Describing a Knowledge Base

**Qingyun Wang[1], Xiaoman Pan[1], Lifu Huang[1], Boliang Zhang[1], Zhiying Jiang[1],
Heng Ji[1], Kevin Knight[2]**

[1] Rensselaer Polytechnic Institute
jih@rpi.edu
[2] DiDi Labs and University of Southern California
knight@isi.edu

Abstract

We aim to automatically generate natural language descriptions about an input structured knowledge base (KB). We build our generation framework based on a pointer network which can copy facts from the input KB, and add two attention mechanisms: (i) *slot-aware attention* to capture the association between a slot type and its corresponding slot value; and (ii) a new *table position self-attention* to capture the inter-dependencies among related slots. For evaluation, besides standard metrics including BLEU, METEOR, and ROUGE, we propose a *KB reconstruction* based metric by extracting a KB from the generation output and comparing it with the input KB. We also create a new data set which includes 106,216 pairs of structured KBs and their corresponding natural language descriptions for two distinct entity types. Experiments show that our approach significantly outperforms state-of-the-art methods. The reconstructed KB achieves 68.8% - 72.6% F-score.[1]

1 Introduction

Show and tell, showing an audience something and telling them about it, is a common classroom activity for early elementary school kids. As a similar practice for knowledge propagation, we often need to describe and/or explain the information in a structured knowledge base (KB) in natural language, in order to make the knowledge elements and their connections easier to comprehend.

For example, (Cawsey et al., 1997) presents a natural language generation system to convert structured medical records to natural language text descriptions, which enables more effective communication between health care providers and their patients and among health care providers themselves.

Moreover, 51% of entity attributes in the current English Wikipedia Infoboxes are not described in English articles in the Wikipedia dump of April 1, 2018. The availability of vast amounts of Linked Open Data (LOD) and Wikipedia derived resources such as DBPedia, WikiData and YAGO encourages pursuing a new direction of knowledge-driven (Whitehead et al., 2018; Lu et al., 2018) or semantically oriented (Bouayad-Agha et al., 2013) Natural Language Generation (NLG). We aim to fill in this knowledge gap by developing a system that can take a KB (consisted of a set of slot types and their values) about an entity as input (see example in Table 1), and automatically generate a natural language description (Table 2).

Slot Type	Row	Slot Value		
Name	1	Silvi Jan		
Member of Sports team	2	ASA Tel Aviv University		
	3	Hapoel Tel Aviv F.C.(women)		
	4	Maccabi Holon F.C. (women)		
	5	Israel women's national football team	Matches	22
			Goals	29
Date of Birth	6	27 October 1973		
Country of Citizenship	7	Israel		
Position	8	Forward (association football)		

Table 1: Input: Structured Knowledge Base

Neural generation to generalize linguistic expressions. One major challenge lies in generalizing a wide variety of expressions, patterns, tem-

---

Reference	Silvi Jan (born 27 October 1973) is a retired female Israeli . Silvi Jan has been a Forward (association football) for the Israel women's national football team for many years appearing in 22 matches and scoring 29 goals. After Hapoel Tel Aviv F.C.(women) folded, Jan signed with Maccabi Holon F.C. (women) where she played until her retirement in 2007. In January 2009, Jan returned to league action and joined ASA Tel Aviv University . In 1999, with the establishment of the Israeli Women's League, Jan returned to Israel and signed with Hapoel Tel Aviv F.C.(women) .
Seq2seq	(born 23 April 1981) is a retired Israeli footballer. He played for the Thailand 's (scoring one goal) and was a member of the team that won the first ever player in the history of the National Basketball League. She played for the team from 1997 to 2001 scoring 29 goals. She played for the team from 1997 to 2001 scoring 29 goals. She played for the team from 1999 to 2001 and played for the team in the 1997 and 2003 seasons.
Pointer	Silvi Jan the fourth past the Maccabi Holon F.C. (women). On 27 October 1973 in 29 2014) (born 22) is a former Israel . She was a Forward (association football) and currently plays for Hapoel Tel Aviv F.C.(women) in the Swedish league. She played for the ASA Tel Aviv University in the Swedish league. She was a member of the Israel women's national football team at the beginning of the 2008 season.
+ Type	Silvi Jan (born 27 October 1973) is a former Israeli footballer. He played for Hapoel Tel Aviv F.C.(women) and ASA Tel Aviv University .
+ Type & Position	Silvi Jan (born 27 October 1973) is a former Israel . He played for Israel women's national football team , Hapoel Tel Aviv F.C.(women) , ASA Tel Aviv University and Maccabi Holon F.C. (women) . He was capped 22 times for the Israel women's national football team .

Table 2: Human and System Generated Descriptions about the KB in Table 1

plates and styles which human use to describe the same slot type. For example, to describe a football player's membership with a team, we can use various phrases including *member of*, *traded to*, *drafted by*, *played for*, *face of*, *loaned to* and *signed for*. Instead of manually crafting patterns for each slot type, we leverage the existing pairs of structured slots from Wikipedia infoboxes and Wikidata (Vrandečić and Krötzsch, 2014) and the corresponding sentences describing these slots in Wikipedia articles as our training data, to learn a deep neural network based generator.

Pointer network to copy over facts. The previous work (Liu et al., 2018) considers the slot type and slot value as two sequences and applies a sequence to sequence (seq2seq) framework (Cho et al., 2014) for generation. However, the task of describing structured knowledge is fundamentally different from creative writing, because we need to cover the knowledge elements contained in the input KB, and the goal of generation is mainly to clearly describe the semantic connections among these knowledge elements in an accurate and coherent way. The seq2seq model fails to capture such connections and tends to generate wrong information (e.g., *Thailand* in Table 2). To address this challenge, we choose a pointer network (See et al., 2017) to copy slot values directly from the input KB.

Slot type attention. However, the copying mechanism in the pointer network is not able to capture the alignment between a slot type and its slot value, and thus it often assigns facts to wrong slots. For example, *22* in Table 2 should be the number of matches instead of birth date. It also tends to repeat the same slot value based on language model, e.g., "*Uroplatus ebenaui is a of gecko endemic to* **Madagascar**. *The Uroplatus is a member of the species of the genus* **Madagascar**.". We propose a **Slot-aware Attention** mechanism to compute slot type attention and slot value attention simultaneously and capture their correlation. Attention mechanism in deep neural networks (Denil et al., 2012) is inspired from human visual attention, which refers to human's capability to focus on a certain region of an image with high resolution while perceiving the surrounding image in low resolution. It allows the neural network to have access to the hidden state of the encoder, and thus learn what to attend to. For example, for a *Date of Birth* slot type, words such as *born* may receive higher attention than *female*. As we can see in Table 2 (*+Type*), the output with slot type attention contains more precise slots.

Table position attention. Multiple slots are often interdependent. For example, a football player

may join multiple teams, with each team associated with a certain number of points, goals, scores and games participated. We design a new table position based self-attention to capture correlations among interdependent slots and put them in the same sentence. For example, our model successfully associates the number of matches *22* with the *Israel women's national football team* as shown in Table 2.

The major contributions of this paper are:

- For the first time, we propose a new table position attention which proves to be effective at capturing inter-dependencies among facts. This new approach achieves 2.5%-7.8% F-score gain at KB reconstruction.

- We propose a *KB reconstruction based metric* to evaluate how many facts are correctly expressed in the generation output.

- We create a large dataset of KBs paired with natural language descriptions for 106,216 entities, which can serve as a new benchmark.

2 Model

We formulate the input structured KB to the model as a list of triples: $L = [(s_1, v_1, (r_1, \hat{r_1})), ..., (s_n, v_n, (r_n, \hat{r_n}))]$, where s_i denotes a slot type (e.g., *Country of Citizenship*), v_i denotes the corresponding slot value (e.g., *Israel*), and $(r_i, \hat{r_i})$ denotes the position of the triple in the input list and consists of the forward position r_i and the backward position $\hat{r_i} = n - r_i + 1$. The outcome of the model is a paragraph $Y = [y_1, y_2, ..., y_m]$. The training instances for the generator are provided in the form of: $T = [(L_1, Y_1), ..., (L_k, Y_k)]$.

2.1 Sequence-to-Sequence with Slot-aware Attention

Following previous studies on describing structured knowledge (Lebret et al., 2016; Sha et al., 2018; Liu et al., 2018), we apply a sequence-to-sequence based approach, and incorporate a slot-aware attention to generate the descriptions.

Encoder Given a structured KB input: $L = [(s_1, v_1, (r_1, \hat{r_1})), ..., (s_n, v_n, (r_n, \hat{r_n}))]$, where s_i, v_i, r_i, $\hat{r_i}$[2] are randomly embedded as vectors $\mathbf{s}_i$, $\mathbf{v}_i$, $\mathbf{r}_i$, $\hat{\mathbf{r}}_i$[2] respectively, we concatenate the vector rep-

---

[2]We use bold mathematical symbols to denote vector representations for the whole paper.

resentations of these fields as $\mathbf{l}_i = [\mathbf{s}_i, \mathbf{v}_i, \mathbf{r}_1, \hat{\mathbf{r}}_1]$, and obtain $\mathbf{L} = [\mathbf{l}_1, \mathbf{l}_2, ..., \mathbf{l}_n]$.

We attempted to apply the average of $\mathbf{L}$ as the representation for the input KB. However, such flat representation vectors fail to capture the structured contextual information in the entire KB. Therefore, we apply a bi-directional Gated Recurrent Unit (GRU) encoder (Cho et al., 2014) on $\mathbf{L}$ to produce the encoder hidden states $\mathbf{H} = [\mathbf{h}_1, \mathbf{h}_2, ..., \mathbf{h}_n]$, where $\mathbf{h}_i$ is a hidden state for $\mathbf{l}_i$.

Decoder with Slot-aware Attention The decoder is a forward GRU network with an initial hidden state $\mathbf{h}_n$, which is the encoder hidden state of the last token. In order to capture the association between a slot type and its slot value, we design a **Slot-aware Attention**. At each step t, we compute the attention distribution over the sequence of input triples. For each triple i, we assign it an attention weight:

$$e_i^t = v^\top \tanh\left(W_h \tilde{\mathbf{h}}^t + W_s \mathbf{s}_i + W_v \mathbf{v}_i + W_c c_i^t + b_e\right)$$
$$\alpha^t = \text{Softmax}\left(e^t\right)$$

where $\tilde{\mathbf{h}}^t$ is the decoder hidden state at step t. $\mathbf{s}_i$ and $\mathbf{v}_i$ denote the embedding representations of slot type s_i and slot value v_i respectively. $c_i^t = \sum_{k=0}^{t-1} \alpha_i^k$ is a coverage vector, which is the sum of attention distributions over all previous decoder time steps and can be used to reduce repetition (See et al., 2017).

The source attention distribution α^t can be considered as the contribution of each source triple to the generation of the target word. Next we use α^t to compute two context vectors $\mathbf{L}_s^*$ and $\mathbf{L}_v^*$ as the representation of the slot types and values respectively:

$$\begin{aligned} \mathbf{L}_s^* &= \sum_{i=1}^n \alpha_i^t \mathbf{s}_i \\ \mathbf{L}_v^* &= \sum_{i=1}^n \alpha_i^t \mathbf{v}_i \end{aligned} \tag{1}$$

At step t, the vocabulary distribution P_{vocab} is computed with the context vectors $\mathbf{L}_s^*$, $\mathbf{L}_v^*$ and the decoder hidden state $\tilde{\mathbf{h}}^t$, using an affine-Softmax layer:

$$P_{vocab} = \text{Softmax}\left(V[\tilde{\mathbf{h}}^t; \mathbf{L}_s^*; \mathbf{L}_v^*] + b_{vocab}\right)$$

The loss function is computed as:

$$Loss = \sum_t \left\{ -\log P_{vocab}(y_t) + \lambda \sum_i \min\left(\alpha_i^t, c_i^t\right) \right\}$$

where $P_{vocab}(y^t)$ is the prediction probability of the ground truth token y_t. λ is a hyperparameter.
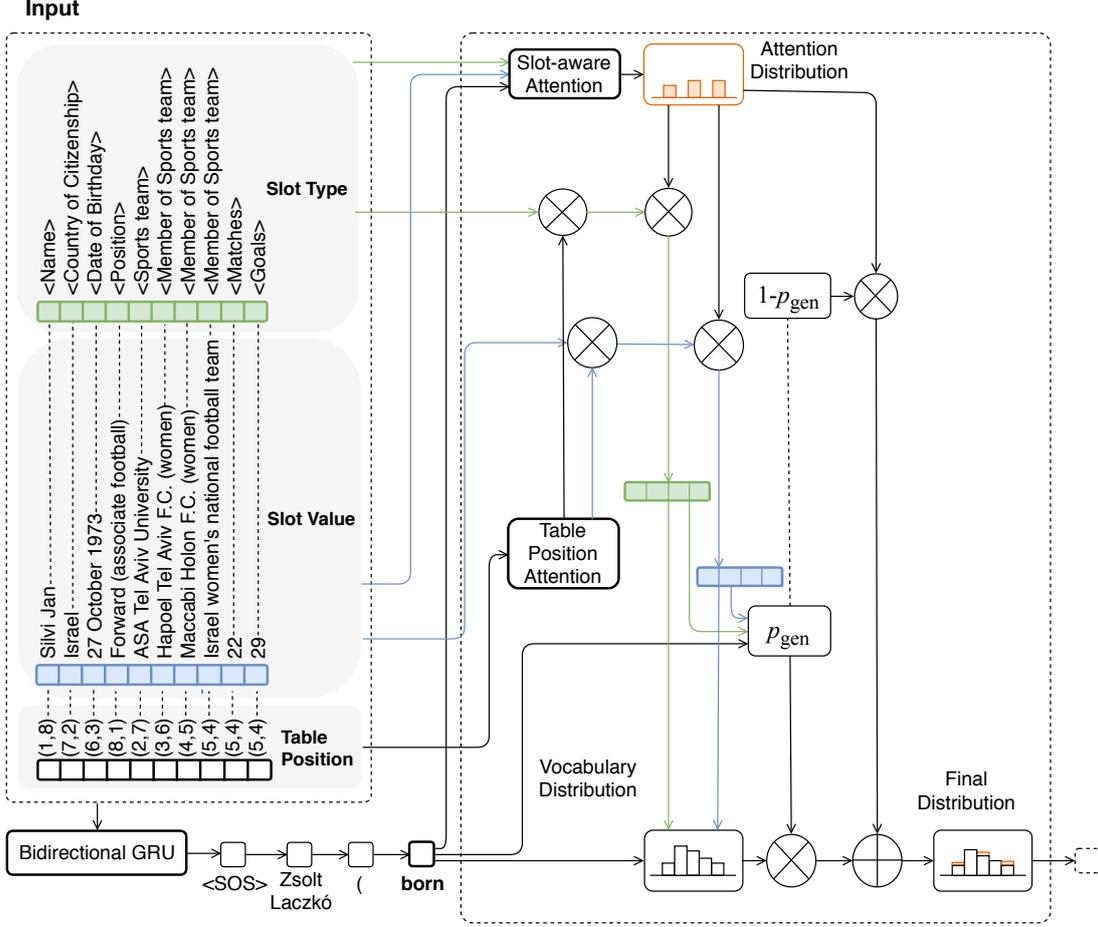
Figure 1: KB-to-Language Generation Model Overview

2.2 Table Position Self-attention

Although the sequence-to-sequence attention model takes into account the information of input triples, it still encodes the structured knowledge as sequential facts while ignoring the correlations between facts. In our task, multiple inter-dependent slots should be described within one sentence. For example, in Table 1, the sport team *Israel women's national football team* should be described together with *22* matches and *29* goals. Previous studies (Lin et al., 2017; Vaswani et al., 2017) applied self-attention on sentence level to capture the correlation between continuous tokens. Inspired by these approaches, we design a new table position based self-attention and incorporate it into the slot-aware attention.

In our task, since most triples are organized in temporal order, we use the row index r and the reverse row index $\hat{r}$ to denote the position information of each triple in the input KB. Given a structured KB as input: $L = [(s_1, v_1, (r_1, \hat{r}_1)), ..., (s_n, v_n, (r_n, \hat{r}_n))]$, we obtain a sequence of row index embeddings $R = [\mathbf{r}'_1, \mathbf{r}'_2, ..., \mathbf{r}'_n]$ with random initialization, where $\mathbf{r}'_i = [\mathbf{r}_i; \hat{\mathbf{r}}_i]$. We model the inter-dependencies among slots as a latent structure, where for each position i we assume it has a latent in-link and an out-link to denote where it is linked to or from. This assumption is similar to the structure attention applied in Liu and Lapata (2018), which assumes each word within a sentence can be a parent node or a child node in a latent tree structure. For each pair of slots i and j, we compute the attention score f_{ij} as follows:

$$g_{in} = \tanh\left(W_{in}\mathbf{r}'_i\right)$$

$$g_{out} = \tanh\left(W_{out}\mathbf{r}'_j\right)$$

$$f_{ij} = \text{Softmax}\left(g_{in}^\top W_g g_{out}\right)$$

where W_{in}, W_{out}, and W_g are learnable parameters. The attention score will not change during the decoding process.

f_{ij} can be viewed as the contribution from a context triple j to triple i. For each slot $\mathbf{s}_i$ and value $\mathbf{v}_i$, we obtain a context vector by collecting information from other slot types and their values:

$$\mathbf{s}_i^* = \sum\nolimits_{k=1}^n f_{ik}\mathbf{s}_k$$

$$\mathbf{v}_i^* = \sum\nolimits_{k=1}^n f_{ik}\mathbf{v}_k$$

We further encode position-aware representation of each slot type and value, and update their context vectors $\mathbf{L}_t^*$ and $\mathbf{L}_v^*$ in Equation 1 as:

$$\mathbf{L}_s^* = \sum\nolimits_{i=1}^n \alpha_i^t \mathbf{s}_i^*$$

$$\mathbf{L}_v^* = \sum\nolimits_{i=1}^n \alpha_i^t \mathbf{v}_i^*$$

2.3 Structure Generator

Traditional sequence-to-sequence models predict a target sequence by only selecting words from a vocabulary with a fixed size. However, in our task, we regard the slot value as a single information unit. Therefore, there is a certain amount of out-of-vocabulary (OOV) words during the test phase. Inspired by the pointer-generator (Gu et al., 2016; See et al., 2017), which is designed to automatically locate particular source words and directly copy them into the target sequence, we design a structure-aware generator as follows.

We first obtain a source attention distribution of all unique input slot values. Since one particular slot value may occur in the structure input for many times, we aggregate the attention weights for each unique slot value v_j from α_t and obtain its aggregated source attention distribution P_{source}^j by

$$P_{source}^j = \sum_{m \mid v_m = v_j} \alpha_m^t$$

The *gates* in neural networks act on the signals they receive, and block or pass on information based on its strength. In order to combine two types of attention distribution P_{source} and P_{vocab}, we compute a structure-aware gate $p_{gen} \in [0,1]$ as a soft switch between generating a word from the fixed vocabulary and copying a slot value from the structured input:

$$p_{gen} = \sigma\left(W_s^\top \mathbf{L}_s^* + W_v^\top \mathbf{L}_v^* + W_h^\top \tilde{\mathbf{h}}^t + W_y^\top \mathbf{y}^{t-1} + b_{gen}\right)$$

where $\mathbf{y}^{t-1}$ is the embedding of the previous generated token at time $t-1$, and σ is a Sigmoid function.

The final probability of a token y at time t can be computed by p_{gen}, P_{vocab} and P_{source}:

$$P(y_t) = p_{gen}P_{vocab} + (1 - p_{gen})P_{source}$$

The loss function, combining with the coverage loss (See et al., 2017), is presented as:

$$Loss = \sum_t \left\{ -\log P(y^t) + \lambda \sum_i \min\left(\alpha_i^t, c_i^t\right) \right\}$$

where $P(y^t)$ is the prediction probability of the ground truth token y. λ is a hyperparameter.

3 Experiments

3.1 Data

Using person and animal entities as case studies, we create a new dataset based on Wikipedia dump (2018/04/01) and Wikidata (2018/04/12) as follows: (1). Extract Wikipedia pages and Wikidata tables about person and animal entities, and align them according to their unique KB IDs. (2). For each Wikidata table, filter out the slot types of which frequency is less than 3. For each Wikipedia article, use its anchor links (clickable texts in hyperlinks) to locate all the entities and determine their KB IDs. (3). For each Wikidata table, search each value (including Number, Date) and entity contained in the table in the corresponding Wikipedia article according to its KB ID, and remove the values and entities which cannot be found in the corresponding Wikipedia article. (4). For each Wikipedia article, remove the sentences which contain no values, and remove sentences which only contain entities that do not exist in the Wikidata table. The remaining sentences will be taken as ground-truth reference descriptions. (5). Index the row numbers for each slot type according to their orders in the Wikidata table. The ground-truth structured KB is then created. (6). Build a fixed vocabulary for the whole corpus of ground-truth descriptions and label the words with frequency < 5 as OOV.

We further randomly shuffle and split the dataset into training (80%), development (10%) and test (10%) subsets for person and animal entities respectively. Table 3 shows the detailed statistics. Compared with the Wikibio dataset used in previous studies (Lebret et al., 2016; Sha et al., 2018; Liu et al., 2018), which contains one sentence only as the ground-truth description, our dataset contains multiple sentences to cover as many facts as possible in the input structured KB.

Entity type	# entity	# types before filtering	# types after filtering	# slots / sentence	# words / sentence	# slots / table	# words / entity	# sentence / entity
Person	100,000	109	76	1.9	16.8	8.0	70.9	4.2
Animal	6,216	30	12	1.3	17.1	3.2	42.2	2.5

Table 3: Data Statistics

Figure 2: KB Reconstruction based Evaluation (Scores for the example: Overall Slot Filling P=$\frac{6}{7}$=85.7%, R=$\frac{6}{11}$=54.5%, F1=66.7%; Inter-dependent Slot Filling P=$\frac{5}{7}$=71.4%, R=$\frac{5}{9}$=55.6%, F1=62.5%)

It makes the generation task more challenging, practical and interesting.

3.2 Evaluation Metrics

We apply the standard BLEU (Papineni et al., 2002), METEOR (Denkowski and Lavie, 2014), and ROUGE (Lin, 2004) metrics to evaluate the generation performance, because they can measure the content overlap between system output and ground-truth and also check whether the system output is written in sufficiently good English.

In addition, we can also consider natural language as the most expressive way for knowledge transmission via a *noisy channel*. If we are able to reconstruct the input KB from the generated description, our generator achieves a 100% success rate at knowledge propagation. We propose a *KB reconstruction based metric* as follows: for each entity, construct a KB from the generated paragraph, and compute precision, recall and F-score by comparing it with the input KB from two aspects: (1). **Overall Slot Filling**: If a pair of slot type and its slot value exists in both of the reconstructed KB and the input KB, it's considered as a correct slot. (2). **Inter-dependent Slot Filling**: If a row that consists one or multiple slot types and their slot values exist in both of the reconstructed KB and the input KB, it's considered as a correct row.

If the same slot/row is correctly described multiple times in the system generation output, it's only counted as correct once, i.e., redundant descriptions will be penalized. This metric is further

illustrated in Figure 2. It's similar to the relation extraction based generation evaluation metric proposed by (Wiseman et al., 2017) and entity/event extraction based metric proposed by (Whitehead et al., 2018; Lu et al., 2018). They compared automatic Information Extraction results from the reference description and the system generation output. However, the performance of state-of-the-art open-domain slot filling (Wu and Weld, 2010; Fader et al., 2011; Min et al., 2012; Xu et al., 2013; Angeli et al., 2015; Bhutani et al., 2016; Yu et al., 2017) is still far from satisfactory to serve as an automatic extraction tool for evaluating generation results. Therefore for the pilot study in this paper we manually reconstruct KBs from the generation output for evaluation. Notably none of the above automatic metrics is sufficient to capture adequacy, grammaticality and fluency of the generated descriptions. However extrinsic metrics such as system purpose and user task are expensive, while cheaper metrics such as human rating do not correlate with extrinsic metrics (Gkatzia and Mahamood, 2015). Moreover the task we address in this paper requires essential domain knowledge for a human user to assess the generated descriptions.

3.3 Baseline Models

We compare our approach with the following models: (1). **Seq2seq attention model** (Bahdanau et al., 2015). We concatenate slot types and values as a sequence, e.g., {*Name, Silvi Jan, Sports team, ASA Tel Aviv University, Hapoel Tel Aviv F.C. ...*} for Table 1, and apply the sequence to

sequence with attention model to generate a description. (2). **Pointer-generator** (See et al., 2017) which introduces a soft switch to choose between generating a word from the fixed vocabulary and copying a word from the input sequence. Here, we concatenate all slot values as the input sequence, e.g., {*Silvi Jan, ASA Tel Aviv University, Hapoel Tel Aviv F.C. ...*} for Table 1. (3). **Pointer-generator + slot type attention** which incorporates the slot type attention (Section 2.1) into the pointer-generator. We use the sequence of (slot type, slot value) pairs as input, e.g., {*(Name, Silvi Jan), (Sports team, ASA Tel Aviv University), (Sports team, Hapoel Tel Aviv F.C.) ...*} for Table 1.

3.4 Hyperparameters

Table 4 shows the hyperparameters of our model.

Parameter	Value
Vocabulary size (|s|+|v|)	46,776
Value\type embedding size	256
Position embedding size	5
Slot embedding size	522
Decoder hidden size	256
Coverage loss λ	1.5
Optimization	Adam (Hu et al., 2009)
Learning rate	0.001

Table 4: Hyperparameters

3.5 Results and Analysis

Table 5 shows the performance of various models with standard metrics. We can see that our attention mechanisms achieve consistent improvement. We conduct paired t-test between our proposed model and all the other baselines on 10 randomly sampled subsets. The differences are statistically significant with $p \leq 0.016$ for all settings.

As shown in Table 6 and Table 7, the KBs reconstructed from models with these two attention mechanisms achieve much higher quality.

Figure 3 and Figure 4 visualize the attentions applied to the walk-through example in Table 1.

Impact of Slot-aware Attention. The same string can be filled into various slots of multiple types. For example, dates, ages, the number of matches and goals can all be presented as numbers. The pointer network often mistakenly mixes them up. For example, it produces "*24 September 1979 was born 3 October 1903 in **17** on **33** October 1906*", where *33* should be the number
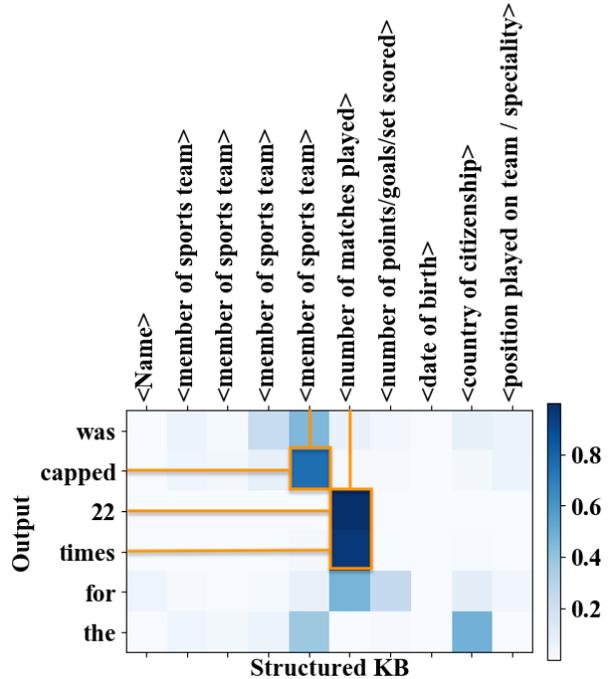

Figure 3: Slot Type Attention Visualization (Context words strongly associated with certain slot types receive high weights, e.g., *capped* to describe *member of sports team*, and *times* to describe *the number of matches played*.)

of matches and *17* should be the number of goals. In contrast our model with slot type attention correctly generates "*he made **33** appearances and scored **17** goals*". In addition, as mentioned earlier, the pointer network often produces redundant slot values because it loses control of slot types, e.g., "*He was born in the city of **Association football**. In the late 1990s he was appointed manager of the **Association football** team of the team.*".

Impact of Table Position Attention. The table position attention successfully captures interdependent slots, such as a membership with a sports team and its corresponding number of matches and games: "*Bill Sampy ... who played for **Sheffield United F.C. 41** times.*"; "*Giancarlo Antognoni ... he was also a member of the **Italy national football team** at the **1982 FIFA World Cup**.*".

Remaining Challenges. Some remaining errors are trivial to fix, such as fixing a country name to its adjective form when it appears right before a position slot (e.g., *Italian professional Association football player* instead of *Italy professional Association football player*). The KB reconstruction recall of person entities is relatively low mainly

Model	Person			Animal		
	BLEU	METEOR	ROUGE	BLEU	METEOR	ROUGE
Seq2seq	11.3	16.9	28.8	5.8	11.5	20.5
Pointer	17.2	21.1	37.4	6.6	13.7	37.8
+Type	23.1	22.2	39.5	**17.2**	**17.3**	42.8
+Type & Position	**23.2**	**23.4**	**42.0**	14.8	17.2	**45.0**

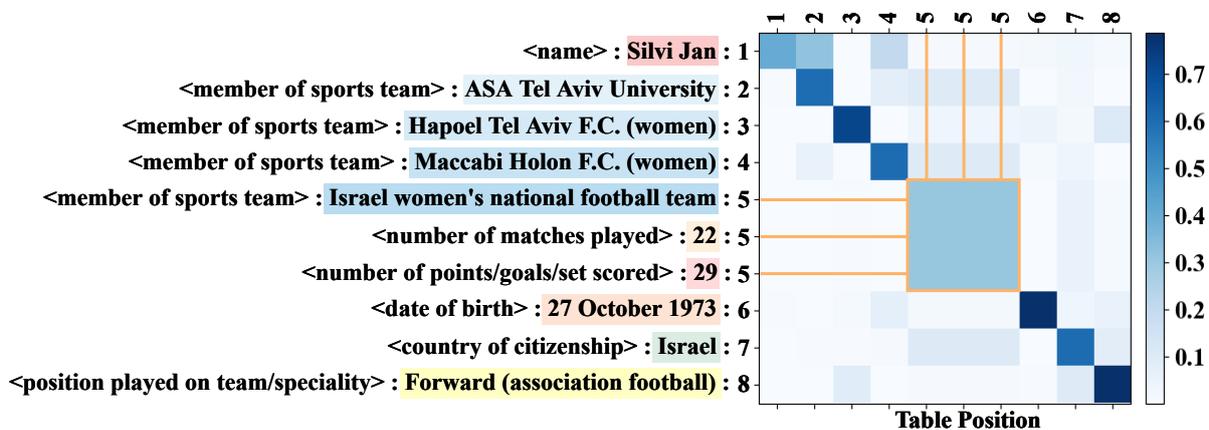Table 5: Generation Performance based on Standard Metrics %)

Figure 4: Table Position Self Attention Visualization (the highlighted inter-dependent slots appear in the same row and the same sentences, and thus they receive the same high weight.)

Model	Person			Animal		
	P	R	F1	P	R	F1
Seq2seq	74.6	29.3	42.0	82.5	27.8	41.6
Pointer	72.6	56.4	62.8	58.5	37.5	45.7
+Type	75.9	58.8	66.3	65.9	63.8	64.8
+Type & Position	**76.3**	**62.7**	**68.8**	**73.4**	**71.8**	**72.6**

Table 6: Overall Slot Filling Precision (P), Recall (R), F-score (F1) (%)

Model	Person			Animal		
	P	R	F1	P	R	F1
Seq2seq	74.7	30.0	43.4	82.5	27.9	41.7
Pointer	73.0	56.4	63.6	57.7	37.2	45.2
+Type	75.8	58.9	66.3	66.3	64.2	65.2
+Type & Position	**77.2**	**63.5**	**69.7**	**72.6**	**71.0**	**71.8**

Table 7: Inter-dependent Slot Filling Precision (P), Recall (R), F-score (F1) (%)

because we don't have enough training data for some rare slot types.

Contextual words generated by the LM introduces some incorrect facts, especially temporal expressions. For example, the generator does not have the commonsense knowledge that football players could not play before they were born: "*Aleksei Gasilin (born **1 March 1996**) is a Russian Association football Forward (association football). He made his professional debut in*

*the Russian Second Division in **1992** for Russia national under-19 football team.* ". Similarly, a football player would probably not be still active when he was already 72 years old: "*Basil Rigg (born **12 August 1926**) is a former Australian rules football Rigg played for the Perth Football Club in the Western Australia cricket team from **1998** to **1998**.*".

Our approach sometimes fails to detect person gender so as to generate incorrect pronouns. For animal entities, human writers are able to elaborate more details. For example, human writes the specific endemic places for *Brown treecreeper*: "*The bird endemic to eastern Australia has a broad distribution occupying areas from **Cape York Queensland** throughout **New South Wales** and **Victoria** to **Port Augusta** and the **Flinders Ranges South Australia**.*" while our system is only able to cover the generic location information "*It is endemic to **Australia**.*" from the input KB.

4 Related work

Our task is similar to the WebNLG challenge generating text from DBPedia data (Gardent et al., 2017a). Previous approaches on generating natural language sentences from structured input KB can be divided into two categories: the first is to

induce templates and then fill appropriate content into slots (Kukich, 1983; Cawsey et al., 1997; Angeli et al., 2010; Duma and Klein, 2013; Konstas and Lapata, 2013a; Flanigan et al., 2016a). These methods can generate high-quality descriptions but heavily rely on information redundancy to create templates. The second category is to directly generate a sequence of words using language model (Belz, 2008; Chen and Mooney, 2008; Liang et al., 2009; Angeli et al., 2010; Konstas and Lapata, 2012a,b, 2013a,b; Mahapatra et al., 2016) or deep neural networks (Sutskever et al., 2011; Wen et al., 2015; Kiddon et al., 2016; Mei et al., 2016; Gardent et al., 2017b; Wiseman et al., 2017; Wang et al., 2018; Song et al., 2018). Several studies (Lebret et al., 2016; Chisholm et al., 2017; Kaffee et al., 2018a,b; Liu et al., 2018; Sha et al., 2018) generate a person's biography from an input structure, which are closely related to our task. However, instead of modeling the input structure as a sequence of facts and generating one sentence only, we introduce a table position self-attention, inspired from structure attention (Lin et al., 2017; Kim et al., 2017; Vaswani et al., 2017; Shen et al., 2018a,b), to capture the dependencies among facts and generate a paragraph to describe all facts.

In contrast to some recent work on converting structured Abstract Meaning Representation (Banarescu et al., 2013) into natural language (Pourdamghani et al., 2016; Flanigan et al., 2016b), our task requires us to capture inter-dependent relation links in a knowledge base and use them to generate multiple sentences in most cases. Our work is also related to attention mechanisms for sequence-to-sequence generation (Bahdanau et al., 2015; Mei et al., 2016; Ma et al., 2017). Different from previous studies, our task requires the slot type and slot value to appear in the generated sentences in pairs. Thus we design a slot-aware attention to obtain two context vectors for both slot type and slot value simultaneously. To deal with OOV words, we use a structure generator, which is similar to the pointer-generator networks (Vinyals et al., 2015; Luong et al., 2015; Gulcehre et al., 2016; See et al., 2017) and copy mechanism (Gu et al., 2016).

5 Conclusions and Future Work

We develop an effective generator to produce a natural language description about an input knowledge base. Our experiments show that two attention mechanisms focusing on slot type and table position advance state-of-the-art on this task, and provide a KB reconstruction F-score up to 73%. We propose a new KB reconstruction based evaluation metric which can be used for other knowledge-driven NLG tasks such as news image/video captioning. In the future, we aim to address the remaining challenges as summarized in Section 3.5, and tackle the setting where multiple facts of the same slot type are not presented in temporal order in the input KB. We also plan to extend the framework to cross-lingual cross-media generation, namely to produce a foreign language description or an image/video about the KB.

References

Gabor Angeli, Percy Liang, and Dan Klein. 2010. A simple domain-independent probabilistic approach to generation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*.

Gabor Angeli, Melvin Jose Johnson Premkumar, and Christopher D Manning. 2015. Leveraging linguistic structure for open domain information extraction. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *International Conference on Learning Representations*.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract meaning representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*.

Anja Belz. 2008. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering*, 14(4):431–455.

Nikita Bhutani, HV Jagadish, and Dragomir Radev. 2016. Nested propositions in open information extraction. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Nadjet Bouayad-Agha, Gerard Casamayor, and Leo Wanner. 2013. Natural language generation in the context of the semantic web. *Semantic Web Journal*.

Alison J Cawsey, Bonnie L Webber, and Ray B Jones. 1997. Natural language generation in health care. *Journal of the American Medical Informatics Association*, 4.

David L Chen and Raymond J Mooney. 2008. Learning to sportscast: a test of grounded language acquisition. In *Proceedings of the 25th international conference on Machine learning*.

Andrew Chisholm, Will Radford, and Ben Hachey. 2017. Learning to generate one-sentence biographies from wikidata. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics*.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*.

Misha Denil, Loris Bazzani, Hugo Larochelle, and Nando de Freitas. 2012. Learning where to attend with deep architectures for image tracking. *Neural Computation*, 24(8):2151–2184.

Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*.

Daniel Duma and Ewan Klein. 2013. Generating natural language from linked data: Unsupervised template extraction. In *Proceedings of the 10th International Conference on Computational Semantics*.

Anthony Fader, Stephen Soderland, and Oren Etzioni. 2011. Identifying relations for open information extraction. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*.

Jeffrey Flanigan, Chris Dyer, Noah A Smith, and Jaime Carbonell. 2016a. Generation from abstract meaning representation using tree transducers. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Jeffrey Flanigan, Chris Dyer, Noah A. Smith, and Jaime Carbonell. 2016b. Generation from abstract meaning representation using tree transducers. In *Proc. the 15th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT2016)*.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017a. Creating training corpora for micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Vancouver, Canada. Association for Computational Linguistics.

Claire Gardent, Anastasia Shimorina, Shashi Narayan, and Laura Perez-Beltrachini. 2017b. Creating training corpora for nlg micro-planners. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Dimitra Gkatzia and Saad Mahamood. 2015. A snapshot of nlg evaluation practices 2005 - 2014. In *Proceedings of the 15th European Workshop on Natural Language Generation (ENLG)*.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Caglar Gulcehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Chonghai Hu, Weike Pan, and James T. Kwok. 2009. Accelerated gradient methods for stochastic optimization and online learning. In *Advances in Neural Information Processing Systems 22*, pages 781–789.

Lucie-Aimée Kaffee, Hady Elsahar, Pavlos Vougiouklis, Christophe Gravier, Frédérique Laforest, Jonathon Hare, and Elena Simperl. 2018a. Learning to generate wikipedia summaries for underserved languages from wikidata. *arXiv preprint arXiv:1803.07116*.

Lucie-Aimée Kaffee, Hady Elsahar, Pavlos Vougiouklis, Christophe Gravier, Frédérique Laforest, Jonathon Hare, and Elena Simperl. 2018b. Mind the (language) gap: Generation of multilingual wikipedia summaries from wikidata for articleplaceholders. In *Proceedings of the 15th European Semantic Web Conference*.

Chloé Kiddon, Luke Zettlemoyer, and Yejin Choi. 2016. Globally coherent text generation with neural checklist models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Yoon Kim, Carl Denton, Luong Hoang, and Alexander M Rush. 2017. Structured attention networks. In *International Conference on Learning Representations*.

Ioannis Konstas and Mirella Lapata. 2012a. Concept-to-text generation via discriminative reranking. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*.

Ioannis Konstas and Mirella Lapata. 2012b. Unsupervised concept-to-text generation with hypergraphs. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Ioannis Konstas and Mirella Lapata. 2013a. A global model for concept-to-text generation. *Journal of Artificial Intelligence Research*, 48:305–346.

Ioannis Konstas and Mirella Lapata. 2013b. Inducing document plans for concept-to-text generation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*.

Karen Kukich. 1983. Design of a knowledge-based report generator. In *Proceedings of the 21st annual meeting on Association for Computational Linguistics*.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*.

Percy Liang, Michael I Jordan, and Dan Klein. 2009. Learning semantic correspondences with less supervision. 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*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Proceedings of Text Summarization Branches Out*.

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *International Conference on Learning Representations*.

Tianyu Liu, Kexiang Wang, Lei Sha, Baobao Chang, and Zhifang Sui. 2018. Table-to-text generation by structure-aware seq2seq learning. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*.

Yang Liu and Mirella Lapata. 2018. Learning structured text representations. *Transactions of the Association for Computational Linguistics*, 6:63–75.

Di Lu, Spencer Whitehead, Lifu Huang, Heng Ji, and Shih-Fu Chang. 2018. Entity-aware image caption generation. In *Proc. 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP2018)*.

Thang Luong, Ilya Sutskever, Quoc Le, Oriol Vinyals, and Wojciech Zaremba. 2015. Addressing the rare word problem in neural machine translation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*.

Shuming Ma, Xu Sun, Jingjing Xu, Houfeng Wang, Wenjie Li, and Qi Su. 2017. Improving semantic relevance for sequence-to-sequence learning of chinese social media text summarization. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Joy Mahapatra, Sudip Kumar Naskar, and Sivaji Bandyopadhyay. 2016. Statistical natural language generation from tabular non-textual data. In *Proceedings of the 9th International Natural Language Generation conference*.

Hongyuan Mei, Mohit Bansal, and Matthew R. Walter. 2016. What to talk about and how? selective generation using lstms with coarse-to-fine alignment. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Bonan Min, Shuming Shi, Ralph Grishman, and Chin-Yew Lin. 2012. Ensemble semantics for large-scale unsupervised relation extraction. In *Proceedings Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

Nima Pourdamghani, Kevin Knight, and Ulf Hermjakob. 2016. Generating english from abstract meaning representations. In *Proc. The International Natural Language Generation conference (INLG2016)*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Lei Sha, Lili Mou, Tianyu Liu, Pascal Poupart, Sujian Li, Baobao Chang, and Zhifang Sui. 2018. Order-planning neural text generation from structured data. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*.

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Shirui Pan, and Chengqi Zhang. 2018a. Disan: Directional self-attention network for rnn/cnn-free language understanding. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*.

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, and Chengqi Zhang. 2018b. Bi-directional block self-attention for fast and memory-efficient sequence modeling. In *International Conference on Learning Representations*.

Linfeng Song, Yue Zhang, Zhiguo Wang, and Daniel Gildea. 2018. A graph-to-sequence model for amr-to-text generation. *arXiv preprint arXiv:1805.02473*.

Ilya Sutskever, James Martens, and Geoffrey E Hinton. 2011. Generating text with recurrent neural networks. In *Proceedings of the 28th International Conference on Machine Learning*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 22*.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2692–2700.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Qingyun Wang, Zhihao Zhou, Lifu Huang, Spencer Whitehead, Boliang Zhang, Heng Ji, and Kevin Knight. 2018. Paper abstract writing through editing mechanism. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned lstm-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*.

Spencer Whitehead, Heng Ji, Mohit Bansal, Shih-Fu Chang, and Clare Voss. 2018. Incorporating background knowledge into video description generation. In *Proc. 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP2018)*.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*.

Fei Wu and Daniel S. Weld. 2010. Open information extraction using wikipedia. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*.

Ying Xu, Mi-Young Kim, Kevin Quinn, Randy Goebel, and Denilson Barbosa. 2013. Open information extraction with tree kernels. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Dian Yu, Lifu Huang, and Heng Ji. 2017. Open relation extraction and grounding. In *Proceedings of the 8th International Joint Conference on Natural Language Processing*.