

# Towards Faithfulness in Open Domain Table-to-text Generation from an Entity-centric View

Tianyu Liu<sup>1\*</sup>, Xin Zheng<sup>2,3\*</sup>, Baobao Chang<sup>1,4</sup>, Zhifang Sui<sup>1,4</sup>

<sup>1</sup> Ministry of Education (MOE) Key Laboratory of Computational Linguistics, School of EECS, Peking University

<sup>2</sup> Chinese Information Processing Laboratory, Institute of Software, Chinese Academy of Sciences, Beijing, China

<sup>3</sup> University of Chinese Academy of Sciences, Beijing, China

<sup>4</sup> Pengcheng Laboratory, Shenzhen, China

tianyu0421@pku.edu.cn, zheng\_xin@bupt.edu.cn, chbb@pku.edu.cn, szf@pku.edu.cn

## Abstract

In open domain table-to-text generation, we notice that the unfaithful generation usually contains hallucinated content which can not be aligned to any input table record. We thus try to evaluate the generation faithfulness with two entity-centric metrics: table record coverage and the ratio of hallucinated entities in text, both of which are shown to have strong agreement with human judgements. Then based on these metrics, we quantitatively analyze the correlation between training data quality and generation fidelity which indicates the potential usage of entity information in faithful generation. Motivated by these findings, we propose two methods for faithful generation: 1) augmented training by incorporating the auxiliary entity information, including both an augmented plan-based model and an unsupervised model and 2) training instance selection based on faithfulness ranking. We show these approaches improve generation fidelity in both full dataset setting and few shot learning settings by both automatic and human evaluations.

## 1 Introduction

The difficulty of faithful table-to-text generation originates from the divergence of source tables and reference text (Perez-Beltrachini and Lapata 2018) in the training stage, especially for open domain datasets without human curation, e.g. WIKIPERSON (Wang et al. 2018). In WIKIPERSON, we observe that the unfaithful generation, as well as some reference text in the training data, often contains hallucinated entities which can not be aligned with any input table record. This motivates us to evaluate the faithfulness of a text to a given table by two entity-centric metrics: the statistics of table record coverage and hallucinated entities in both training and evaluation stages. The metrics are computed based on the name entities recognition (NER) in text with off-the-shelf tools, and their alignment with table records. We find the proposed metrics have high correlation with human judgements (Sec 2.1).

Then we quantitatively study how training data quality affects the generation fidelity (Sec 2.2) based on the proposed metrics. Specifically we show that filtering uncovered

records in source table would increase generation coverage<sup>1</sup>, and truncating the sentences prone to hallucination in reference text would accordingly reduce generation hallucination. However, we also observe a trade-off between generation faithfulness and coverage, e.g. truncating reference text would possibly lower the generation coverage to source table. We thus seek for an alternative method to reduce the hallucinations in the training stage while maintaining high coverage.

To this end, we firstly propose plan-based generation with pseudo parallel training corpus, which we call the augmented plan-based method (Sec 3.1). Some prior work (Moryossef, Goldberg, and Dagan 2019; Ferreira et al. 2019) have shown that the two-stage plan-based generation could lead to a more accuracy and controllable generation in data-to-text generation. However, most previous work focus on high quality datasets without much content mismatch between the input table and the reference text (e.g. WEBNLG and E2E (Dušek, Novikova, and Rieser 2019)). However, in the open domain setting, we have to face noisy training data<sup>2</sup>, i.e. hallucinations in reference text. Thus we exploit serialized plans which are formulated as text sequences, in this way, we can easily use any established sequence-to-sequence model as neural planner rather than a dedicated planning module (Moryossef, Goldberg, and Dagan 2019; Moryossef, Dagan, and Goldberg 2019). Our plans model the number of sentences to be generated and the order of given table records in each sentence. In the plan-to-text phase, we create a pseudo parallel training corpus which incorporates the auxiliary uncovered entities in the reference text into an augmented plan, and then the model generates text according to the augmented plan. The augmented plans effectively reduce the hallucinations in the pseudo training corpus and correspondingly encourage the generator to stick to the designated plans. The experiments show that the proposed augmented plan based methods not only reduce about

<sup>1</sup>Measuring generation coverage can be useful in some specific scenarios. The generator would be expected to cover all records (e.g. WEBNLG (Colin et al. 2016)) or only summarize salient records (e.g. WIKIBIO (Lebret, Grangier, and Auli 2016)).

<sup>2</sup>The main reason for data noise is the lack of human intervention in data collection. It could be of high cost to monitor the faithfulness of multi-sentence reference in the open domain.

\*The first two authors contribute equally to this work.  
Copyright © 2021, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

60% hallucinations compared with the counterparts trained in the end-to-end fashion in full dataset setting but also greatly enhance the model performance in few shot learning. We also achieve the state-of-the-art performance on WIKIPERSON dataset and show that the proposed methods work well on both pre-trained models and (non pre-trained) Transformer models.

We also propose a training instance selection method based on proposed hallucination evaluation metrics and related faithfulness ranking (Sec 3.2). We show that with only 5% top ranking training instances, pretrained model BART (Lewis et al. 2019) outperforms its variant trained on full dataset by about 7 PARENT (Dhingra et al. 2019) points. This also implies the effectiveness of the proposed entity-based metrics for faithfulness evaluation.

## 2 Faithfulness in Table-to-text

In this section, we introduce two evaluation metrics called table record coverage  $P_{cover}$  and the ratio of hallucination  $R_{hallu}$  based on the alignment between source table records and recognized entities in the text. Through human annotations, we verified that the proposed metrics have high correlation with human judgement. Finally, we practice quantitative analyses to show how the proposed two metrics correlate with generation fidelity and gain insights on how to increase table coverage or reduce hallucinations in open domain table-to-text generation.

### 2.1 Entity-centric Evaluation

A structured table  $T$  can be expressed by a set of records  $\{r_k\}_{k=1}^K$ , where each record is an (*attribute, value*) pair. There is also a reference description  $R$  available for each table. The task aims at generate a text  $G$  which describes the records in  $T$  fluently and faithfully. Due to the lack of human intervention, the reference  $R$  may contain hallucinations and have low table records coverage in the open domain datasets.

**Entity recognition and alignment** With off-the-shelf NER tools<sup>3</sup>, we could recognize entities like *person, location, date, event* in text, which characterize the expressed facts. Suppose we have recognized a group of entities  $\{e_i\}_{i=1}^M$  from a specific text  $G$  ( $G$  can be training reference  $R$ , which aims at evaluating training data quality or predicted table description  $D$ , which aims at assessing model performance on test sets.). We then heuristically decide whether a table record  $r_k$  is **covered** by  $G$ .

Our heuristic matching rules include: 1) Exact Match: a text span in  $G$  can exactly match  $r_k$ 's *value*. 2) Filtered Sub-sequence Match: For non-numeric entity  $e_i$  with *PERSON, NORP, FAC, ORG, GPE, LOC, PRODUCT, EVENT or WORK\_OF\_ART* labels, after filtering the stop words in  $e_i$  and  $r_k$ 's *value*,  $r_k$  matches  $e_i$  if and only if  $e_i^{filter}$  is a sub-sequence of  $r_k^{filter}$ .

In our benchmark dataset WIKIPERSON, 99.3% of table records in the training set could be aligned with corresponding reference text with the above mentioned heuristic match-

<sup>3</sup>We use Stanza (Qi et al. 2020) toolkit throughout our experiments. <https://stanfordnlp.github.io/stanza/>

Model	BLEU PARENT	$P_{cover}(\uparrow)$	$R_{hallu}(\downarrow)$	LEN
PG-Net	23.56	50.14	88.63%	<b>0.091</b> 59.1
Transformer	24.63	51.86	89.74%	0.093 63.4
BART	<b>31.16</b>	<b>52.54</b>	<b>98.31%</b>	0.099 81.6

Table 1: Model performances on the WIKIPERSON dataset in terms of BLEU-4 (Papineni et al. 2002), PARENT and the entity-centric statistics (Sec 2.1).

ing strategies. As suggested by training set, the test-time generation could cover most table records without too much pain when using a powerful generator, however reducing the hallucinations in the generated text is non-trivial.

**Entity-centric metrics** After recognizing  $\{e_i\}_{i=1}^M$  in text  $G$ , we use the above mentioned heuristic strategies to establish an alignment between  $\{e_i\}_{i=1}^M$  and table records  $\{r_k\}_{k=1}^K$ . We consider the unaligned entities in  $\{e_i\}_{i=1}^M$  as hallucinated entities in  $G$ . For a pair of table and text  $(T^j, G^j)$ , we define  $n_{hallu}^j, p_{cover}^j, l^j$  as the number of hallucinated entities in  $G^j$ , the ratio of covered records in  $T^j$  and the length of  $G^j$  respectively. For a set of  $N$  (table, text) pairs  $\{(T^j, G^j)\}_{j=1}^N$ , the corpus level table coverage  $P_{cover}$  and the hallucinated ratio  $R_{hallu}$  are shown as follows:

$$P_{cover} = \sum_j p_{cover}^j / N; R_{hallu} = \sum_j n_{hallu}^j / (N * L)$$

in which  $L = \sum_j l^j / N$  is the average length of text  $\{G^j\}_{j=1}^N$ .

**Correlation with human judgement** To assess the correlation of  $P_{cover}$  and  $R_{hallu}$  with human judgement, we sample 400 (*table, text*) pairs, of which 100 pairs from training set, 300 pairs from the 3 model outputs (in Table 1) while testing (100 pairs each). We hire three well-educated annotators to decide 1) for each table record whether it is covered by corresponding text and 2) for each recognized hallucinated entity in text whether it is really a hallucination or not. The Fleiss' kappa between annotators is 0.71, which indicates a 'substantial agreement' (Landis and Koch 1977). We then use majority vote to decide the final annotations. Our coverage and hallucination decisions achieve 95.2% and 93.7% accuracy respectively according to human annotations.

**Limitations** Note that we have shown that proposed metrics have high *precision* in identifying hallucinations. However it is not enough to merely consider recognized entities in text, in other words, the proposed  $R_{hallu}$  may not have a high *recall* on recognizing other forms of hallucinations (e.g. verbal hallucination). We thus ask the human judges to mark any hallucinated word in the generated text while comparing different systems.

### 2.2 Quantitative Analysis

Firstly we benchmark Pointer-Generator (PG-Net) (See, Liu, and Manning 2017), Transformer (Vaswani et al. 2017) and BART models on WIKIPERSON dataset in Table 1. We then analyze how  $P_{cover}$  and  $R_{hallu}$  (Sec 2.1) in training data affect the generation fidelity. Note that we do not change the test and dev sets in all following experiments.

	$\lambda = 0$	$\lambda = 0.25$	$\lambda = 0.5$	$\lambda = 0.75$	$\lambda = 1.0$
Table Record Coverage (%) $P_{cover}(\uparrow)$					
Train	67.7	72.5	<b>78.7</b>	<b>87.3</b>	<b>100.0</b>
PG-Net	69.6	71.4	73.6	77.4	85.7
Transformer	71.2	72.8	75.2	79.0	86.3
BART	<b>75.4</b>	<b>77.2</b>	78.1	82.6	91.9
Ratio of hallucinated entities $R_{hallu}(\downarrow)$					
Train	0.060	0.060	0.060	0.060	0.060
PG-Net	0.049	0.051	0.052	0.054	0.057
Transformer	0.045	0.046	<b>0.047</b>	<b>0.049</b>	<b>0.054</b>
BART	<b>0.043</b>	<b>0.045</b>	0.048	0.053	0.063
Avg. length of (tokenized) text					
Train	27.7	27.7	27.7	27.7	27.7
PG-Net	25.3	25.8	26.0	27.2	28.6
Transformer	25.7	26.1	26.3	27.5	29.2
BART	25.5	26.0	26.6	28.2	34.7

(a) Filtering uncovered table records

	No Trunc	$N_{keep}=3$	$N_{keep}=2$	$N_{keep}=1$
Table Record Coverage (%) $P_{cover}(\uparrow)$				
Train	<b>99.3</b>	91.4	82.8	67.7
PG-Net	88.6	85.7	78.4	69.6
Transformer	89.7	86.9	81.4	71.2
BART	98.3	<b>96.9</b>	<b>89.6</b>	<b>75.4</b>
Ratio of hallucinated entities $R_{hallu}(\downarrow)$				
Train	0.096	0.086	0.077	0.060
PG-Net	<b>0.091</b>	<b>0.078</b>	<b>0.061</b>	0.049
Transformer	0.093	0.080	0.064	0.045
BART	0.099	0.083	0.067	<b>0.043</b>
Avg. length of (tokenized) text				
Train	88.3	62.7	46.1	27.7
PG-Net	59.1	44.7	32.3	25.3
Transformer	63.4	48.2	36.2	25.7
BART	81.6	58.6	40.9	25.5

(b) Truncating reference text

Table 2: The statistics of training data and model outputs on test set in two different settings (Sec 2.2).

**Filtering uncovered table records** To better investigate how  $P_{cover}$  in training set affects generation coverage, we only keep *the first sentence*<sup>4</sup> in reference text and drop other sentences while training. Then we filter the uncovered table records by setting a threshold  $\lambda$ , e.g. when  $\lambda=0.75$ , we randomly filter 75% of uncovered records in training set.

**Truncating reference text** On average, each reference text in WIKIPERSON training set contains 4.33 sentences. We observe that the sentences near the end are prone to hallucination, so we try to reduce  $R_{hallu}$  in the training set by truncating the reference text. We let  $N_{keep}$  to denote the max number of sentences we keep in each reference text and then drop all the following sentences.

**Insights from quantitative analysis** In Table 2a, as  $P_{cover}$  of training set increases ( $\lambda$  increases), all three models show higher coverage for the input table records and generate longer sentences. Notably, although  $R_{hallu}$  of training set remains the same for all  $\lambda$ , we see a small growth on

<sup>4</sup>As the original training dataset has 99.3% table record coverage (Table 2a), ‘one sentence’ setting allows a wider range (67.7%-100% in Table 2a) in controlling  $P_{cover}$  by  $\lambda$ .

$R_{hallu}$  of model outputs while increasing  $\lambda$ <sup>5</sup>.

In Table 2b, when decreasing  $N_{keep}$  (truncating more sentences), the hallucination ratio  $R_{hallu}$  in model outputs are reduced as the same metric drops in the training set. However, while reducing hallucination,  $P_{cover}$  in training set and model outputs declines at the same time, which reminds us the potential trade-off in truncation.

We have applied dataset modification, i.e. filtering and truncation, on the training set and seen there might be a trade-off between reducing hallucination (decreasing  $R_{hallu}$ ) and increasing table coverage (increasing  $P_{cover}$ ). For faithful description generation, we aim to find a way to reduce  $R_{hallu}$  in the training phase while keeping table coverage (almost) unchanged. We come up with pseudo training data augmentation in plan-based generation and instance selection based on faithfulness ranking according to the quantitative analysis in this section.

### 3 Entity-centric Faithful Generation

In this section, based on the alignment between source tables and reference text, we propose two methods for faithful generation: augmented plan based generation and training instance selection according to faithfulness ranking.

#### 3.1 Augmented Plan-based Generation

**Two-step Plan-based Generation** Moryossef, Goldberg, and Dagan (2019) have shown a *two-step generator* with a separate text planner could improve generation faithfulness on WebNLG, a human curated dataset where the reference text barely contains any hallucination. The number of input table records in WebNLG is always smaller than 8. They view the input records as directed connected graph and treat depth first search (DFS) trajectory as the intermediate plan. However, in real world open domain setting, the input tables might contain diverse distinct table records which makes DFS trajectory plans very time-consuming to train. Moreover, the reference text in the training data might contain hallucinations which can not be aligned with any record in the table, which greatly hinder the generation fidelity.

To this end, we firstly extend the usage of plan-based method to the open domain setting using serialized plans. We decompose the generation into two stages: table-to-plan and plan-to-text by introducing a trainable text planner. As exemplified in Fig 1, for the text planner, we propose a neural plan generator (‘R’ $\rightarrow$ ‘R+P’) which transforms the input table records to the serialized plans. The plans contain sentence planning (‘SEP’ token) and order planning of table records in each sentence. Specifically we constrain the plan generator to output only the attribute names that appear in the input table besides the ‘SEP’ token and then add the paired attribute value for each generated attribute name by post editing<sup>6</sup>. For the neural planner, we can easily use

<sup>5</sup>We assume this is due to the information loss when filtering the records, as a poorly-informed decoder may behave like an open-ended language model (Maynez et al. 2020).

<sup>6</sup>In testing, we also post-process the learned plans to avoid empty sentence plans (i.e. two consecutive ‘SEP’ tokens) and repetitive table records (i.e. limiting each table record to appear only



Figure 1: For the left figure, we show different forms of model inputs (Table 4) for the text generator in the two-phase generation. We mark the covered table records, the recognized hallucinated entities and the sentence delimiter in red, blue and green respectively. For the right figure, we show the training and inference procedures in the augmented plan-based generation (ID=8 in Table 4).

state-of-the-art sequence-to-sequence models thanks to textual serialized plans.

**Pseudo Parallel Training Corpus** For text realization, we acquire descriptions by feeding the plans to a plan-to-text generator. In this process, we introduce augmented plans and pseudo parallel training corpus to reduce hallucinations in the generation. We incorporate the auxiliary entity information extracted from reference text into the augmented plans (‘Aug-plan’, ‘R+E+P’ in Fig 1) and feed them to the plan-to-text generator in the training phase. In this way, the augmented plans and related text form a pseudo parallel training corpus which does not contain any hallucinated entity ( $R_{hallu} = 0$  in this scenario). According to the experimental results, the pseudo parallel corpus greatly reduces the hallucinations in the generated descriptions, which is consistent with our findings in Sec 2.2.

Please refer to Table 4 for the different input forms of the plan-to-text generator in training and evaluation phases.

**Unsupervised generation** As a byproduct of entity recognition, we also propose an unsupervised model which does not need parallel table-to-text corpus, instead it uses text-only resources and the corresponding recognized entities in text. Concretely in training we only feed a sequence of entity mentions (‘E’ without NER tags in Fig 1) extracted from text to the generator. While testing we use the values of table records (‘R’ without attribute names in Fig 1) as input.

### 3.2 Instance Selection for Efficient Training

We have shown in the last sections that the divergence between source tables and reference text is one of the main reasons that hallucination exist in the generated text. We thus wonder if given high quality training sets, how much performance gain we could expect from the end-to-end trained models. We focus on reducing hallucination, which is shown to be the bottleneck for faithful generation. We rank the training instances with instance-level hallucination ratio  $r_{hallu} = n_{hallu}/l$  (lower is better)<sup>7</sup>. Then we view the top-1%/5%/10% ranking data as ‘selected’ training sets (‘S’), and compare them to randomly sampled training sets (‘R’) in Table 3. ‘S-n%’ can be viewed as a training set which has higher quality than the corresponding ‘R-n%’. When n grows in S-n%, we see the increase of table coverage  $P_{cover}$ , generation length and also a slight increase on  $R_{hallu}$ , which consistently outperform their counterparts R-n%. S-5% even outperforms full-dataset baseline (ID=1 in Table 4) by 7 PARENT points.

Setting	BLEU	PARENT	$P_{cover}$	$R_{hallu}(\downarrow)$	LEN
R-1%	<b>11.15</b>	<b>51.87</b>	88.67%	0.087	50.1
S-1%	7.40	49.66	<b>90.27%</b>	<b>0.015</b>	32.1
R-5%	<b>26.76</b>	52.46	98.44%	0.094	61.7
S-5%	11.80	<b>59.25</b>	<b>98.86%</b>	<b>0.018</b>	46.7
R-10%	<b>26.84</b>	52.73	98.55%	0.096	62.2
S-10%	12.52	<b>59.63</b>	<b>99.17%</b>	<b>0.020</b>	56.7

Table 3: Model performances of different downsampling methods (Sec 3.2). ‘R’, ‘S’ means randomly or ranking-based instance selection methods respectively.

When n grows in S-n%, we see the increase of table coverage  $P_{cover}$ , generation length and also a slight increase on  $R_{hallu}$ , which consistently outperform their counterparts R-n%. S-5% even outperforms full-dataset baseline (ID=1 in Table 4) by 7 PARENT points.

<sup>7</sup> $n_{hallu}$  and  $l$  are the number of hallucinated entities and reference length as introduced in Sec 2.1.

ID	Setting	Comp.	Train Input	Test&Dev Input	BLEU	PARENT	$P_{cover}(\%,\uparrow)$	$R_{hallu}(\downarrow)$	LEN
1	Baseline (E2E)	–	R	R	31.16	52.54	98.31	0.099	81.6
2	Unsupervised	Yes	E	R	<b>17.28</b>	<b>43.04</b>	<b>97.91</b>	<b>0.029</b>	50.0
3	Unsupervised	No	E	E	<u>48.27</u>	<u>63.94</u>	–	–	82.4
4	Plan	Yes	R+P <sub>gold</sub>	R	<b>18.23</b>	51.53	99.48	0.069	46.1
5	Plan	Yes	R+P <sub>gold</sub>	R+P <sub>learn</sub>	7.67	<b>54.81</b>	<b>99.66</b>	<b>0.053</b>	53.2
6	Plan	No	R+P <sub>gold</sub>	R+P <sub>gold</sub>	<u>34.86</u>	<u>54.25</u>	<u>97.76</u>	<u>0.097</u>	81.6
7	Aug-plan	Yes	R+E+P <sub>gold</sub>	R	12.10	50.70	98.68	0.015	35.8
8	Aug-plan	Yes	R+E+P <sub>gold</sub>	R+P <sub>learn</sub>	<b>17.12</b>	<b>56.75</b>	<b>99.73</b>	<b>0.006</b>	39.7
9	Aug-plan	No	R+E+P <sub>gold</sub>	R+P <sub>gold</sub>	29.12	57.84	97.98	0.029	60.6
10	Aug-plan	No	R+E+P <sub>gold</sub>	R+E+P <sub>gold</sub>	<u>53.33</u>	<u>66.12</u>	–	–	81.6

Human Evaluation (Sec 4.2)- Ratio of hallucinated words:

End-to-end [ID=1]: 20.9%; Unsupervised [ID=2]: 12.0%; Plan-based [ID=5]: 15.3%; Augplan-based [ID=8]: 8.1%

Table 4: The performances of BART-based unsupervised and plan-based models on WIKIPERSON. The baseline is an end-to-end (E2E) trained BART model in Table 1. ‘R’, ‘E’, ‘P<sub>gold</sub>’ and ‘P<sub>learn</sub>’ represent table records, recognized entities, gold plans extracted from reference text and learned plans respectively as shown in Fig 1. ‘Comp.’ indicates the comparability to baseline as gold plan P<sub>gold</sub> or entities E are actually not accessible while testing. In each setting, we underline the best scores, and show the best scores comparable to baseline in bold.

Model	BLEU	PARENT
RNN Structure		
RNN (Bahdanau, Cho, and Bengio 2014)	22.24	43.41
Structure-aware (Liu et al. 2018)	22.76	46.47
PG-Net (See, Liu, and Manning 2017)	<b>23.56</b>	50.14
KB Description (Wang et al. 2018)	16.20	<b>51.03</b>
Transformer Structure		
Transformer (Wang et al. 2020b)	24.57	51.87
+Content Matching (Wang et al. 2020b)	24.56	53.06
Transformer (Our Implementation)	<b>24.63</b>	51.86
w/ Plan	7.15	52.92
w/ Augmented Plan	14.56	<b>54.78</b>
BART (Our Implementation)	<u>31.16</u>	52.54
w/ Plan	7.67	54.81
w/ Augmented Plan	17.12	<b>56.75</b>

Table 5: The automatic evaluation of augmented plan-based method and other baselines on WIKIPERSON dataset. Note that in this dataset, the reference text may be noisy due to the hallucinated content, so BLEU scores can not measure the generation fidelity. The proposed augmented plan-based generation enhances faithfulness in both Transformer and BART structures and achieve the best PARENT scores.

## 4 Experiments

### 4.1 Datasets

The WIKIPERSON dataset<sup>8</sup> (Wang et al. 2018) contains 250186, 30487, and 29982 (*table*, *text*) pairs in training, dev and test sets respectively, which is exactly the same setting as Wang et al. (2018) and Wang et al. (2020b). The average sentence number in this dataset is 4.33 (88.3 tokens).

### 4.2 Automatic and Human Evaluation

Apart from the proposed metrics  $P_{cover}$  and  $R_{hallu}$ , we also report the model performances on BLEU and PARENT. PARENT is hybrid measurement which not only encourages

<sup>8</sup>[https://github.com/EagleW/Describing\\_a\\_Knowledge\\_Base](https://github.com/EagleW/Describing_a_Knowledge_Base)

Model	100	250	500	2500	All
PARENT ( $\uparrow$ )					
BART	25.48	41.62	47.34	51.72	52.54
w/ Plan	37.23	47.57	50.92	52.67	54.81
w/ Aug-Plan	<b>40.46</b>	<b>50.32</b>	<b>52.38</b>	<b>54.13</b>	<b>56.75</b>
$P_{cover}(\%)(\uparrow)$					
BART	77.46	80.32	82.48	89.02	98.31
w/ Plan	<b>79.21</b>	83.53	<b>88.59</b>	92.19	99.66
w/ Aug-Plan	78.93	<b>84.05</b>	88.39	<b>92.74</b>	<b>99.73</b>
$R_{hallu}(\downarrow)$					
BART	0.077	0.080	0.085	0.089	0.099
w/ Plan	0.043	0.045	0.048	0.048	0.053
w/ Aug-Plan	<b>0.008</b>	<b>0.007</b>	<b>0.005</b>	<b>0.005</b>	<b>0.006</b>

Table 6: Model performance of plan-based models in the few shot learning setting. 100/250/500/2500 represent the size of training instances.

n-gram overlap with reference text but also rewards the generation with high coverage on the source table, thus it can be used to measure the generation fidelity.

For human evaluation, as we have mentioned in Sec 2.2, the proposed two metrics have high precision in identifying hallucinations, however it may not find all hallucinations in the generation. For four models in Table 4: baseline (ID=1), unsupervised (ID=2), plan-based (ID=5) and augplan-based (ID=8) models, we conduct human evaluation<sup>9</sup> by randomly sampling 100 generations from each model and hiring three annotators to independently annotate *any* hallucinated word in generation. The Fleiss’ kappa of human judgement is 0.52. Through the human judgement, we can obtain the ratio of hallucinated words. We then calculate the overall hallucinated word ratio by averaging the annotators’ decisions.

<sup>9</sup>Here we omit human evaluations on the fluency and table record coverage as we have seen all the outputs have high coverage (Table 4) and good fluency (thanks to BART).

Attribute	Value	Attribute	Value
Name_ID	①Tom J. Murray	End time	⑧1914
Country of citizenship	②United States	Member of political party	⑨Democratic Party (United States)
Occupation	③District attorney	Date of birth	⑩August 1 1894
Position held	④Delegate	Date of death	⑪November 28 1971
Instance of	⑤Democratic National Convention	Conflict	⑫World War I
Place of birth	⑥Jackson, Tennessee	Military branch	⑬United States Army
Educated at	⑦Union University		

Model	Miss	Generated Text
BART	③	①Tom J. Murray ( ⑩August 1 1894 – ⑪November 28 1971 ) was an ②United States politician and a ⑨Democratic Party (United States) member of the U.S. House of Representatives from Tennessee . during ⑫World War I he served in the ⑬United States Army <b>in France</b> . he graduated from ⑦Union University ⑧in 1914 . murray was born in ⑥Jackson, Tennessee . he served as a ④Delegate to the ⑤Democratic National Convention <b>in 1968 and 1969</b> . he died in Jackson on November 28 1971 and <b>was interred in Jackson 's Oakwood Cemetery</b> .
w/ Plan		①Tom J. Murray ( ⑩August 1 1894 – ⑪November 28 1971 ) was an ②United States ⑨Democratic Party (United States) politician ③District attorney and ④Delegate from the U.S. state of Tennessee . during ⑫World War I he served in the ⑬United States Army . murray was born in ⑥ Jackson, Tennessee . murray was a delegate to the ⑤Democratic National Convention <b>in 1968</b> . he attended the public schools and graduated from ⑦Union University <b>in Union, Tennessee</b> ⑧in 1914 .
w/ Aug-plan		①Tom J. Murray ( ⑩August 1 1894 – ⑪November 28 1971 ) was an ②United States ⑨Democratic Party (United States) politician who served as a ③District attorney and ④Delegate . during ⑫World War I he served in the ⑬United States Army . murray was born in ⑥Jackson, Tennessee . murray was a delegate to the ⑤Democratic National Convention . he graduated from the law department of ⑦Union University ⑧in 1914 .

Figure 2: An example from the test set of WIKIPERSON. We mark the hallucinated content in red. Note that the marked content is *factually* wrong, e.g. in end of ‘BART’ generated text, Murray was actually interred in ‘the city’s Hollywood Cemetery’ rather than ‘Jackson’s Oakwood Cemetery’, which shows that those hallucinated content actually comes from the misguidance of hallucinations in the training stage, rather than the implicit knowledge acquisition of large-scale pre-trained models.

### 4.3 Experimental Settings

In this paper we implement Pointer-Generator (PG-Net) (See, Liu, and Manning 2017), Transformer (Vaswani et al. 2017) and BART (Lewis et al. 2019) models on WIKIPERSON dataset. Our implementation for Transformer and BART is based on fairseq (Ott et al. 2019)<sup>10</sup>. For PG-Net, our implementation is based on a pytorch version re-implementation on github<sup>11</sup>. For Transformer model, we use the same parameter setting as Wang et al. (2020b) (with copy mechanism). The hidden units of the multi-head attention and the feed-forward layer are set to 2048. The embedding size is set to 512. The number of heads is set to 8, and the number of Transformer blocks is 3. Beam size is set to be 5. For other parameters except learning rate, we use the default setting in fairseq according to Vaswani et al. (2017). For BART, we use the pretrained BART (large) architecture in fairseq which has 12 layers. For PG-Net, we set the embedding size, hidden size and beam size as 256, 512 and 5 respectively. For all three models, we use adam optimizer (Kingma and Ba 2014) ( $\beta_1 = 0.9, \beta_2 = 0.999$ ). The learning rates are set as  $1e-5, 1e-5, 3e-5$  for PG-Net, Transformer and BART models respectively. The vocabulary size is limited to 50k for every model.

<sup>10</sup><https://github.com/pytorch/fairseq>

<sup>11</sup>[https://github.com/atulkum/pointer\\_summarizer](https://github.com/atulkum/pointer_summarizer)

## 5 Analyses and Discussions

### 5.1 Augmented Plan-based Method

**Comparison with other baselines** We compare the proposed augmented plan-based model with the other baselines in Table 5. The plan-based method outperform the state-of-the-art baseline (Wang et al. 2020b). Note that our implemented transformer has similar model capacity with Wang et al. (2020b), i.e. the same hidden dimension setting. The experiments with BART model show that the proposed method also works well with large-scale pre-trained models.

**A closer look at plan-based models** We list the model input in the plan-to-text stage (Sec 3.1) of plan-based generation while training or testing the model in Table 4. We summarize our findings from the automatic and human evaluations in Table 4 as follows:

1) Both augmented plan-based and unsupervised models greatly reduce generation hallucination as we create pseudo parallel corpus by explicitly adding hallucinated entities in reference text while training. The new model inputs perfectly match the references as all the entities in reference text appear in the new inputs ( $R_{hallu} = 0$  in training set). This phenomenon comports with our findings in Table 2 (b): decreasing  $R_{hallu}$  in training set correspondingly reduce hallucination in the evaluation phase.

2) For testing with gold plans (ID=6,9,10), we use the gold plans extracted from test reference text *without* the post-

Attribute	Value	Attribute	Value
Name ID	①Thorsten Teichert	member of political party	⑤Nasjonal Samling
Date of birth	②1 February 1917	Sibling	⑥John Thronsen
date of death	③15 June 2003	Member of	⑦Hirden
country of citizenship	④Norway		

  

Plans	Generated Text
①②③④⑤SEP⑥SEP⑦	①Thorvald Thronsen ( ②1 February 1917 – ③15 June 2003 ) was a ④Norway politician for ⑤ Nasjonal Samling . he was the brother of ⑥John Thronsen . he was also a member of ⑦Hirden .
①②③④⑤SEP⑦SEP⑥	①Thorvald Thronsen ( ②1 February 1917 – ③15 June 2003 ) was a ④Norway politician for ⑤ Nasjonal Samling . he was a member of ⑦Hirden . he was the brother of ⑥John Thronsen .
①②③④⑤⑦SEP⑥	①Thorvald Thronsen ( ②1 February 1917 – ③15 June 2003 ) was a ④Norway politician for the ⑤ Nasjonal Samling and ⑦Hirden parties . he was the brother of ⑥John Thronsen .

Figure 3: A demonstration of different model outputs given different plans. Here we show the generated text from augmented plan-based models on the test set. The generated text perfectly follows the given plans.

processing mentioned in Footnote 6<sup>12</sup>, in order to get better BLEU scores, however at the same time sacrificing fidelity ( $R_{hallu}$  in ID=5,6 and ID=8,9). This suggests BLEU may not be a good metric for faithfulness evaluation in the loosely aligned table-to-text datasets<sup>13</sup>.

3) In ID = 4 or 7, plan-based models would generate one-sentence output given the original table records ‘R’, which means that the generation sticks to its plan<sup>14</sup>. We find the same tendency in case studies, showing model’s potentials to allow user control. Furthermore, we find the explicit planing would also reduce the repetitions of table records.

**Plan-based Generation helps few shot learning** In Table 3, we show that plan-based generation can make the generated text more faithful as well as have higher coverage compared with end-to-end training. Augmented plans further reduce the hallucinations in the generated text (much smaller  $R_{hallu}$ ) compared with its variant without auxiliary entities.

## 5.2 Case Studies

Firstly, we show an example taken from the test set of WIKIPERSON dataset in Fig 2. The baseline model, BART trained in the end-to-end fashion misses the ‘occupation’ attribute in the source table, while the other two models cover all the attributes in the table. The augmented plan-based

<sup>12</sup>This means these gold plans may contain two consecutive ‘SEP’ tokens and repetitive table records.

<sup>13</sup>PARENT is a better option which encourages the overlap with source table. However it also rewards the ngram overlap with reference text, which may contain hallucination. We would suggest human evaluation for faithfulness analysis.

<sup>14</sup>No ‘SEP’ token in the input plan means we want to describe each record in the input sequentially in one sentence.

generation does not contain any hallucinated content while some factually wrong information exist in the baseline and plan-based methods.

Then we show in Fig 3 that the generated text from augmented plan-based method strictly sticks to the corresponding plans. It also shows that the augmented plan-based method has the potential to generate more diverse content by exploiting different plans.

## 6 Related Work

**Faithfulness Evaluation:** Prior work evaluates the faithfulness in generation by human (Chen and Bansal 2018) or automatic metrics using natural language inference (Kryściński et al. 2019; Falke et al. 2019), information extraction (Zhang et al. 2019; Goodrich et al. 2019) or question answering (Scialom et al. 2019; Eyal, Baumel, and Elhadad 2019; Wang, Cho, and Lewis 2020). For faithfulness evaluation, prior work introduces soft constraints (Tian et al. 2019; Wang et al. 2020b), template-like (Liu et al. 2017; Wiseman, Shieber, and Rush 2018; Shen et al. 2020; Li and Rush 2020; Ye et al. 2020) or controllable generation (Peng et al. 2018; Fan, Lewis, and Dauphin 2019; Shen et al. 2019; Parikh et al. 2020), which encourage the overlap between given structured data and generated text. Some work also incorporate source input (table or text) in evaluation (Liu et al. 2019b; Dhingra et al. 2019; Wang et al. 2020a).

**Plan-base Generation:** Most work before deep learning era treats data-to-text as two sub-tasks: content selection and surface realization (Reiter and Dale 1997; Duboue and McKeown 2002; Barzilay and Lapata 2005, 2006; Belz 2008), which carefully learns the alignment between data and text (Liang, Jordan, and Klein 2009; Angeli, Liang, and Klein 2010; Kim and Mooney 2010; Konstas and Lapata 2013). Recently, end-to-end learning becomes a trend (Mei, Bansal, and Walter 2016; Chisholm, Radford, and Hachey 2017; Kaffee et al. 2018; Jhamtani et al. 2018; Bao et al. 2018; Liu et al. 2019a; Dušek, Novikova, and Rieser 2020) in this field. Among them, some work introduces differentiable planning modules (Sha et al. 2018; Laha et al. 2018; Puduppully, Dong, and Lapata 2018). Our paper focuses on a two-stage generation which incorporate separate text planner (Ferreira et al. 2019; Moryossef, Dagan, and Goldberg 2019; Ma et al. 2019). The separate planning methods could be easier to deploy and debug in the real world scenarios, it also shares the burden of end-to-end learning with two separate modules.

## 7 Conclusion

We try to evaluate faithfulness in table-to-text generation by entity-centric statistics on table coverage and hallucination in text, which help us analyze the correlation between training data quality and generation fidelity. Then we accordingly propose two methods based on the analysis: augmented training by incorporating auxiliary entity information and instance selection based on faithfulness ranking. We show these methods improve generation faithfulness on both automatic and human evaluations.

## Acknowledgments

We would like to thank the anonymous reviewers for the helpful discussions and suggestions. Our work is supported by the National Key Research and Development Program of China No.2020AAA0106701, NSFC project No.U19A2065, No.61876004, and Beijing Academy of Artificial Intelligence (BAAI).

## References

- Angeli, G.; Liang, P.; and Klein, D. 2010. A simple domain-independent probabilistic approach to generation. In *EMNLP 2010*, 502–512.
- Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. Neural Machine Translation by Jointly Learning to Align and Translate. *CoRR* abs/1409.0473. URL <http://arxiv.org/abs/1409.0473>.
- Bao, J.; Tang, D.; Duan, N.; Yan, Z.; Lv, Y.; Zhou, M.; and Zhao, T. 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*, 5020–5027. URL <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16138>.
- Barzilay, R.; and Lapata, M. 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*, 331–338. Association for Computational Linguistics.
- Barzilay, R.; and Lapata, M. 2006. Aggregation via set partitioning for natural language generation. In *NAACL*, 359–366. Association for Computational Linguistics.
- Belz, A. 2008. Automatic generation of weather forecast texts using comprehensive probabilistic generation-space models. *Natural Language Engineering* 14(4): 431–455.
- Chen, Y.-C.; and Bansal, M. 2018. Fast abstractive summarization with reinforce-selected sentence rewriting. *arXiv preprint arXiv:1805.11080*.
- Chisholm, A.; Radford, W.; and Hachey, B. 2017. Learning to generate one-sentence biographies from wikidata. *arXiv preprint arXiv:1702.06235*.
- Colin, E.; Gardent, C.; M’rabet, Y.; Narayan, S.; and Perez-Beltrachini, L. 2016. The webnlg challenge: Generating text from dbpedia data. In *Proceedings of the 9th International Natural Language Generation conference*, 163–167.
- Dhingra, B.; Faruqui, M.; Parikh, A.; Chang, M.-W.; Das, D.; and Cohen, W. W. 2019. Handling divergent reference texts when evaluating table-to-text generation. *arXiv preprint arXiv:1906.01081*.
- Duboue, P. A.; and McKeown, K. R. 2002. Content planner construction via evolutionary algorithms and a corpus-based fitness function. In *Proceedings of INLG 2002*, 89–96.
- Dušek, O.; Novikova, J.; and Rieser, V. 2019. Evaluating the State-of-the-Art of End-to-End Natural Language Generation: The E2E NLG Challenge. *arXiv preprint arXiv:1901.11528* URL <https://arxiv.org/abs/1901.11528>.
- Dušek, O.; Novikova, J.; and Rieser, V. 2020. Evaluating the state-of-the-art of end-to-end natural language generation: The E2E NLG Challenge. *Computer Speech & Language* 59: 123–156.
- Eyal, M.; Baumel, T.; and Elhadad, M. 2019. Question answering as an automatic evaluation metric for news article summarization. *arXiv preprint arXiv:1906.00318*.
- Falke, T.; Ribeiro, L. F.; Utama, P. A.; Dagan, I.; and Gurevych, I. 2019. Ranking generated summaries by correctness: An interesting but challenging application for natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2214–2220.
- Fan, A.; Lewis, M.; and Dauphin, Y. 2019. Strategies for structuring story generation. *arXiv preprint arXiv:1902.01109*.
- Ferreira, T. C.; van der Lee, C.; van Miltenburg, E.; and Kraemer, E. 2019. Neural data-to-text generation: A comparison between pipeline and end-to-end architectures. *arXiv preprint arXiv:1908.09022*.
- Goodrich, B.; Rao, V.; Liu, P. J.; and Saleh, M. 2019. Assessing the factual accuracy of generated text. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 166–175.
- Jhamtani, H.; Gangal, V.; Hovy, E.; Neubig, G.; and Berg-Kirkpatrick, T. 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, 1661–1671.
- Kaffee, L.-A.; Elsahar, H.; Vougiouklis, P.; Gravier, C.; Laforest, F.; Hare, J.; and Simperl, E. 2018. Learning to generate wikipedia summaries for underserved languages from wikidata. *arXiv preprint arXiv:1803.07116*.
- Kim, J.; and Mooney, R. J. 2010. Generative alignment and semantic parsing for learning from ambiguous supervision. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, 543–551. Association for Computational Linguistics.
- Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.
- Konstas, I.; and Lapata, M. 2013. A global model for concept-to-text generation. *Journal of Artificial Intelligence Research* 48: 305–346.
- Kryściński, W.; McCann, B.; Xiong, C.; and Socher, R. 2019. Evaluating the factual consistency of abstractive text summarization. *arXiv arXiv:1910*.
- Laha, A.; Jain, P.; Mishra, A.; and Sankaranarayanan, K. 2018. Scalable Micro-planned Generation of Discourse from Structured Data. *CoRR* abs/1810.02889. URL <http://arxiv.org/abs/1810.02889>.
- Landis, J. R.; and Koch, G. G. 1977. The measurement of observer agreement for categorical data. *biometrics* 159–174.
- Lebret, R.; Grangier, D.; and Auli, M. 2016. Neural text generation from structured data with application to the biography domain. *arXiv preprint arXiv:1603.07771*.
- Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; and Zettlemoyer, L. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.
- Li, X. L.; and Rush, A. M. 2020. Posterior Control of Blackbox Generation. *arXiv preprint arXiv:2005.04560*.

- Liang, P.; Jordan, M. I.; and Klein, D. 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*, 91–99. Association for Computational Linguistics.
- Liu, T.; Luo, F.; Xia, Q.; Ma, S.; Chang, B.; and Sui, Z. 2019a. Hierarchical encoder with auxiliary supervision for neural table-to-text generation: Learning better representation for tables. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 6786–6793.
- Liu, T.; Luo, F.; Yang, P.; Wu, W.; Chang, B.; and Sui, Z. 2019b. Towards comprehensive description generation from factual attribute-value tables. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 5985–5996.
- Liu, T.; Wang, K.; Sha, L.; Chang, B.; and Sui, Z. 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*, 4881–4888. URL <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16599>.
- Liu, T.; Wei, B.; Chang, B.; and Sui, Z. 2017. Large-scale simple question generation by template-based seq2seq learning. In *National CCF Conference on Natural Language Processing and Chinese Computing*, 75–87. Springer.
- Ma, S.; Yang, P.; Liu, T.; Li, P.; Zhou, J.; and Sun, X. 2019. Key Fact as Pivot: A Two-Stage Model for Low Resource Table-to-Text Generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 2047–2057. Florence, Italy: Association for Computational Linguistics. doi:10.18653/v1/P19-1197. URL <https://www.aclweb.org/anthology/P19-1197>.
- Maynez, J.; Narayan, S.; Bohnet, B.; and McDonald, R. 2020. On Faithfulness and Factuality in Abstractive Summarization. *arXiv preprint arXiv:2005.00661*.
- Mei, H.; Bansal, M.; and Walter, M. R. 2016. What to talk about and how? Selective Generation using LSTMs with Coarse-to-Fine Alignment. In *NAACL HLT 2016*, 720–730. URL <http://aclweb.org/anthology/N/N16/N16-1086.pdf>.
- Moryossef, A.; Dagan, I.; and Goldberg, Y. 2019. Improving Quality and Efficiency in Plan-based Neural Data-to-Text Generation. *arXiv preprint arXiv:1909.09986*.
- Moryossef, A.; Goldberg, Y.; and Dagan, I. 2019. Step-by-step: Separating planning from realization in neural data-to-text generation. *arXiv preprint arXiv:1904.03396*.
- Ott, M.; Edunov, S.; Baevski, A.; Fan, A.; Gross, S.; Ng, N.; Grangier, D.; and Auli, M. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.
- Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. BLEU: a method for automatic evaluation of machine translation. In *ACL2002*, 311–318.
- Parikh, A. P.; Wang, X.; Gehrmann, S.; Faruqui, M.; Dhingra, B.; Yang, D.; and Das, D. 2020. ToTTo: A Controlled Table-To-Text Generation Dataset. *arXiv preprint arXiv:2004.14373*.
- Peng, N.; Ghazvininejad, M.; May, J.; and Knight, K. 2018. Towards controllable story generation. In *Proceedings of the First Workshop on Storytelling*, 43–49.
- Perez-Beltrachini, L.; and Lapata, M. 2018. Bootstrapping generators from noisy data. *arXiv preprint arXiv:1804.06385*.
- Puduppully, R.; Dong, L.; and Lapata, M. 2018. Data-to-Text Generation with Content Selection and Planning. *arXiv preprint arXiv:1809.00582*.
- Qi, P.; Zhang, Y.; Zhang, Y.; Bolton, J.; and Manning, C. D. 2020. Stanza: A Python Natural Language Processing Toolkit for Many Human Languages. *arXiv preprint arXiv:2003.07082*.
- Reiter, E.; and Dale, R. 1997. Building applied natural language generation systems. *Natural Language Engineering* 3(1): 57–87.
- Scialom, T.; Lamprier, S.; Piwowarski, B.; and Staiano, J. 2019. Answers unite! unsupervised metrics for reinforced summarization models. *arXiv preprint arXiv:1909.01610*.
- See, A.; Liu, P. J.; and Manning, C. D. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.
- Sha, L.; Mou, L.; Liu, T.; Poupart, P.; Li, S.; Chang, B.; and Sui, Z. 2018. Order-Planning Neural Text Generation From Structured Data. In McIlraith, S. A.; and Weinberger, K. Q., eds., *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*, 5414–5421. AAAI Press. URL <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/view/16203>.
- Shen, X.; Chang, E.; Su, H.; Zhou, J.; and Klakow, D. 2020. Neural Data-to-Text Generation via Jointly Learning the Segmentation and Correspondence. *arXiv preprint arXiv:2005.01096*.
- Shen, X.; Suzuki, J.; Inui, K.; Su, H.; Klakow, D.; and Sekine, S. 2019. Select and Attend: Towards Controllable Content Selection in Text Generation. *arXiv preprint arXiv:1909.04453*.
- Tian, R.; Narayan, S.; Sellam, T.; and Parikh, A. P. 2019. Sticking to the Facts: Confident Decoding for Faithful Data-to-Text Generation. *arXiv preprint arXiv:1910.08684*.
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. In *Advances in neural information processing systems*, 5998–6008.
- Wang, A.; Cho, K.; and Lewis, M. 2020. Asking and Answering Questions to Evaluate the Factual Consistency of Summaries. *arXiv preprint arXiv:2004.04228*.
- Wang, K.; Liu, T.; Chang, B.; and Sui, Z. 2020a. An Anchor-Based Automatic Evaluation Metric for Document Summarization. In *Proceedings of the 28th International Conference on Computational Linguistics*, 5696–5701.
- Wang, Q.; Pan, X.; Huang, L.; Zhang, B.; Jiang, Z.; Ji, H.; and Knight, K. 2018. Describing a knowledge base. *arXiv preprint arXiv:1809.01797*.
- Wang, Z.; Wang, X.; An, B.; Yu, D.; and Chen, C. 2020b. Towards Faithful Neural Table-to-Text Generation with Content-Matching Constraints. *arXiv preprint arXiv:2005.00969*.
- Wiseman, S.; Shieber, S. M.; and Rush, A. M. 2018. Learning neural templates for text generation. *arXiv preprint arXiv:1808.10122*.
- Ye, R.; Shi, W.; Zhou, H.; Wei, Z.; and Li, L. 2020. Variational Template Machine for Data-to-Text Generation. *arXiv preprint arXiv:2002.01127*.
- Zhang, Y.; Merck, D.; Tsai, E. B.; Manning, C. D.; and Langlotz, C. P. 2019. Optimizing the Factual Correctness of a Summary: A Study of Summarizing Radiology Reports. *arXiv preprint arXiv:1911.02541*.