A Simple Recipe towards Reducing Hallucination in Neural Surface Realisation

Feng Nie[1*] **Jin-Ge Yao**[2] **Jinpeng Wang**[2] **Rong Pan**[1] **Chin-Yew Lin**[2]
[1]Sun Yat-Sen University [2]Microsoft Research Asia
[1]fengniesysu@gmail.com, [1]panr@sysu.edu.cn
[2]{jinge.yao, jinpwa, cyl}@microsoft.com

Abstract

Recent neural language generation systems often *hallucinate* contents (i.e., producing irrelevant or contradicted facts), especially when trained on loosely corresponding pairs of the input structure and text. To mitigate this issue, we propose to integrate a language understanding module for data refinement with self-training iterations to effectively induce strong equivalence between the input data and the paired text. Experiments on the E2E challenge dataset show that our proposed framework can reduce more than 50% relative unaligned noise from the original data-text pairs. A vanilla sequence-to-sequence neural NLG model trained on the refined data has improved on content correctness compared with the current state-of-the-art ensemble generator.

1 Introduction

Neural models for natural language generation (NLG) based on the encoder-decoder framework have become quite popular recently (Wen et al., 2015; Mei et al., 2016; Wiseman et al., 2017; Wen et al., 2017; Chisholm et al., 2017; Nie et al., 2018, *inter alia*). Albeit being appealing for producing fluent and diverse sentences, neural NLG models often suffer from a severe issue of content *hallucination* (Reiter, 2018a), which refers to the problem that the generated texts often contain information that is irrelevant to or contradicted with the input.

Given that similar issues have been less reported or noticed in the latest neural machine translation systems, we believe that the origin of the issue for neural NLG comes from the data side. Current datasets used for training neural NLG systems often include instances that do not contain the same amount of information from the input structure and the output text (Perez-Beltrachini and Gardent, 2017). There is no exception for datasets originally intended for surface realisation ("*how to say*") without focusing on content selection ("*what to say*"). Table 1 depicts an example, where the attribute `Rating=5 out of 5` in the input meaning representation (MR) is not verbalised in a reference text written by human, while the word *restaurant* in the reference should refer to an attribute value `EatType=Restaurant` not contained in the MR. Without explicit alignments in between MRs and the corresponding utterances for guidance, neural systems trained on such data often produce unexpected errors.

MR	Name	Rating	Price
	Golden Palace	*5 out of 5*	Cheap
Reference: **Golden Palace** is a restaurant specializing in breakfast in the **low price range**.			

Table 1: A loosely corresponded MR-text pair. **Bolded phrases** conforms to the MR, underlined words are domain-specific additional information, and *italic values* in the MR are not realised in the reference.

Previous work attempted at injecting indirect semantic control over the encoder-decoder architecture (Wen et al., 2015; Dušek and Jurcicek, 2016; Agarwal et al., 2018) or encouraging consistency during training (Chisholm et al., 2017), without essentially changing to the noisy training data. One exception is the Slug2Slug system (Juraska et al., 2018), where the authors use an aligner with manually written heuristic rules to filter out unrealized attributes from data.

In this paper, we propose a simple, automatic recipe towards reducing hallucination for neural surface realisers by enhancing the semantic equivalence between pairs of MRs and utterances. The steps include: (1) Build a language understanding module (ideally well-calibrated) that tries to parse the MR from an utterance; (2) Use it to reconstruct the correct attribute values revealed in the reference texts; (3) With proper confidence thresh-

[*] Contribution during internship at Microsoft.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pages 2673–2679
Florence, Italy, July 28 - August 2, 2019. ©2019 Association for Computational Linguistics

olding, conduct self-training to iteratively recover data pairs with identical or equivalent semantics.

Experiments on the E2E challenge benchmark (Novikova et al., 2017b) show that our framework can reduce more than 50% relative unaligned noise from original MR-text pairs, and a vanilla sequence-to-sequence model trained on the refined data can improve content correctness in both human and automatic evaluations, when compared with the current state-of-the-art neural ensemble system (Juraska et al., 2018).

2 Approach

Our proposed framework consists of a neural natural language understanding (NLU) module with iterative data refinement to induce semantically equivalent MR-text pairs from a dataset containing a moderate level of noise.

2.1 Notation

Formally, given a corpus with paired meaning representations and text descriptions $\{(R, X)\}_{i=1}^{N}$, the input MR $R = (r_1, \ldots, r_M)$ is a set of slot-value pairs $r_j = (s_j, v_j)$, where each r_j contains a slot s_j (e.g., `rating`) and a value v_j (e.g., `5 out of 5`). The corpus has M pre-defined slots , and each slot s_j has K_j unique categorical values $v_j \in (c_{j,1}, \ldots, c_{j,K_j})$. The corresponding utterance $X = (x_1, \ldots, x_T)$ is a sequence of words describing the MR.

2.2 Neural NLU Model

As shown in Figure 1, the NLU model consists of a self-attentive encoder and an attentive scorer.

Self-Attentive Encoder. The encoder produces the vector representations of slot-value pairs in MR and its paired utterance. A slot-value pair r can be treated as a short sequence $W = (w_1, \ldots, w_n)$ by concatenating words in its slot and value. The word sequence W is first represented as a sequence of word embedding vectors $(\mathbf{v}_1, \ldots, \mathbf{v}_n)$ from a pre-trained embedding matrix E, and then passed through a bidirectional LSTM layer to yield the contextualized representations $U^{sv} = (\mathbf{u}_1^{sv}, \ldots, \mathbf{u}_n^{sv})$. To produce a summary context vector for U^{sv}, we adopt the same self-attention structure in Zhong et al. (2018) to obtain the sentence vector $\mathbf{c}_s$, due to the effectiveness of self-attention modules over variable-length sequences. Similarly, we obtain the contextualized representations $\mathbf{U}^o = (\mathbf{u}_1^o, \ldots, \mathbf{u}_T^o)$ for the utterance X.

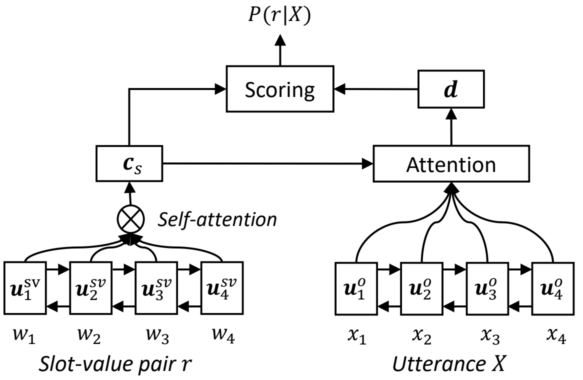

Figure 1: The structure of the neural NLU model.

Attentive Scorer. The scorer calculates the semantic similarity between a slot-value pair r (e.g., `Price=Cheap`) and the utterance X (e.g., reference in Table 1). Firstly, an attention layer is applied to select the most salient words in X related to r, which yields the attentive representation $\mathbf{d}$ of utterance X. Given the sentence vector $\mathbf{c}_s$ of the slot-value pair r and the attentive vector $\mathbf{d}$ of the utterance X, the normalized semantic similarity is defined as:

$$p(r|X) = \text{softmax}(-||\mathbf{d} - \mathbf{c}_s||_2), \text{ where}$$
$$\mathbf{d} = \sum_{t=1}^{T} b_t \mathbf{u}_t^o, \text{ with } b_t = \text{softmax}((\mathbf{u}_t^o)^T \mathbf{c}_s). \quad (1)$$

Model Inference. Each utterance X will be parsed to an MR $R^e = (r_1^e, \ldots, r_M^e)$, with each slot-value pair $r_j^e = (s_j, v_j)$ determined by selecting the candidate value v_j with the maximum semantic similarity for each slot s_j:

$$v_j = c_{j,k}, \quad k = \arg\max_k p(r_j^e = (s_j, c_{j,k})|X), \quad (2)$$

where $c_{j,k}$ denotes the kth categorical value for jth slot. Since an utterance may not describe any information about a specific slot s, we add a `NONE` value as a candidate value of each slot.

Model Training. The NLU model is optimized by minimizing the cross-entropy loss:

$$\mathcal{L}(\theta) = -\sum_{i}^{N} \sum_{j}^{M} \log p(r_{i,j}|X_i; \theta) \quad (3)$$

where θ denotes model parameters, and $r_{i,j}$ denotes the jth slot-value pair in the ith training MR.

2.3 Iterative Data Refinement

The performance of NLU can be inaccurate when trained on noisy data-text pairs. However, models trained on data with a moderate level of noise could still be well-calibrated. This could enable an iterative relabeling procedure, where we only take MRs produced by NLU with high confidence together with their utterances as new training MR-text pairs to bootstrap the NLU training.

Algorithm 1 describes the training procedure. We first pre-train the NLU model using the original data-text pairs for N_{pre} iterations. Then the NLU model parses relevant MR for every utterance in training data, which can be used as new training examples (Line 4). However, due to the inaccuracy of the NLU results, we only use a small portion (ϕ is set to 40% on validation) with high confidence. Moreover, as each MR consists of up to M slots with some of them being unreliable, we filter the slot-value pairs with slot probability below average according to slot confidence (Line 8 - 14). Finally, the NLU model is fine-tuned with the new training corpus D^e. This process is repeated for N_{tune} epochs. The final NLU model is leveraged to parse all utterances in the training corpus. The resulting MRs paired with original utterances form the refined training corpus for NLG.

Algorithm 1 Iterative Data Refinement

```
Require MR-text pairs $D = \{(R, X)\}_1^N$, confidence threshold $\phi$, pre-training epochs $N_{pre}$, tuning epochs $N_{tune}$,
 1: Train $\theta$ with Eq. 3 on $D$ for $N_{pre}$ iterations
 2: for $iter$ = 1 to $N_{tune}$ do
 3:   Reset self-training corpus $D^e = \{\}$
 4:   Parse the MR $R_i^e = (r_{i,1}^e, \dots, r_{i,M}^e)$ for every $X_i$ using Eq. 2
 5:   Slot confid. $p_j = \sum_{i=1}^{N} p(r_{i,j}^e|X_i)$ for $s_j$
 6:   MR confid. $f_i = \sum_{j=1}^{M} p(r_{i,j}^e|X_i)$ for $R_i^e$
 7:   Sort $\{(R^e, X)\}_1^N$ by MR confidence in reverse order
 8:   for $i$ = 1 to $\lfloor \phi \cdot N \rfloor$ do
 9:     for $j$ = 1 to $M$ do
10:       if $p(r_{i,j}^e|X_i) < p_j/N$ then
11:         Remove $r_{i,j}^e$ from $R_i^e$
12:       end if
13:     end for
14:     $D^e \leftarrow D^e \cup (R_i^e, X_i)$
15:   end for
16:   Update $\theta$ with Eq. 3 on $D^e$
17: end for
```

3 Experiments

3.1 Setup

Dataset. Our experiments are conducted on E2E challenge (Novikova et al., 2017b) dataset, which aims at verbalizing all information from the MR. It has 42,061, 4,672 and 4,693 MR-text pairs for training, validation and testing, respectively. Note that every input MR in this dataset has 8.65 different references on average. The test set has 630 unique input MRs. We examine the effectiveness of our proposed method in two aspects: 1) reducing the noise in data-text pairs (NLU), 2) reducing hallucinated contents in surface realisation (NLG).

Automatic metrics. The well-crafted rule-based aligner built by Juraska et al. (2018)[1] is adopted to approximately reflect the semantic correctness of NLU and NLG models. The error rate is calculated by matching the slot values in output utterance: Err $= \frac{M}{N}$, where N is the total number of MR-text pairs, and M is the number of wrong MR-text pairs which contain missing or conflict slots in the realization given its input MR. BLEU-4 (Papineni et al., 2002) is also reported, although currently neither BLEU nor any other automatic metrics could be convincingly used for evaluating language generation (Novikova et al., 2017a; Chaganty et al., 2018; Reiter, 2018b, *inter alia*).

Human Evaluation. We randomly sample 100 data-text pairs from test set and ask three crowd workers to manually annotate *missed* (M), *added* (A), and *contradicted* (C) slot values in NLG outputs with respect to the input MR, or *exact match* (E) if all slot values have been realized in the given utterance which contains no additional hallucinated information. When evaluating the NLU systems, *missed* and *added* slots refer to the opposite directions, respectively.

Compared Systems. Systems in comparison:

- `TGen` (Dušek et al., 2018): a sequence-to-sequence (Seq2Seq) model with reranking.
- `Slug2Slug` (Juraska et al., 2018): current state-of-the-art method on E2E challenge dataset. It is an ensemble model and uses a rule based aligner for data cleaning and reranking.

[1] We use the public available evaluation script in https://github.com/jjuraska/slug2slug/blob/master/slot_aligner/data_analysis.py

- Seq2Seq: a basic Seq2Seq model trained on original MR-text pairs with the copy mechanism (Gu et al., 2016; See et al., 2017).
- Seq2Seq+aug: Seq2Seq trained on the MR-text pairs reconstructed by pre-trained NLU.
- Seq2Seq+aug+iter: Seq2Seq trained on the MR-text pairs reconstructed by NLU model with iterative data refinement algorithm.
- Seq2Seq+aligner: Seq2Seq trained on the MR-text pairs produced by the rule based aligner (Juraska et al., 2018).

Implementation Details. For all models, we use fixed pre-trained GloVe vectors (Pennington et al., 2014) and character embeddings (Hashimoto et al., 2017). The dimensions of trainable hidden units in LSTMs are all set to 400. The epochs for pre-training N_{pre} and bootstrapping N_{tune} are all set to 5 on validation. During training, we regularize all layers with a dropout rate of 0.1. We use stochastic gradient descent (SGD) for optimisation with learning rate 0.1. The gradient is truncated by 5. For hyper-parameter ϕ, we conduct experiments with different values ($\phi = 0.2, 0.4, 0.6, 0.8, 1.0$), details in Appendix A.

3.2 Main Results

NLU Results. One challenge in E2E dataset is the need to account for the noise in the corpus as some of the MR-text pairs are not semantically equivalent due to the data collection process (Dušek et al., 2018). We examine the performance of the NLU module by comparing noise reduction of the reconstructed MR-text pairs with the original ones in both training and test sets. Table 2 shows the automatic results. Applying our NLU model with iterative data refinement, the error rates of refined MR-text pairs yields 23.33% absolute error reduction on test set. Human evaluation in Table 3 shows that our proposed method achieves 16.69% improvement on information equivalence between MR-text pairs. These results confirm the effectiveness of our method in reducing the unaligned data noise, and the large improvement (i.e, 15.09%) on exact match when applying self-training algorithm suggests the importance of iterative data refinement.

	Train Err(%)	Test Err(%)
Original data	35.50	37.59
NLU refined data	**16.31**	**14.26**
w/o self-training	25.14	22.69

Table 2: Automatic evaluation results of different NLU models on both training and test sets

	E(%)	M(%)	A(%)	C(%)
Original data	71.93	0	24.13	3.95
NLU refined data	**88.62**	5.45	2.48	3.47
w/o self-training	73.53	13.23	8.33	4.91

Table 3: Human evaluation results for NLU on test set (inter-annotator agreement: Fleiss' kappa = 0.855)

	BLEU(%)	Err(%)
TGen	65.90	18.09 (114/630)
Slug2Slug	66.19	6.51 (41/630)
Seq2Seq	66.15	69.37 (374/630)
Seq2Seq+aug	**66.49**	28.89 (182/630)
Seq2Seq+aug+iter	65.63	2.07 (13/630)
Seq2Seq+aligner	63.81	**1.75** (11/630)

Table 4: Automatic metrics for NLG

	E(%)	M(%)	A(%)	C(%)
TGen	78.49	15.12	2.69	3.3
Slug2Slug	91.36	2.98	0	5.66
Seq2Seq	44.07	50.65	4.03	0.65
Seq2Seq+aug+iter	**93.93**	3.36	2.69	0

Table 5: Human evaluation results for NLG (inter-annotator agreement: Fleiss' kappa = 0.832)

NLG Results. Table 4 presents the automatic results of different neural NLG systems. We can see that Seq2Seq+aug+iter achieves comparable BLEU score as Slug2Slug but with 4.44% error reduction on content correctness over Slug2Slug. Seq2Seq+aug+iter largely improves the content correctness over the baseline Seq2Seq with 67.3% error reduction. Besides, we also replace our NLU module with the rule based aligner crafted by Juraska et al. (2018) for data refinement to inspect the difference between our proposed method and manually designed rich heuristics. We can observe that these two models (Seq2Seq+aug+iter and Seq2Seq+aligner) achieve comparable performance, while our approach is fully automatic and requires no domain knowledge.

The human evaluation results are shown in Table 5. We can find that Seq2Seq+aug+iter improves 2.59% accuracy on exact match over Slug2Slug. Specifically, Slug2Slug augments original training data by only deleting additional slot values not realized in the utterance with an aligner, which is not capable of the situation where the given utterance contains incorrect or additional slot values and leads more con-

Utterance: Located in riverside, near Caf Sicilia, is the Phoenix, a French pub that is family-friendly and has average prices and an average rating.

Original MR: name[The Phoenix], eatType[pub], food[French], priceRange[20-25], area[riverside], customer rating[3 out of 5], familyFriendly[no], near[Caf Sicilia]

Refined MR: name[The Phoenix], eatType[pub], food[French], priceRange[moderate], area[riverside], customer rating[average], familyFriendly[yes], near[Caf Sicilia]

Table 6: Example for data refinement; The underscored item is incorrect.

MR	Name:[The Mill]; EatType:[pub]; Food:[Fast Food];PriceRange:[high]; FaimilyFriendly:[yes];Near:[Caf Sicilia]; Area:[riverside]; Rating:[average]
TGen	The Mill is a high priced family friendly fast food pub located near Caf Sicilia in the riverside area.
Slug2Slug	children friendly pub in the riverside area near Caf Sicilia. It has a high price range and a *high* customer rating
Seq2Seq	The Mill is a family friendly pub located near Caf Sicilia.
Seq2Seq+ aug+iter	The Mill is a children friendly fast food pub near Caf Sicilia in the riverside area. It has a high price range and an average customer rating.

Table 7: Examples of different system outputs.

tradicted errors. Our method can complement and correct original MR with additional slot values described in the paired texts to effectively alleviate generating contradicted facts. However, due to the imperfection of NLU model, our method may ignore part of slot values realized in utterances and produce some additional errors.

3.3 Case Study

Example for refined data. Table 6 depicts a case for one pair with originally inaccurate MR while being corrected by NLU module and iterative refinement. Our proposed method is capable of reducing the unaligned noise for original data.

Example for NLG. Table 7 shows the sentences generated by different NLG systems. `Seq2Seq` without any semantic control tends to generate shorter descriptions. `Slug2Slug` and `TGen` with reranker to control the content coverage can generate more input information, but still misses one input information and `Slug2Slug` produces a contradicted fact (i.e., customer rating). Our proposed method `Seq2Seq+aug+iter` trained on refined MR-text pairs, verbalises all the input information correctly, which shows the importance of data quality in terms of strong equivalence between MR and utterance.

4 Discussion

In this paper, we present a simple recipe to reduce the hallucination problem in neural language generation: introducing a language understanding module to implement confidence-based iterative data refinement. We find that our proposed method can effectively reduce the noise in the original MR-text pairs from the E2E dataset and improve the content coverage for standard neural surface realisation (no focus on content selection).

However, the currently presented approach still has two clear limitations. One is that this simple approach is implicitly built on an assumption of a moderate level of noise in the original data, which makes it possible to bootstrap a well-calibrated NLU module. We are still on the way to find out solutions for cases with huge noise (Perez-Beltrachini and Lapata, 2018; Wiseman et al., 2017), where heavy manual intervention or external knowledge should be desperately needed.

The other limitation of this preliminary work is that it currently overlooks the challenges of lexical choices for quantities, degrees, temporal expressions, etc, which are rather difficult to learn merely from data and should require additional commonsense knowledge. An example case is in Table 6, where the original `priceRange=20-25` is refined to be `priceRange=moderate`, which enhances the correspondence between the MR and the text but sidesteps the lexical choice for numbers which requires localised numerical commonsense. Additional modules for lexical choices should be expected for a refined system.

5 Acknowledgement

We thank Zhirui Zhang, Shuangzhi Wu, and the anonymous reviewers for helpful comments. Feng Nie is partially supported by National Key R&D Program of China (2018YFB1004404) and Key R&D Program of Guangdong Province (2018B010107005). The contact author of this paper, according to the meaning given to this role by Sun Yat-Sen University, is Rong Pan.

References

Shubham Agarwal, Marc Dymetman, and Eric Gaussier. 2018. Char2char generation with reranking for the E2E NLG challenge. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 451–456, Tilburg University, The Netherlands. Association for Computational Linguistics.

Arun Chaganty, Stephen Mussmann, and Percy Liang. 2018. The price of debiasing automatic metrics in natural language evalaution. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 643–653, Melbourne, Australia. Association for Computational Linguistics.

Andrew Chisholm, Will Radford, and Ben Hachey. 2017. Learning to generate one-sentence biographies from Wikidata. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 633–642, Valencia, Spain. Association for Computational Linguistics.

Ondřej Dušek and Filip Jurcicek. 2016. Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. Association for Computational Linguistics.

Ondřej Dušek, Jekaterina Novikova, and Verena Rieser. 2018. Findings of the E2E NLG challenge. In *Proceedings of the 11th International Conference on Natural Language Generation*, pages 322–328, Tilburg University, The Netherlands. Association for Computational Linguistics.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640, Berlin, Germany. Association for Computational Linguistics.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A joint many-task model: Growing a neural network for multiple NLP tasks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1923–1933, Copenhagen, Denmark. Association for Computational Linguistics.

Juraj Juraska, Panagiotis Karagiannis, Kevin Bowden, and Marilyn Walker. 2018. A deep ensemble model with slot alignment for sequence-to-sequence natural language generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 152–162, New Orleans, Louisiana. Association for Computational Linguistics.

Hongyuan Mei, Mohit Bansal, and Matthew R. Walter. 2016. What to talk about and how? selective generation using LSTMs with coarse-to-fine alignment. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 720–730, San Diego, California. Association for Computational Linguistics.

Feng Nie, Jinpeng Wang, Jin-Ge Yao, Rong Pan, and Chin-Yew Lin. 2018. Operation-guided neural networks for high fidelity data-to-text generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3879–3889, Brussels, Belgium. Association for Computational Linguistics.

Jekaterina Novikova, Ondřej Dušek, Amanda Cercas Curry, and Verena Rieser. 2017a. Why we need new evaluation metrics for NLG. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2241–2252, Copenhagen, Denmark. Association for Computational Linguistics.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. 2017b. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 201–206, Saarbrücken, Germany. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Laura Perez-Beltrachini and Claire Gardent. 2017. Analysing data-to-text generation benchmarks. In *Proceedings of the 10th International Conference on Natural Language Generation*, pages 238–242, Santiago de Compostela, Spain. Association for Computational Linguistics.

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, Volume 1 (Long Papers)*, pages 1516–1527, New Orleans, Louisiana. Association for Computational Linguistics.

Ehud Reiter. 2018a. Hallucination in neural NLG.

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

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Tsung-Hsien Wen, Milica Gasic, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721, Lisbon, Portugal. Association for Computational Linguistics.

Tsung-Hsien Wen, David Vandyke, Nikola Mrkšić, Milica Gasic, Lina M. Rojas Barahona, Pei-Hao Su, Stefan Ultes, and Steve Young. 2017. A network-based end-to-end trainable task-oriented dialogue system. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 438–449, Valencia, Spain. Association for Computational Linguistics.

Sam Wiseman, Stuart Shieber, and Alexander Rush. 2017. Challenges in data-to-document generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2253–2263, Copenhagen, Denmark. Association for Computational Linguistics.

Victor Zhong, Caiming Xiong, and Richard Socher. 2018. Global-locally self-attentive encoder for dialogue state tracking. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1458–1467, Melbourne, Australia. Association for Computational Linguistics.

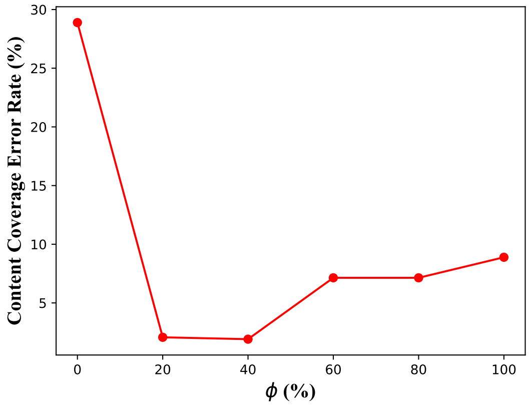

Figure 2: The effect of hyperparameter ϕ for NLG content coverage performance.

A Effect of ϕ on NLG model

The parameter ϕ controls the proportion of relevant MRs produced by NLU model for iterative training. Figure 2 shows its influence for NLG on the content coverage measurement. The experimental result shows NLG models trained on data produced by self-training achieve error reduction in content coverage. As the NLU model can bring inaccurate instances when performing iterative data augmentation, controlling the proportion ϕ from 20% to 40% can yield better results compared to introducing all the MRs produced by NLU for self-training.