therefore, we further propose to enhance the co-attention mechanism by adding type information to the affinity matrix A in Eq. 3 as follows:

$$A_{ij} = \overrightarrow{c_i}^T W_a \overrightarrow{s}_j + \overrightarrow{c_i}^T W_t \overrightarrow{t}_e \quad (11)$$

where $W_a, W_t \in \mathbb{R}^{k \times k}$ are bilinear terms. In this way, the weights of context attention are not only effected by every sentence in entity description but also entity type.

Integrating with Other Evidences

Apart from the information provided with mention context ctx and entity description $desc$ (i.e., Eq. 10), there are other levels of contextual information (i.e., mention m itself, mention document doc and entity title ttl) that are also helpful for entity disambiguation. Following (Francis-Landau, Durrett, and Klein 2016), we integrate other levels of information as:

$$f_{nn}(q,e) = \overrightarrow{w}_{evi}\Big[\text{sim}(\overrightarrow{c}_{att}, \overrightarrow{e}_{att}), F_{cnn}(q,e)\Big] \quad (12)$$

where $\overrightarrow{w}_{evi}$ is the weight vector for these evidences; $F_{cnn}(q,e)$ is the incorporated evidence vector which contains five elements $\text{sim}(\overrightarrow{ctx}, \overrightarrow{ttl})$, $\text{sim}(\overrightarrow{doc}, \overrightarrow{desc})$, $\text{sim}(\overrightarrow{doc}, \overrightarrow{ttl})$, $\text{sim}(\overrightarrow{m}, \overrightarrow{desc})$ and $\text{sim}(\overrightarrow{m}, \overrightarrow{ttl})$, where its every element is modeled by a CNN with sum pooling (Francis-Landau, Durrett, and Klein 2016).

Similar to (Francis-Landau, Durrett, and Klein 2016), we also incorporate human-defined features (referred to as sparse features) into our model

$$f_{full}(q,e) = \overrightarrow{w}_{evi}\Big[\text{sim}(\overrightarrow{c}_{att}, \overrightarrow{e}_{att}), F_{sparse}(q,e), F_{cnn}(q,e)\Big] \quad (13)$$

where $F_{sparse}(q,e)$ contains three sparse features: mention prior, co-reference, and the graph based collective feature from (Han, Sun, and Zhao 2011).

Although we combine other evidences into our model, the semantic similarity between the mention context and entity description is the key for entity disambiguation. For model training, we jointly optimize the parameters of CNNs, LSTMs, and weight vectors by maximizing the log-likelihood of the labeled corpus during model training.

Experiments

In this section, we describe our experimental results on entity disambiguation. Particularly, we investigate the use of the attention mechanism and the use of the type-aware mechanism.

Evaluation Data & Metric

We evaluate `TypeCoAtt` with the following three datasets.

ACE[1] (Bentivogli et al. 2010)**.** We use the version of ACE 2005 that contains Wikipedia link annotations. ACE 2005 consists of 597 articles. The test queries consists of 3,920 inKB queries (having target entities in KB) and 348 NIL queries (not having target entities in KB).

CoNLL (Hoffart et al. 2011)**.** This dataset is from the CoNLL 2003 shared task of named entity recognition for English. The documents are partitioned into train, test-a and test-b. We report performance on the 231 test-b documents with 4,485 (inKB) test queries.

KBP 2010. The KBP 2010 dataset comes from the KBP's annual tracks. It includes 1,020 (inKB) test queries. The standard training data of KBP 2010 contains only 1,500 queries. In our experiment, we additionally collect 55,388 queries by making use of anchor texts from Wikipedia (Sun et al. 2015). The newly-collected queries cover the same set of entity occurring in the original training dataset. We randomly split the new dataset into 10 folds, and then use 9 of them for model training and the remaining one for hyper-parameter tuning.

We employe inKB *F1* as the evaluation metric, which measures whether a top-ranked entity candidate is the ground truth for those non-NIL mentions. Note that since the test sets CoNLL and KBP 2010 do not include any NIL queries inKB *F1* is equivalent to inKB *accuracy*.

Experimental Setup

Knowledge Base and Entity Types The knowledge base (KB) that we utilize is derived from the English Wikipedia Dump on the December 2014 provided by (Francis-Landau, Durrett, and Klein 2016).

We collect type information for entities from Freebase tags (with the format, domain/type_class). Following (Ling and Weld 2012), we filter out irrelevant types to reduce the data noise, and only keep the well-maintained types (e.g. /location/city). For entities that have multiple types, we keep only the most frequent type[2] for each entity. Finally, 74 types are remained for use as our tag set. We will publish the 'type' data for entities in the evaluation dataset.

Model Training For all the datasets, we use the words surrounding the mention with the window size 15 as mention contexts. As most important information on entities is usually included at the beginning of Wikipedia articles, we utilize only the first 200 words in the articles as entity description, and we use the default English sentence tokenizer in NLTK[3] to split sentences, and limit the maximum sentence length to 30 in Wikipedia articles.

We pre-train word embeddings with the whole English Wikipedia Dump using the word2vec toolkit (Mikolov et al. 2013). The training parameters are set to the default values in this toolkit. The dimensionality of the word embeddings is set to 300. We do not update the word embeddings during training `TypeCoAtt`.

For the model training, we first pre-train the model parameters on 40,000 queries that are randomly selected from the anchor text of Wikipedia, then fine-tune our model parameters using training datasets [4]. We use the stochastic gradient descent algorithm and the AdaDelta optimizer (Zeiler 2012). The gradients are computed via back-propagation. The dimensionality of the hidden units in LSTM is set to 300. The parameters in LSTM are initialized using a normal distribution with a mean of 0 and a variance of $\sqrt{6/(d_{in}+d_{out})}$, where d_{in} is the dimensionality of the input layer and d_{out} is the dimensionality of the output layer (Glorot and Ben-

[1]To have a fair comparison, we use the version of the datasets provided by (Francis-Landau, Durrett, and Klein 2016). It is slightly different from the standard ones.

[2]The frequencies are calculated in all collected tags.

[3]Natural Language Toolkit. http://www.nltk.org/

[4]Different datasets exists domain difference. We train models on each datasets following (Francis-Landau, Durrett, and Klein 2016)

	Method	ACE	CoNLL	KBP2010
Neural Approach	(Sun et al. 2015)	-	-	83.9
	FL-Neural	84.5	81.2	-
	(Nitish Gupta and Roth 2017)	85.6	82.9	-
	CoAtt	85.8	82.5	84.0
	TypeCoAtt	86.3	82.9	84.6
	CoAtt (pre-train)	86.4	82.9	85.7
	TypeCoAtt (pre-train)	**86.8**	**83.4**	**86.4**
Collective Approach	(Globerson et al. 2016)	-	89.5	87.2
	FL-Full	89.9	85.5	-
	sparse	83.6	81.9	78.6
	TypeCoAtt+sparse	90.7	89.6	88.2
	TypeCoAtt+sparse (pre-train)	**91.1**	**89.8**	**89.1**

Table 1: Comparison with the state of the arts over the datasets ACE, CoNLL, and KBP 2010.

gio 2010) . And all the other parameters for co-attention are initialized with a uniform distribution $U(-0.01, 0.01)$.

Main Results

The overall performance of various approaches is shown in Table 1. We divide the baselines into two categories, the pure neural network approaches, and the collective approaches which combines multiple sparse features.

Our neural-network-based method `CoAtt` (defined by Eq. 12 and Eq. 3) that uses only the co-attention mechanism without type information, outperforms the other three state-of-the-art methods within neural approaches. Comparing `CoAtt` with `FL-Neural` (Francis-Landau, Durrett, and Klein 2016) which substitutes co-attention similarity of mention contexts and entity descriptions in Eq. 12 with a CNN based cosine similarity, the improvement over `FL-Neural` shows `CoAtt`'s effectiveness of modeling semantics between mention context and entity description. In addition, after introducing entity type information to co-attention mechanism, `TypeCoAtt` achieves more improvements. Comparing `CoAtt` with the method `(Sun et al. 2015)` which is trained over one million Wikipedia anchor texts, our method `CoAtt` achieves comparable performance even trained with less training data. We therefore investigate the effectiveness of introducing more training data from Wikipedia anchor texts, and our method `CoAtt (pre-train)` get improvements over `CoAtt`. The results show the effectiveness of our model, and the importance of the scale of training data for neural approaches. `(Nitish Gupta and Roth 2017)` uses similar contextual information (e.g. mention contexts, entity descriptions and fine-grained entity types) for this task. Moreover, it incorporates the mention prior feature for entity disambiguation and is trained over the full Wikipedia anchor texts. Compared with this work, the improvement of `TypeCoAtt (pre-train)` shows the effectiveness of co-attention mechanism. `(Nitish Gupta and Roth 2017)` focus on the encoding of contextual information, while ours is focus on the co-attention mechanism. In the future, we can try to combine these two work.

For collective approaches, we first present the performance of using only sparse features, which is worse than `CoAtt`. Similar to `FL-Full` (Francis-Landau, Durrett, and Klein 2016), we combine our neural-network model `CoAtt` with sparse features introduced in Eq. 13. When combining with sparse features, the improvements shows that neural-network-based methods and sparse features capture orthogonal sources of information. Therefore our proposed method `CoAtt+sparse` outperforms not only the state-of-the-art neural methods (i.e., `FL-Full`), but also the state-of-the-art methods solely relying on sparse features[5](i.e., (Globerson et al. 2016)). In addition, our method `TypeCoAtt+sparse(pre-train)` achieves improvements by introducing more training data from Wikipedia.

Discussions

In this section, we discuss several key observations based on the experimental results. In the experiments, to avoid the influences from other evidences such as more information pairs and sparse features, we report our results of *only* modeling semantic similarities between mention contexts and entity descriptions. And we provide the performance of the pre-trained results for all the methods in this section for the stableness of the model and fair comparison.

Effect of Co-Attention We compare our proposed method with five baseline methods to investigate the necessity of co-attention mechanism in the modeling semantics of mention contexts and entity descriptions for entity disambiguation.

`FL-Single` is a convolutional neural network method (Francis-Landau, Durrett, and Klein 2016) with a setting of only taking the information pair of mention contexts and entity descriptions.

`CoAtt-Single` is our proposed model with co-attention mechanism defined by Eq. 10 and Eq. 3.

`LSTM-AVG` is a method only taking our components for context encoding and description encoding. It is the simple version of `CoAtt-Single` which the attentions for both mention contexts and entity descriptions are replaced by a uniform distribution.

`Context-ATT` is a method considering only attention to context. It is the simple version of `CoAtt-Single` which

[5]We compare the baselines in CoNLL without using the additional knowledge base YAGO for fair comparison.

Method	ACE	CoNLL	KBP 2010
FL-Single	75.7	79.7	79.4
LSTM-AVG	82.9	81.6	80.1
Context-ATT	83.5	80.8	80.5
Description-ATT	82.7	82.0	79.8
CoAtt-Single	84.0	82.3	82.4
TypeCoAtt-Single	**84.9**	**83.1**	**84.6**

Table 2: Results for mention and entity description disambiguation over the datasets ACE, CoNLL, KBP 2010.

Dataset		FL-CNN (Single)	CoAtt (Single)
ACE	Total	75.8	**84.0**
	Hard	70.8	**79.5**
CoNLL	Total	79.8	**82.3**
	Hard	64.5	**66.7**
KBP2010	Total	79.4	**82.4**
	Hard	68.8	**73.9**

Table 3: Comparison over hard queries.

the attentions for entity descriptions are replaced by a uniform distribution.

`Description-ATT` is a method considering only attention to entity description. It is the simple version of `CoAtt-Single` which the attentions for mention contexts are replaced by a uniform distribution.

The evaluation results on our proposal and the baselines are listed in Table 2. From the table, we can see that even without the attention mechanism and the type information, the simple method `LSTM-AVG` outperforms `FL-Single`, which illustrates that our encoding method based on bi-LSTM is more effective in modeling mention contexts and entity descriptions.

`Context-ATT` and `Description-ATT` which employes single side attention mechanism have similar performance to `LSTM-AVG`. These single side attention models pick up discriminative information from only one side while remaining redundant and noisy information in the other side, which is not enough to capture the semantic between mention context and entity description.

`CoAtt-Single` with co-attention mechanism gets better performance on all the datasets. By picking up discriminative information from mention context and entity description simultaneously, the co-attention mechanism aligns content from both sides and helps to capture the semantic similarity.

Our full model `TypeCoAtt-Single` that further incorporates entity type information into co-attention mechanism yields the best result. The result demonstrates that content sparseness of entity descriptions does exist and the proposed type-aware co-attention mechanism can effectively make use of type information to further boost performance.

Performance over Hard Cases In entity disambiguation, a large portion of entity mentions can be well solved by matching them with the most popular entity in knowledge base, as these are the cases that mentions can be linked correctly without checking any contexts. We exclude this set of cases from the overall test set and name the remaining part as **Hard** set, which our method aims to tackle. We obtain the **Hard** sets for ACE, CoNLL and KBP 2010 with the sizes of 1,281, 2,042 and 429.

Method	ACE	CoNLL	KBP 2010
CoAtt-Single	84.0	82.3	82.4
TYPE-ONLY	56.5	46.6	45.5
ATT+TYPE	84.3	81.9	82.8
TypeCoAtt-Single	**84.9**	**83.1**	**84.6**

Table 4: Different ways to utilize type information.

In order to have a fair comparison, we exclude the improvements brought by entity type information by comparing the model `CoAtt-Single` with current state-of-the-art neural model `FL-Single` (Francis-Landau, Durrett, and Klein 2016). From Table 3, we can see that `CoAtt-Single` can achieve more improvements over the **Hard** sets than over the total sets when the datasets ACE and KBP 2010 are used. And the improvements over the **Hard** set and the total set are comparable when CoNLL is used. To certain extent, this confirms that our proposal can really work as designed over the queries that it is targeted at (namely hard queries).

Effect of Type-Aware Co-Attention In this subsection, we empirically explain the reason that we introduce type information to help better co-attention between mention contexts and entity descriptions.

We first exam the effect of entity type information by using only the entity type information to represent entity (the entity descriptions have been removed), denoted as `TYPE-ONLY`, and then perform context side attention similar to `Context-ATT-Single`. The results in Table 4 shows that using only type information can only solve part of the disambiguation.

A straightforward way of introducing type information for co-attention mechanism is to treat entity type t as a yet-another sentence in entity description $desc$. That is to replace the entity embedding matrix $D = [\vec{s}_1, \vec{s}_2, \cdots, \vec{s}_m]$ with $D^{'} = [\vec{t}_e, \vec{s}_1, \vec{s}_2, \cdots, \vec{s}_m]$ and keep rest of the model unchanged, denoted as `ATT+TYPE`.

We can see that by incorporating type information, `ATT+TYPE` and `TypeCoAtt-Single` both outperform `CoAtt-Single` on ACE and KBP 2010, except `ATT+TYPE` on CoNLL. The CoNLL dataset is special. For example, given mention "England" with context "Cricket England vs. Parkistan Final test scoreboard", two candidate entities "England Cricket Team" and "England" with different entity types are highly related to the context (the answer is the "England"). Adding entity type information to `ATT+TYPE` directly would mislead the model, however, `TypeCoAtt-Single` dynamically controls the weight of adopting entity type or entity sentences by co-attention, thus outperforms the `CoAtt-Single` on three datasets. The result confirms that the type-aware co-attention mechanism

Context Attention	
Mention: Chris Johnson	**Mention**: Colorado City
so let him do that for some reason though I envision him performing phenomenally elsewhere Chris Johnson Johnson had just 13 takles last season and the Raiders currently have 11 defensive backs	helps boys who have been pushed out of the FLDS communicate in Hidale Utah and Colorado City Ariz Price spent several days in Eldorado advising Texas officials he said she told them that
Candidate Entity: Chris Johnson (cornerback)	**Candidate Entity**: Colorado City, Arizona
Top sentences: • At Louisville, Johnson total tackles solo one tackle for a loss and seven defended. • At Pine tree high school, Johnson started in the footballteam for years during his senior season. He interceptedand helped the team reach a 73 record.	**Top sentences**: • Colorado city is a town in mohave county Arizona, United States andis located in region known as theArizona strip. • The twolayer legal battle followed became a publicrelation disaster damaged pyles political career and set a handsoff tone toward the town in Arizona for the next years.

Table 5: Two co-attention examples from the **Hard** set in KBP 2010 disambiguated correctly. Context attentions (darker colors depict higher probabilities) and top two sentences with largest attention scores in the entity description are given.

is more effective to leverage entity type information than adding type information into entity descriptions directly.

A Case of Attention To illustrate how our attention model actually works, we provide two examples from the **Hard** set of KBP 2010 that are solved by our co-attention mechanism in Table 5. For the example in the first column, except the mention words, co-attention mechanism picks up two discriminative words "tackles" and "Raiders" from context side, and picks up the sentences most related to "tackles" and "Raiders" correspondingly. And for the example in the second column, the co-attention mechanism again selects the most informative words "Colorado", "Arizona" and "Utah", and selects the sentences related to these three words.

Related Work

Entity disambiguation methods can roughly fall into local approaches and global (collective) approaches. Local approaches (Zheng et al. 2010; Ji and Grishman 2008; Bunescu and Pasca 2006) focus on internal structures between each mention and candidate entity separately. (Milne and Witten 2008) use the mention popularity and 'unambiguous links' to compute entity relatedness. Global approaches take all the entities in a document into consideration (Cucerzan 2007; Hoffart et al. 2011). (Cucerzan 2007) use the Wikipedia category structure and (Hoffart et al. 2011) use Wikipedia link based measures.

Neural networks methods are recently applied to entity disambiguation. (He et al. 2013) investigate Stacked Denoising Auto-encoders to learn entity representation. (Sun et al. 2015) apply convolutional neural networks and neural tensor networks to model mentions and entities. While (Francis-Landau, Durrett, and Klein 2016) use the convolutional neural networks to learn multiple granularities of contextual information, and combine it with sparse features. (Nguyen et al. 2016) use the recurrent neural networks to model the global connections of mentions within a single document. (Nitish Gupta and Roth 2017) encodes the mention contexts, entity descriptions, and entity types via a neural network jointly. Instead of modeling entity descriptions at document level, we choose sentence as basic semantic unit since it covers an aspect of the described entity. In addition, existing works do not model attentions between the mention context and entity descriptions explicitly.

Attention mechanism is recently applied to entity disambiguation (Lazic et al. 2015; Globerson et al. 2016; Ganea and Hofmann 2017; Eshel, Cohen, and Radinsky 2017). (Lazic et al. 2015) use EM algorithm to pick up most discriminative contextual words for disambiguation. The function provided by the EM algorithm is similar to the ablated version of proposal with only context attention. (Ganea and Hofmann 2017) and (Eshel, Cohen, and Radinsky 2017) apply the local attention of contexts side with neural network based methods. However, these works ignore the other aspect of attention mechanism, i.e., description attention. (Globerson et al. 2016) apply an attention mechanism to select coherent entities using sparse features. Comparing with our proposal, their attention is only over contextual entities, but not all contextual words.

The Co-attention has been applied to the areas such as visual question answering (Lu et al. 2016) and reading comprehension (dos Santos et al. 2016; Xiong, Zhong, and Socher 2016). However, none of previous work considers such an attention for entity disambiguation.

Conclusion

We propose a type-aware co-attention model for entity disambiguation. The co-attention mechanism associates different importance to words in contexts and to sentences in entity description. And introducing entity type information bridges the semantic gap between mention contexts and entity descriptions, which further enhances the co-attention mechanism. We have shown the effectiveness of our proposed model over three public datasets through extensive experiments. In the future, we will try our model on predicting NIL entities and allow entities to have multiple entity types in type-aware co-attention mechanism.

Acknowledgements

We thank Jing Liu, Jin-Ge Yao, Zhirui Zhang and Shuangzhi Wu for helpful discussion. The fifth author was supported

by the National Key R&D Program of China under Grant 2016YB 0201900, and the Fundamental Research Funds for the Central Universities under Grant 17LGJC23.

References

Bentivogli, L.; Forner, P.; Giuliano, C.; Marchetti, A.; Pianta, E.; and Tymoshenko, K. 2010. Extending english ace 2005 corpus annotation with ground-truth links to wikipedia. In *In Proceedings of the 2nd Workshop on The People's Web Meets NLP: Collaboratively Constructed Semantic Resources*.

Bunescu, R., and Pasca, M. 2006. Using encyclopedic knowledge for named entity disambiguation. In *EACL*.

Cheng, X., and Roth, D. 2013. Relational inference for wikification. In *EMNLP 2013*.

Cucerzan, S. 2007. Large-scale named entity disambiguation based on wikipedia data. In *EMNLP-CoNLL*, volume 2007.

dos Santos, C. N.; Tan, M.; Xiang, B.; and Zhou, B. 2016. Attentive pooling networks. *CoRR* abs/1602.03609.

Durrett, G., and Klein, D. 2014. A joint model for entity analysis: Coreference, typing, and linking. *TACL* 2:477–490.

Eshel, Y.; Cohen, N.; and Radinsky, K. 2017. Named entity disambiguation for noisy text. In *CoNLL*, volume 2017.

Fang, W.; Zhang, J.; Wang, D.; Chen, Z.; and Li, M. 2016. Entity disambiguation by knowledge and text jointly embedding. In *CoNLL*.

Francis-Landau, M.; Durrett, G.; and Klein, D. 2016. Capturing semantic similarity for entity linking with convolutional neural networks. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, 1256–1261.

Ganea, O., and Hofmann, T. 2017. Deep joint entity disambiguation with local neural attention. In *EMNLP*.

Globerson, A.; Lazic, N.; Chakrabarti, S.; Subramanya, A.; Ringgaard, M.; and Pereira, F. 2016. Collective entity resolution with multi-focal attention. In *ACL*.

Glorot, X., and Bengio, Y. 2010. Understanding the difficulty of training deep feedforward neural networks. In *AISTATS*.

Han, X.; Sun, L.; and Zhao, J. 2011. Collective entity linking in web text: a graph-based method. In *SIGIR*, 765–774.

He, Z.; Liu, S.; Li, M.; Zhou, M.; Zhang, L.; and Wang, H. 2013. Learning entity representation for entity disambiguation. In *ACL*.

Heinzerling, B.; Strube, M.; and Lin, C.-Y. 2017. Trust, but verify better entity linking through automatic verification. EACL.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation* 9(8):1735–1780.

Hoffart, J.; Yosef, M. A.; Bordino, I.; Fürstenau, H.; Pinkal, M.; Spaniol, M.; Taneva, B.; Thater, S.; and Weikum, G. 2011. Robust disambiguation of named entities in text. In *EMNLP*.

Huang, H.; Cao, Y.; Huang, X.; Ji, H.; and Lin, C. 2014. Collective tweet wikification based on semi-supervised graph regularization. In *ACL*.

Ji, H., and Grishman, R. 2008. Refining event extraction through cross-document inference. In *ACL*.

Ji, H., and Grishman, R. 2011. Knowledge base population: Successful approaches and challenges. In *ACL-HLT*.

Lazic, N.; Subramanya, A.; Ringgaard, M.; and Pereira, F. 2015. Plato: A selective context model for entity resolution. *TACL* 3:503–515.

Ling, X., and Weld, D. S. 2012. Fine-grained entity recognition. In *AAAI*.

Lu, J.; Yang, J.; Batra, D.; and Parikh, D. 2016. Hierarchical question-image co-attention for visual question answering. In *NIPS*.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In Burges, C. J. C.; Bottou, L.; Welling, M.; Ghahramani, Z.; and Weinberger, K. Q., eds., *NIPS*. 3111–3119.

Milne, D. N., and Witten, I. H. 2008. Learning to link with wikipedia. In *CIKM*.

Nguyen, T. H.; Fauceglia, N.; Muro, M. R.; Hassanzadeh, O.; Gliozzo, A. M.; and Sadoghi, M. 2016. Joint learning of local and global features for entity linking via neural networks. In *Coling*.

Nitish Gupta, S. S., and Roth, D. 2017. Entity linking via joint encoding of types, descriptions, and context. In *EMNLP*.

Shen, W.; Wang, J.; and Han, J. 2015. Entity linking with a knowledge base: Issues, techniques, and solutions. *IEEE Trans. Knowl. Data Eng.* 27(2):443–460.

Sun, Y.; Lin, L.; Tang, D.; Yang, N.; Ji, Z.; and Wang, X. 2015. Modeling mention, context and entity with neural networks for entity disambiguation. In *IJCAI*.

Xiong, C.; Zhong, V.; and Socher, R. 2016. Dynamic coattention networks for question answering. *CoRR* abs/1611.01604.

Zeiler, M. D. 2012. Adadelta: an adaptive learning rate method.

Zheng, Z.; Li, F.; Huang, M.; and Zhu, X. 2010. Learning to link entities with knowledge base. In *NAACL-HLT*.