Towards Comprehensive Description Generation from Factual Attribute-value Tables

Tianyu Liu[1], **Fuli Luo**[1], **Pengcheng Yang**[1], **Wei Wu**[1], **Baobao Chang**[1,2] and **Zhifang Sui**[1,2]
[1]MOE Key Lab of Computational Linguistics, School of EECS, Peking University
[2]Peng Cheng Laboratory, Shenzhen, China
{tianyu0421, luofuli, yang_pc, wu.wei, chbb, szf}@pku.edu.cn

Abstract

The comprehensive descriptions for factual attribute-value tables, which should be accurate, informative and loyal, can be very helpful for end users to understand the structured data in this form. However previous neural generators might suffer from *key attributes missing*, *less informative* and *groundless information* problems, which impede the generation of high-quality comprehensive descriptions for tables. To relieve these problems, we first propose force attention (FA) method to encourage the generator to pay more attention to the uncovered attributes to avoid potential *key attributes missing*. Furthermore, we propose reinforcement learning for information richness to generate *more informative* as well as more *loyal* descriptions for tables. In our experiments, we utilize the widely used `WIKIBIO` dataset as a benchmark. Additionally we create `WB-filter` based on `WIKIBIO` to test our model in the simulated user-oriented scenarios, in which the generated descriptions should accord with particular user interests. Experimental results show that our model outperforms the state-of-the-art baselines on both automatic and human evaluation.

1 Introduction

Generating descriptions for the factual attribute-value tables has attracted widely interests among NLP researchers especially in a neural end-to-end fashion (e.g. Lebret et al. (2016); Liu et al. (2018); Sha et al. (2018); Bao et al. (2018); Puduppully et al. (2018); Li and Wan (2018); Nema et al. (2018)) as shown in Fig 1a. For broader potential applications in this field, we also simulate user-oriented generation, whose goal is to provide comprehensive generation for the *selected attributes* according to particular user interests like Fig 1b.

Attribute	**Value**
Birthplace	*Utah, America*
Position	*forward (soccer player)*

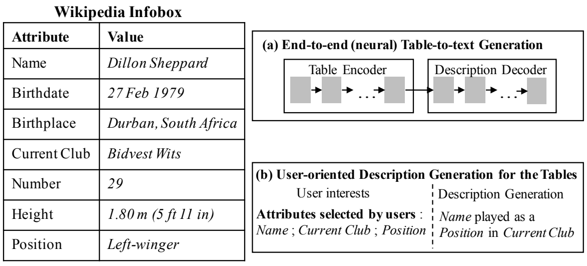
Comprehensive: A *Utah soccer player* who plays as *forward*
Missing Key Attri.: A *soccer player* who plays as *forward*
Groundless info: A *Utah forward* in the national team
Less Informative: An *American forward*

Table 1: An example for comprehensive generation. Suppose we only have two attribute-value tuples, the underlined content is *groundless information* not mentioned in source tables.

However, we find that previous models might miss key information and generate less informative and groundless content in its generated descriptions towards source tables. For example, in Table 1, the 'missing key attribute' case doesn't mention where the player comes from (*birthplace*) while the 'less informative' one chooses *American* rather than *Utah*. The case with groundless information contains '*in the national team*' which is not mentioned in the source attributes. Although the 'key points missing' problem exists in many text-to-text and data-to-text datasets, for large-scale structured tables with vast heterogeneous attributes such as Wikipedia infoboxes, 'Key attribute missing' and 'less informative' problems might be even more challenging. As the key attributes, like the '*position*' of a basketball player or the '*political party*' of a senator, are very likely to be unique features to particular tables, which usually appear much less frequently and are seldomly mentioned than the common attributes like '*Name*' and '*Birthdate*'. The 'groundless information', which is also known as the 'hallucination' problem, remains a long-standing problem in NLG.

In this paper, we show that our model can generate more accurate and informative descriptions with less groundless content for tables. Firstly we design a force-attention (FA) method to encourage the decoder to pay more attention to the un-

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 5985–5996
Florence, Italy, July 28 - August 2, 2019. ©2019 Association for Computational Linguistics



Wikipedia Infobox

Attribute	Value
Name	*Dillon Sheppard*
Birthdate	*27 Feb 1979*
Birthplace	*Durban, South Africa*
Current Club	*Bidvest Wits*
Number	*29*
Height	*1.80 m (5 ft 11 in)*
Position	*Left-winger*

(a) End-to-end (neural) Table-to-text Generation
Table Encoder
Description Decoder
(b) User-oriented Description Generation for the Tables
User interests
Description Generation
Attributes selected by users :
Name ; Current Club ; Position
Name played as a
Position in Current Club

Figure 1: The end-to-end (a) and user-oriented table-to-text generation (b) for an infobox (left) in WIKIBIO.

covered attributes to avoid potential *key attributes missing* by both stepwise and global constraints. In addition, we define the 'information richness' measurement of the generated descriptions to the source tables. Based on that, we use the reinforcement learning to encourage the generator to cover infrequent and rarely mentioned attributes as well as generate *more informative* descriptions with *less groundless content*.

We test our models on two settings:

1) For neural table-to-text generation like Fig 1a, we test our model on WIKIBIO (Lebret et al., 2016), a crawled dataset from Wikipedia with paired infoboxes and associated descriptions. It is a widely used benchmark dataset for description generation for factual attribute-value tables and also a quite meaningful testbed in the real-world scenarios with vast and heterogenous attributes.

2) To test our model in the user-oriented setting, we filter WIKIBIO to form WB-filter. In this setting, we suppose all attributes in the source tables of WB-filter are selected by users that should be covered in the corresponding descriptions. We try to make sure the gold descriptions in WB-filter cover all the attributes of the source tables in this condition. Details in Sec 4.

Both automatic and human evaluation show that our model relieves the 3 problems (Table 1) and helps the generator to produce accurate, informative and loyal descriptions. We also achieve the state-of-the-art performance on the end-to-end table description and the user-oriented generation tasks.

The remainder of this paper is organized as follows. We first introduce how we formulate table-to-text generation into encoder-decoder framework in Sec 2. After that, we discuss force-attention method (Sec 3.1) and richness-oriented reinforcement learning (Sec 3.2), which are motivated by the three goals we set up for comprehensive table descriptions (Table 1). Then we demonstrate how and why we create WB-filter (Sec 4.1) as well as evaluations (Sec 4.2), experimental configurations (Sec 4.3 and 4.4), case studies and visualizations (Sec 4.5) and error analysis (Sec 4.6).

2 Background: Table-to-Description

2.1 Table Encoder

Given a structured table like Fig 1 (left), we model the attribute-value tuples in the table as a sequence of words with related attribute names. After serializing all the words in the '*Value*' columns, for the i-th word in the table $x_i^{a_k}$ whose attribute is a_k (the k-th attribute), we use the attribute name a_k and the word's position in that tuple to locate the word (Lebret et al., 2016). Specifically we utilize a triple $z_i^{a_k} = \{a_k, p_{i+}^{a_k}, p_{i-}^{a_k}\}$ to represent the structure information for word $x_i^{a_k}$, in which $p_{i+}^{a_k}$ and $p_{i-}^{a_k}$ are the positions of $x_i^{a_k}$ counted from the beginning and end of a_k, respectively. For example, for the '*Birthplace*' attribute in Fig 1 (left), we can use triples {*birthplace*,1,4} and {*birthplace*,4,1} to represent the structure information for the words '*Durban*' [1] and '*Africa*'. We concatenate the word x_t and its structure representation z_t at the t-th time step and feed them into LSTM (Hochreiter and Schmidhuber, 1997) unit to encode the table. $h_t = \text{LSTM}([x_t; z_t], h_{t-1})$ is the t-th hidden state among the encoder states $H = \{h_t\}_{t=1}^{T}$. In the following sections, we might omit the superscript of $x_i^{a_k}$ if it is not necessary.

2.2 Description Decoder

For the generated description y^*, the generated token y_t^* at the t-th time step is predicted based on all the previously generated tokens $y_{<t}^*$ before y_t^* and the hidden states H of the table encoder:

$$P(y_t^*|H, y_{<t}^*) = \text{softmax}(W_s \odot tanh(W_t[s_t, c_t])) \tag{1}$$

where $\odot$ is element-wise product, $s_t = \text{LSTM}(y_{t-1}^*, s_{t-1})$ is the t-th hidden state of the decoder. $c_t = \sum_{i=1}^{T} \alpha_t^i h_i$ is the context vector, which is the weighted sum of encoder hidden states according to the attention matrix α. $\alpha_t^i \propto e^{g(s_t, h_i)}$ is the attention element of the t-th decoder state s_t and the i-th encoder state h_i.

[1] More concretely, '*Durban*' is the first word counted from the begining and also the fourth word counted from the end of *birthplace* attribute in Fig 1 (left).

where $g(s_t, h_i)$ is a relevance score between s_t and h_i. We use Bahdanau-style attention mechanism (Bahdanau et al., 2014) to calculate $g(s_t, h_i)$.

$$g(s_t, h_i) = tanh(W_p h_i + W_q s_t + b) \qquad (2)$$

W_s, W_t, W_p, W_q are learnable parameters.

3 Comprehensive Table Description

The problems listed in Table 1 not only prevent the generators to produce comprehensive descriptions for selected entries in the tables (Fig 1b), but also prevent the generator to produce informative, accurate and loyal table descriptions (Fig 1a). So we propose two methods: *force-attention* (FA) and *richness-oriented* reinforcement learning to produce accurate, informative and loyal descriptions.

3.1 Force-Attention Module

For 'missing key attributes' problem (Table 1), we find that the generator usually focuses on particular attributes while the other attributes have relatively low attention values in the entire decoding procedure. So force attention method is proposed to guide the decoder to pay more attention to the previous uncovered attributes with low attention values to avoid potential key attribute missing. Note that FA method focuses on attribute-level coverage rather than word-level coverage (Tu et al., 2016) as our goal is to reduce the 'missing key attributes' phenomenons instead of building rigid word-by-word alignment between tables and descriptions.

Stepwise Forcing Attention: We define attribute-level attention $\beta_t^{a_k} = \text{avg}(\sum_{x_i \in a_k} \alpha_t^i)$ at the t-th step for attribute a_k as the average value of the word-level attention values for the words in that attribute. The word-level coverage is defined as the sum of attention vector before the t-th step $\theta_t^i = \theta_{t-1}^i + \alpha_t^i$ (Tu et al., 2016). In the similar way, we define the attribute-level coverage $\gamma_t^{a_k} = \gamma_{t-k}^{a_k} + \beta_t^{a_k}$ as the overall attention for attribute a_k before the t-th time step. The average word-level and attribute-level coverage are $\overline{\theta_t^i} = \theta_t^i / t$ and $\overline{\gamma_t^{a_k}} = \gamma_t^{a_k} / t$, respectively.

Then we propose stepwise attention forcing, which explicitly guides the decoder to pay more attention on the uncovered attributes by calculating a new context vector $\widetilde{c}_t = \pi c_t + (1 - \pi) v_t$ to make compensation for the ignored attributes in the previous time steps. π is a learnable vector.

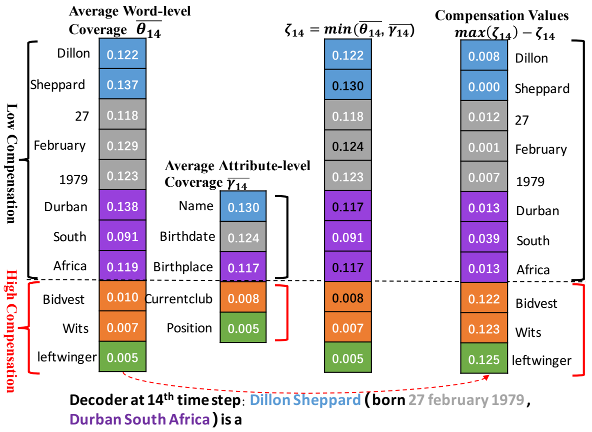


Figure 2: Stepwise forcing attention at the 14-th step for the filtered version of the original infobox in Fig 1 in the `WB-filter` dataset (The next word is '*leftwinger*'). The uncovered attributes like '*currentclub*' and '*position*' (marked in orange and green) get high attention compensation (rightmost). Note that word '*Sheppard*' does not get any compensation (rightmost) because it has got high attention in the previous steps.

v_t is a compensation vector for the low-coverage attributes:

$$v_t = \sum_{i=1}^{T} (\max(\zeta_t) - \zeta_t^i) h_i ; \zeta_t^i = \min(\overline{\theta_t^i}, \overline{\gamma_t^{a_k}}) \qquad (3)$$

ζ_t is the modified average word-level coverage regarding the average attribute-level coverage as the upper bound to avoid excessive compensation.

Fig 2 shows a running example. The motivation behind is that we want the decoder to pay enough attention to all the attributes in the whole decoding process, which prevents missing key attributes because of the low attention value on them. Thus we make compensation for the previous uncovered attributes (like '*currentclub*' and '*position*' in Fig 2) by v_t at the t-th time step.

Global Forcing Attention: Inspired by the soft-attention constraint of (Xu et al., 2015) which encourages the generator to pay equal attention to every part of the image while generating image captions, we propose global forcing attention to avoid insufficient or excessive attention on certain attributes by adding the following loss to the prime seq2seq loss.

$$\mathcal{L}_{FA} = \lambda \sum_{k=1}^{K} [\overline{\gamma_{-1}^{a_k}} - 1/K]^2 \qquad (4)$$

where K is the number of attributes in the table, λ is a hyper-parameter which is set to 0.3 based on evaluations on the validation data. $\overline{\gamma_{-1}^{a_k}}$ is the average attribute-level coverage for attribute a_k at the last time step.

3.2 Reinforced Richness-oriented Learning

We also propose a reinforcement learning framework which encourages the generator to cover rare and seldom mentioned words and attributes in the table. The experiments and case studies show its effectiveness to deal with the 'groundless information' and 'less informative' problems in Table 1.

3.2.1 Information Richness

The information richness (Eq 5) is the multiplication of the attribute-level and word-level richness of the descriptions towards the source tables.

Attribute-level Information Richness: Different tables which describe different objects are always featured by the unique attributes in the table. For example, a sportsman often has the attributes like '*position*', '*debutyear*'. The information in the unique attributes is harder to capture than that in the common attributes like '*name*', '*birthdate*' as the latters are very frequent in the training set. We define the information richness for an attribute a_i as $f(a_k) = [freq(a_k)]^{-1}$ by calculating its frequency in the training set.

Word-level Information Richness: The unique words in the tables are more likely to be informative, such as a specific location, name or book. To calculate the word-level information richness, we firstly lemmatize all the words in the tables and filter the words with a stop-words list which including prepositions, symbols and numbers, etc. Then we randomly sample 5 synonyms of the certain word from WordNet (Miller, 1995). Finally, we calculate the word-level richness $w(x_i^{a_k})$ for the i-th word in attribute a_k by averaging the tf-idf values of $x_i^{a_k}$ and its synonyms in the training set.

For a generated description y^*, we lemmatize all the words in y^* to get $\overline{y^*}$. Then we calculate the information richness based on the related source table with T words and the gold description y, respectively.

$$Rich(\overline{y^*}) = \frac{\sum_{i=1}^{T}[f(a_k) \cdot w(x_i^{a_k}) \cdot \mathbb{1}\{\tilde{x}_i^{a_k} \in \overline{y^*}\}]}{\sum_{i=1}^{T}[f(a_k) \cdot w(x_i^{a_k})]} \quad (5)$$

in which $\tilde{x}_i^{a_k}$ represents any word among $x_i^{a_k}$ and its synonyms in the table. The information richness measures the ratio of covered information in the table by the description.

3.2.2 Reinforcement Learning

Reward Function: Different from previous models which only measures how well the generated sentences match the target sentences, we design a mixed reward R_{mix} which contains both the BLEU-4 scores and the information richness of the generated descriptions towards the source tables.

$$R_{mix} = \lambda R_{info} + (1 - \lambda) R_{BLEU} \quad (6)$$

λ is set to 0.4 and 0.6 for `WIKIBIO` and `WB-filter` based on evaluations on the validation data. Fig 6 shows how we choose λ.

Training Algorithm: We use the REINFORCE algorithm (Williams, 1992) to learn an agent to maximize the reward function R_{mix}. The training loss of sequence generation is defined as the negative expected reward.

$$\mathcal{L}_{RL} = -\mathbb{E}_{y^s \sim p_\phi}[r(y^s) \cdot log(P_\phi(y^s))] \quad (7)$$

where $P_\phi(y^s)$ is the agent's policy, i.e. the word distribution of description decoder (Eq 1), and $r(\cdot)$ is the reward function defined in Eq 6. In the implementation, y^s is a sequence that can be sampled from P_ϕ by Monte-Carlo sampling $y^s = \{y_1^s, y_2^s, \cdots, y_{|Y|}^s\}$. The policy gradients for Eq 7 can be calculated as:

$$\nabla_\phi \mathcal{L}_{RL} = \lambda \nabla_\phi R_{info} + (1 - \lambda) \nabla_\phi R_{BLEU} \quad (8)$$

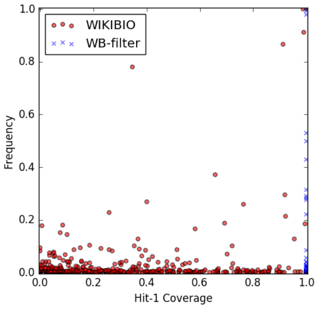
We use self-critical sequence training method (Rennie et al., 2017; Paulus et al., 2017) to reduce the variance of gradients by subtracting a baseline reward for the mix reward in Eq 6.

$$\nabla_\phi R_{BLEU} \approx -[B(y^s, y) - B(y^g, y)] \nabla_\phi log(P_\phi(y^s)) \quad (9)$$

where $B(a, b)$ is the BLEU score of sequence a compared with sequence b, y^g is a generated sequence using greedy search. To calculate the information richness reward R_{info} for the lemmatized sampled sequence $\overline{y^s}$, we use the information richness (Eq 5) of the related lemmatized gold description $\overline{y}$ towards the source table as the baseline reward.

$$\nabla_\phi R_{info} \approx -[Rich(\overline{y^s}) - Rich(\overline{y})] \nabla_\phi log(P_\phi(y^s)) \quad (10)$$

For more technical details, we refer the interested readers to (Williams, 1992; Ranzato et al., 2015; Rennie et al., 2017).



Dataset	WIKIBIO	WB-filter
# instances	728321	88287
# Tokens per Bio	26.1	30.2
# Tokens per Table	53.1	20.8
# Attri. per Table	19.7	6.3
# Word overlap	9.5	12.1

Figure 3: The 'coverage-frequency' figure (left) (each point represents an attribute) shows that many attributes have very low coverage and low frequency in the WIKIBIO dataset. Due to our filtering, the attributes in WB-filter have 100% Hit-1 coverage (Sec 4.2) and more overlapping words with the original tables as shown in the data statistics (right).

4 Experiments

4.1 Datasets

We use two datasets to test our model in the context of end-to-end table description generation and comprehensive generation for selected attributes in user-oriented scenario.

For end-to-end description generation, we use WIKIBIO dataset (Lebret et al., 2016) as the benchmark dataset, which contains 728,321 articles from English Wikipedia (Sep 2015) and uses the first sentence of each article as the description.

To test our model in the user-oriented scenario, we filtered the WIKIBIO dataset to form a new dataset WB-filter. To simulate the user interests, we first select the top 100 frequent [2] attributes in WIKIBIO. After that we manually filter irrelevant attributes (like 'caption', 'website' or 'signature') and merge identical attributes (like 'article title' and 'name') to avoid repetition. Then we leave out all the remaining attributes in the tables and filter the instances in WIKIBIO whose descriptions can not cover the selected attributes to form WB-filter. To achieve this, we firstly lemmatize all the tokens in the infoboxes as well as those in the related gold biographies and filter them by a stop-words list, then we randomly retrieve 5 synonyms for every word in the infoboxes from WordNet. Finally we make sure the gold biographies cover at least one word (or its synonym) for *every* attribute-value tuple among the chosen attributes and filter the unqualified instances in WIKIBIO.

The 'frequency-coverage' figure in Fig 3 shows 1) The filtering ensures that the WB-filter dataset achieves 100% Hit-1 coverage. 2) The WIKIBIO dataset suffers from both 'low frequency' and 'low coverage' problems, which means some key attributes in the tables are seldom mentioned by the descriptions. The cause of 'low coverage' problem is the loosely alignment between structured data and related descriptions. The two datasets are divided in to training (80%), testing (10%) and validation (10%) sets.

4.2 Evaluation Metrics

Automatic Metrics: Following the previous work (Lebret et al., 2016; Sha et al., 2018; Liu et al., 2018), we use BLEU-4 (Papineni et al., 2002) and ROUGE-4 (F measure) (Lin, 2004) for automatic evaluation. Furthermore, to evaluate how the generated biographies cover the key points in the infoboxes, we also use information richness (Eq 5) as one of our automatic evaluation. 'Hit at least 1 word' for an attribute means that a biography has at least one overlapping word with the words (or their synonyms) in that attribute, which are lemmatized and filtered by a stop-words list like the way we get WB-filter in Sec 4.1. '*HIT-1 coverage*' for an attribute is the ratio of the instances involving that attribute whose biographies 'Hit at least 1 word' in that attribute.

Human Evaluation: Since automatic evaluations like BLEU may not be reliable for NLG systems (Callison-Burch et al., 2006; Reiter and Belz, 2009; Reiter, 2018). We use human evaluation which involves the generation fluency, coverage (how much given information in the infobox is mentioned in the related biography) and correctness (how much false or irrelevant information is mentioned in the biography). We firstly sampled 300 generated biographies from the generators for human evaluation. After that, we hired 3 third-party crowd-workers who are equipped with sufficient background knowledge to rank the given biographies. We present the generated descriptions to the annotators in a randomized order and ask them to be objective and not to guess which system a particular generated case is from. Two biographies may have the same ranking if it is hard to decide which one is better. The Pearson correlations of inter-annotator agreement are 0.76 and 0.71 (Table 3) on WIKIBIO and WB-filter, re-

[2] In this setup, the reason of choosing high frequent attributes is to ensure enough training instances in WB-filter for data-driven methods.

spectively.

4.3 Experimental Details

Following previous work (Liu et al., 2018). For WIKIBIO We select the most frequent 20,000 words and 1480 attributes in the training set as the word and attribute vocabulary. We tune the hyper-parameters based on the model performance on the validation set. The dimensions of word embedding, attribute embedding, position embedding and hidden unit are 500, 50, 600, 10 respectively. The batch size, learning rate and optimizer for both two datasets are 32, 5e-4 and Adam (Kingma and Ba, 2014), respectively. We use Xavier initialization (Glorot and Bengio, 2010) for all the parameters in our model. The global constraint of force-attention (Eq 4) is adapted after 4 and 1.5 epochs of training to avoid hurting the primary loss for the WIKIBIO and WB-filter datasets, respectively. Before the richness-oriented reinforced training, the neural generator is pre-trained 8 and 4 epochs for the WIKIBIO and WB-filter datasets (with or without force-attention module), respectively. We replace UNK tokens with the most relevant token in the source table according to the attention matrix (Jean et al., 2015).

4.4 Baselines

KN & Template KN: A template-based Kneser-Ney (KN) language model (Heafield et al., 2013) The extracted template for Table 1 is "*name_1 name_2* (born *birthdate_1* $\cdots$ ". During inference, the decoder is constrained to emit words from the vocabulary or the special tokens in the tables.

Table NLM: Lebret et al. (2016) proposed a neural language model Table NLM taking the attribute information into consideration.

Order-planning: Sha et al. (2018) proposed a link matrix to model the order for the attribute-value tuples while generating biographies.

Struct-aware: Liu et al. (2018) proposed a structure-aware model using a modified LSTM unit and a specific attention mechanism to incorporate the attribute information.

Word & Attribute level Coverage: we also implement the implicit coverage method (Tu et al., 2016) for comparison. For word-level coverage, we replace Eq 2 with $g(s_t, h_i) = tanh(W_p h_i + W_q s_t + W_m \theta_t + b)$. For attribute-level coverage, we replace Eq 2 with $g(s_t, h_i) = tanh(W_p h_i + W_q s_t + W_m \gamma_t + b)$. θ_t and γ_t are the word-level and attribute-level coverage defined in Sec 3.1.

Models	BLEU	ROUGE
KN	2.21	0.38
Template KN	19.80	10.70
NLM	4.17	1.48
Table NLM	34.70	25.80
Order-planning	43.91	37.15
Struct-aware	44.89	41.21
Word-level Coverage*	43.44	39.84
Attri-level Coverage*	42.87	38.95
Seq2seq	43.51	39.61
+ Force-Attention	44.46	40.58
+ Richness RL †	**45.47**	**41.54**

(a) Automatic evaluation on WIKIBIO

Models	BLEU	ROUGE
Struct-aware*	40.81	36.52
Word-level Coverage*	38.85	35.11
Attri-level Coverage*	38.34	34.92
Seq2seq	39.17	35.39
+ Force Attention	41.21	36.71
+ Richness RL †	**42.03**	**37.55**

(b) Automatic evaluation on WB-filter

Table 2: BLEU and ROUGE scores on the WIKIBIO and WB-filter datasets. The baselines with * are based on our implementation while the others are reported by their authors. Models with † are trained using the RL criterion specified in Sec 3.2.2 while the remaining models are trained using the maximum likelihood estimate (MLE).

4.5 Analysis of Experimental Results

Automatic evaluations are shown in Table 2 for WIKIBIO and WB-filter. The proposed force-attention module achieves 1.11/0.98 and 2.04/1.32 BLEU/ROUGE increases on the WIKIBIO and WB-filter datasets, respectively. Although the proposed force attention method does not outperform the 'struct-aware' method in terms of BLEU and ROUGE in the WIKIBIO dataset. We show its advantages in the user-oriented scenario as well as its ability to cover the key attributes as shown in Table 4 and 5. The richness-oriented reinforced module further enhances the model performance, helping our model outperform the state-of-the-art system (Liu et al., 2018) by about 0.79 BLEU and 0.58 ROUGE. Note that the BLEU and ROUGE scores are lower in the WB-filter datasets because firstly, the WIKIBIO has much larger training set. Secondly, the gold biographies might con-

Models	Fluency	Coverage	Correctness
Seq2seq	1.87	1.99	1.95
Struct-aware	1.61	1.80	1.71
Our best	**1.54**	**1.46**	**1.61**

(a) Human evaluation on `WIKIBIO`

Models	Fluency	Coverage	Correctness
Seq2seq	2.02	1.88	1.93
Struct-aware	1.58	1.52	1.65
Our best	**1.54**	**1.39**	**1.54**

(b) Human evaluation on `WB-filter`

Table 3: Average ranking (lower is better) of 3 systems. We calculate the Pearson correlation to show the inter-annotator agreement.

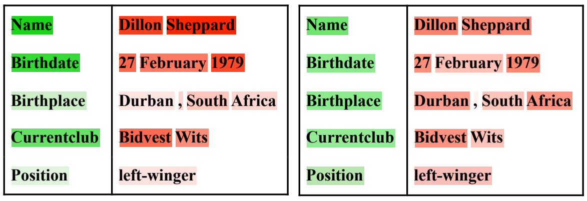
	Models	BLEU	Rich
1	seq2seq	43.51	28.21
2	+ Stepwise (only)	43.69	30.01
3	+ Global loss (only)	44.21	31.65
4	+ Stepwise + Global loss	44.46	32.90
5	+ Richness RL (only)	45.23	35.84
6	+ All	**45.47**	**37.64**

(a) Ablation studies on `WIKIBIO`

	Models	BLEU	Rich
1	seq2seq	39.17	56.30
2	+ Stepwise (only)	39.59	59.29
3	+ Global loss (only)	40.83	61.12
4	+ Stepwise + Global loss	41.21	62.81
5	+ Richness RL (only)	41.66	63.89
6	+ All	**42.03**	**64.41**

(b) Ablation studies on `WB-filter`

Table 4: The ablation studies for our model. Models 2-4 are from the force-attention method. 'Rich' is the 'information richness' defined in Eq 5.



seq2seq   seq2seq+Force-attention

seq2seq: Dillon Sheppard (born 27 february 1979) is a soccer who plays for Bidvest Wits.
seq2seq+FA: Dillon Sheppard (born 27 february 1979, Durban South Africa) is a left-winger in Bidvest Wits.

Figure 4: The average attribute-level (green) and word-level (red) coverage of the seq2seq models with or without force-attention module for an infobox in `WB-filter` (higher values are darker) in the last decoding step. The vanilla seq2seq model ignores the '*birthplace*' and '*position*' attributes as the low coverage on them while the FA module attracts enough attention on them while decoding.

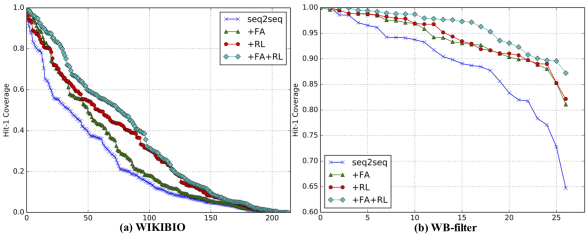


Figure 5: Hit-1 coverage (Sec 4.2) for attributes on the *test sets* of `WIKIBIO` and `WB-filter`. For better visualization, we first select the attributes whose frequencies are larger than 0.1%, then rank the Hit-1 coverage of these attributes (214 attributes in `WIKIBIO`; 26 attributes in `WB-filter`) in the descending ordering.

tain information beyond the tables. Although this phenomenon also occurs in `WIKIBIO`, the filtering of `WB-filter` magnifies this issue. Human evaluations in Table 3 show our model achieves better generation coverage and correctness than all the baselines. Table 4 shows that the ablation studies of our model.

As demonstrated in Table 5, we select an infobox from `WIKIBIO` and `WB-filter` respectively for case studies. By observing the generated description in `WIKIBIO`, we find that 1) compared with the vanilla seq2seq model, our force-attention module can cover the information in the '*Notableworks*' attribute. 2) The richness-oriented module further helps our model to cover the '*Alma mater*' and ' *Notableworks*' attributes as they are infrequent attributes (more informative) in the dataset. Additionally, due to the rareness of the word '*kiev*', our model is able to cover the related information. Similarly, the generated description for `WB-filter` covers the information from '*Organization*' and ' *Birthplace*' with the help of proposed model.

Fig 4 shows the effectiveness of the force-attention module. The decoder is guided to pay more attention to the uncovered attributes ('*birthplace*' and '*position*') while decoding. Fig 5 shows that both two proposed modules can boost the attribute-level coverage on the two datasets. Fig 6(left) explains why our model can also improve end-to-end table description generation. Attributes like '*position*', '*battles*' and '*political party*' are key information to describe the infoboxes for sportsmen, soldiers and politicians. Fig 6(right) shows the effects of λ in Eq 6.

4.6 Error Analysis

Although the proposed models achieve competitive performance, we also observe some failure cases. To sum up, the irrelevant information in the generated descriptions to the source tables. For ex-

Name:*Ivan Ohienko Metropolitan Ilarion* ; **Birthdate**:*2 January 1882* ;**Birthplace**:*Brusilov, Kiev governorate, Russian empire* ; **Deathdate**:*29 March 1972*;**Deathplace**:*Winnipeg, Canada*; **Occupation**:*cleric, historian, ethnographer, and scholar, writer, and translator*; **Language**:*Ukrainian*; **Nationality**:*Ukrainian*; **Alma mater**:*Kiev university* **Notableworks**:*translation of the bible into ukrainian* **Article title**:*Ilarion Ohienko*

Seq2seq: Ivan Ohienko Metropolitan (January 2 , 1882 – March 29 , 1972) was a Ukrainian cleric , historian , ethnographer, writer , linguist , writer and scolar.
+Force-Attention: Ivan Ohienko Metropolitan Ilarion (2 January 1882 in Brusilov – 29 march 1972 in Winnipeg) was a Ukrainian linguist , ethnographer , and scholar , best known for his translation of the bible into ukrainian .
+Richness-oriented RL: Ivan Ohienko Metropolitan Ilarion (2 January 1882 , Krusilov , Kiev governorate– 29 march 1972 , Winnipeg) was a Ukrainian cleric, historian , ethnographer , and scholar of Kiev university , best known for his translation of the bible into ukrainian .

Name:*Rajendra Singh* ; **Birthdate**:*06 August 1959* ;**Birthplace**:*Daula, Bagpat District, Uttar Pradesh* ; **Nationality**: *Indian*; **Organization**:*Tarun Bharat Sangh*; **Occupation**:*water conservationist* **Alma mater**:*Allahabad University*

Seq2seq: Rajendra Singh is an Indian water conservationist.
+Force-Attention: Rajendra Singh (born 6 August 1959) is an Indian conservationist and a senior fellow of the Tarun Bharat Sangh.
+Richness-oriented RL: Rajendra Singh (born 6 august 1959, Uttar Pradesh) is an Indian water conservationist and a member of the Tarun Bharat Sangh.

Table 5: The generated cases in WIKIBIO (above) and WB-filter (below) datasets. The underlined texts, which are the key information of the source tables, are ignored by seq2seq model.

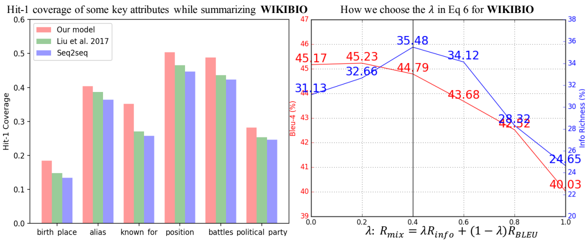


Figure 6: Hit-1 Coverage (Sec 4.2) for some key attributes (left) on the *test set* of WIKIBIO shows that our model can help to cover some key attributes while describing the tables. The right figure is the analysis of λ (Eq 6) for 'Seq2seq + RL' model on the *validation set of* WIKIBIO.

ample, a biography about a football player might contain '*in the national football league*' although the related infobox does not mention this piece of information as the similar expression exists in many instances of the training set. Although our model could largely relieve this problem as shown in human evaluation (Table 3), it is still a general problem in NLG. As for the ability to cover important information in the tables, although our model is able to cover much more comprehensive information than the previous models (Table 2 and 3). Some implicitly expressed (like if a person is retired or not) or rarely covered (like '*spouse*' or '*high school*') attributes in the source tables might still be ignored in the descriptions generated by our model. Furthermore, those pieces of information which need some form of inference across several attributes (like a time span) may not be well represented by our model.

5 Related Work

Data-to-text a language generation task to generate text for structured data. Table-to-text belongs to the data-to-text generation (Reiter and Dale, 2000). Many previous work (Barzilay and Lapata, 2005, 2006; Liang et al., 2009) treated the task as a pipelined systems, which viewed content selection and surface realization as two separate tasks. Duboue and McKeown (2002) proposed a clustering approach in the biography domain by scoring the semantic relevance of the text and paired knowledge base. In a similar vein, Barzilay and Lapata (2005) modeled the dependencies between the American football records and identified the bits of information to be verbalized. Liang et al. (2009); Angeli et al. (2010) extended the work of Barzilay and Lapata (2005) to soccer and weather domains by learning the alignment between data and text using hidden variable models. Androutsopoulos et al. (2013) and Duma and Klein (2013) focused on generating descriptive language for Ontologies and RDF triples. Most recent work utilize neural networks on data-to-text generation (Mahapatra et al., 2016; Wiseman et al., 2017; Laha et al., 2018; Kaffee et al., 2018; Freitag and Roy, 2018; Qader et al., 2018; Dou et al., 2018; Yeh et al., 2018; Jhamtani et al., 2018; Jain et al., 2018; Liu et al., 2017b, 2019; Peng et al., 2019;

Dušek et al., 2019).

Some closely relevant work also focused on the table-to-text generation. Mei et al. (2016) proposed an encoder-aligner-decoder framework for generating weather broadcast. Hachey et al. (2017) used a table-text and text-table autoencoder framework for table-to-text generation. Nema et al. (2018) proposed gated orthogonalization to avoid repetitions. Wiseman et al. (2018) used neural semi-HMM to generate template-like descriptions for structured data. Our work somewhat shares similar goals as Kiddon et al. (2016); Tu et al. (2016); Liu et al. (2017a); Gong et al. (2018) in the sense that they emphasis easily ignored (usually less frequent) features or bits of information in the training procedure by smoothing or regularization. The greatest difference between our work and theirs is that our method is tailored for covering the key information embedded in the attributes (entries) of the key-value tables rather than single words or labels. Although the deficient score of Tu et al. (2016) in Table 2 has demonstrated that word-level coverage oriented methods may not still be suitable to the structured tables, we assume other word-level constraints may easily transfer to the structured tables without losing efficiency. We leave the recognition of potential applicable word-level constraints to the future work.

This paper focused on generating one-sentence biographies for infoboxes like many previous works (Lebret et al., 2016; Hachey et al., 2017; Liu et al., 2018; Bao et al., 2018; Nema et al., 2018; Puduppully et al., 2018; Cao et al., 2018). Perez-Beltrachini and Lapata (2018) used the first paragraph of the wikipedia pages as the gold biographies aiming at generating longer biographies. We tried the same setting and unfortunately found most generated biographies contain *too much groundless* information compared with the source infoboxes. This is because the related gold biographies from first paragraph contain too much groundless information beyond the source infoboxes.

6 Conclusion and Future Work

We set up 3 goals for comprehensive description generation for attribute-value factual tables: accurate, informative and loyal. To achieve these goals, we propose force-attention method, which encourages the generator to pay more attention to previous uncovered attributes to avoid potential key attribute missing. Richness-oriented reinforcement learning is proposed to cover more informative contents in source tables, which help the generator to generate informative and accurate descriptions. The experiments on the `WIKIBIO` and `WB-filter` datasets show the merits of our model. In the future, we will explore the representation for the implicit information like whether a man is retired or not or how long a sportsman's career is given starting and ending years, in the table by including some inference strategies.

Acknowledgments

We would like to thank the anonymous reviewers for their valuable suggestions. This work is supported by the National Science Foundation of China under Grant No. 61876004, No. 61772040. The corresponding authors of this paper are Baobao Chang and Zhifang Sui.

References

Ion Androutsopoulos, Gerasimos Lampouras, and Dimitrios Galanis. 2013. Generating natural language descriptions from owl ontologies: the naturalowl system. *Journal of Artificial Intelligence Research*, 48:671–715.

Gabor Angeli, Percy Liang, and Dan Klein. 2010. A simple domain-independent probabilistic approach to generation. In *EMNLP 2010*, pages 502–512.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473.

Jun-Wei Bao, Duyu Tang, Nan Duan, Zhao Yan, Yuanhua Lv, Ming Zhou, and Tiejun Zhao. 2018. Table-to-text: Describing table region with natural language. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5020–5027.

Regina Barzilay and Mirella Lapata. 2005. Collective content selection for concept-to-text generation. In *Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 331–338. Association for Computational Linguistics.

Regina Barzilay and Mirella Lapata. 2006. Aggregation via set partitioning for natural language generation. In *NAACL*, pages 359–366. Association for Computational Linguistics.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluation the role of bleu in machine translation research. In *EACL 2006*.

Juan Cao, Junpeng Gong, and Pengzhou Zhang. 2018. Open-domain table-to-text generation based on seq2seq. In *Proceedings of the 2018 International Conference on Algorithms, Computing and Artificial Intelligence*, page 72. ACM.

Longxu Dou, Guanghui Qin, Jinpeng Wang, Jin-Ge Yao, and Chin-Yew Lin. 2018. Data2text studio: Automated text generation from structured data. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 13–18.

Pablo A Duboue and Kathleen R McKeown. 2002. Content planner construction via evolutionary algorithms and a corpus-based fitness function. In *Proceedings of INLG 2002*, pages 89–96.

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 (IWCS 2013)–Long Papers*, pages 83–94.

Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. 2019. Evaluating the state-of-the-art of end-to-end natural language generation: The e2e nlg challenge. *arXiv preprint arXiv:1901.07931*.

Markus Freitag and Scott Roy. 2018. Unsupervised natural language generation with denoising autoencoders. *arXiv preprint arXiv:1804.07899*.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 249–256.

Chengyue Gong, Xu Tan, Di He, and Tao Qin. 2018. Sentence-wise smooth regularization for sequence to sequence learning. *arXiv preprint arXiv:1812.04784*.

Ben Hachey, Will Radford, and Andrew Chisholm. 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, EACL 2017, Valencia, Spain, April 3-7, 2017, Volume 1: Long Papers*, pages 633–642.

Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H Clark, and Philipp Koehn. 2013. Scalable modified kneser-ney language model estimation. In *ACL (2)*, pages 690–696.

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

Parag Jain, Anirban Laha, Karthik Sankaranarayanan, Preksha Nema, Mitesh M Khapra, and Shreyas Shetty. 2018. A mixed hierarchical attention based encoder-decoder approach for standard table summarization. *arXiv preprint arXiv:1804.07790*.

Sébastien Jean, KyungHyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On using very large target vocabulary for 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 of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1–10.

Harsh Jhamtani, Varun Gangal, Eduard Hovy, Graham Neubig, and Taylor Berg-Kirkpatrick. 2018. Learning to generate move-by-move commentary for chess games from large-scale social forum data. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1661–1671.

Lucie-Aimée Kaffee, Hady ElSahar, Pavlos Vougiouklis, Christophe Gravier, Frédérique Laforest, Jonathon S. Hare, and Elena Simperl. 2018. Learning to generate wikipedia summaries for underserved languages from wikidata. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 2 (Short Papers)*, pages 640–645.

Chloé Kiddon, Luke Zettlemoyer, and Yejin Choi. 2016. Globally coherent text generation with neural checklist models. In *EMNLP 2016*, pages 329–339.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Anirban Laha, Parag Jain, Abhijit Mishra, and Karthik Sankaranarayanan. 2018. Scalable micro-planned generation of discourse from structured data. *CoRR*, abs/1810.02889.

Rémi Lebret, David Grangier, and Michael Auli. 2016. Neural text generation from structured data with application to the biography domain. In *EMNLP 2016*, pages 1203–1213.

Liunian Li and Xiaojun Wan. 2018. Point precisely: Towards ensuring the precision of data in generated texts using delayed copy mechanism. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1044–1055.

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: Volume 1-Volume 1*, pages 91–99. Association for Computational Linguistics.

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

Tianyu Liu, Fuli Luo, Qiaolin Xia, Shuming Ma, Baobao Chang, and Zhifang Sui. 2019. Hierarchical encoder with auxiliary supervision for neural table-to-text generation: Learning better representation for tables. In *Proceedings of AAAI*.

Tianyu Liu, Kexiang Wang, Baobao Chang, and Zhifang Sui. 2017a. A soft-label method for noise-tolerant distantly supervised relation extraction. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1790–1795.

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 Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4881–4888.

Tianyu Liu, Bingzhen Wei, Baobao Chang, and Zhifang Sui. 2017b. Large-scale simple question generation by template-based seq2seq learning. In *Natural Language Processing and Chinese Computing - 6th CCF International Conference, NLPCC 2017, Dalian, China, November 8-12, 2017, Proceedings*, pages 75–87.

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*, pages 143–152.

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 *NAACL HLT 2016*, pages 720–730.

George A Miller. 1995. Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

Preksha Nema, Shreyas Shetty, Parag Jain, Anirban Laha, Karthik Sankaranarayanan, and Mitesh M Khapra. 2018. Generating descriptions from structured data using a bifocal attention mechanism and gated orthogonalization. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1539–1550.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL2002*, pages 311–318.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304*.

Hao Peng, Ankur P. Parikh, Manaal Faruqui, Bhuwan Dhingra, and Das Dipanjan. 2019. Text generation with exemplar-based adaptive decoding. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

Laura Perez-Beltrachini and Mirella Lapata. 2018. Bootstrapping generators from noisy data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 1516–1527.

Ratish Puduppully, Li Dong, and Mirella Lapata. 2018. Data-to-text generation with content selection and planning. *arXiv preprint arXiv:1809.00582*.

Raheel Qader, Khoder Jneid, François Portet, and Cyril Labbé. 2018. Generation of company descriptions using concept-to-text and text-to-text deep models: dataset collection and systems evaluation. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 254–263.

Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. 2015. Sequence level training with recurrent neural networks. *CoRR*, abs/1511.06732.

Ehud Reiter. 2018. A structured review of the validity of bleu. *Computational Linguistics*, 44(3):393–401.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.

Ehud Reiter and Robert Dale. 2000. *Building natural language generation systems*. Cambridge university press.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *CVPR*, volume 1, page 3.

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 Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5414–5421.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *ACL 2016, Volume 1: Long Papers*.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. In *Reinforcement Learning*, pages 5–32. Springer.

Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 2253–2263.

Sam Wiseman, Stuart M. Shieber, and Alexander M. Rush. 2018. Learning neural templates for text generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 3174–3187.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International Conference on Machine Learning*, pages 2048–2057.

Shyh-Horng Yeh, Hen-Hsen Huang, and Hsin-Hsi Chen. 2018. Precise description generation for knowledge base entities with local pointer network. In *2018 IEEE/WIC/ACM International Conference on Web Intelligence (WI)*, pages 214–221. IEEE.