Controlled Hallucinations: Learning to Generate Faithfully from Noisy Data

Katja Fillippova
Google Research, Berlin, Germany
katjaf@google.com

Abstract

Neural text generation (data- or text-to-text) demonstrates remarkable performance when training data is abundant which for many applications is not the case. To collect a large corpus of parallel data, heuristic rules are often used but they inevitably let noise into the data, such as phrases in the output which cannot be explained by the input. Consequently, models pick up on the noise and may hallucinate – generate fluent but unsupported text. Our contribution is a simple but powerful technique to treat such hallucinations as a controllable aspect of the generated text, without dismissing any input and without modifying the model architecture. On the WikiBio corpus (Lebret et al., 2016), a particularly noisy dataset, we demonstrate the efficacy of the technique both in an automatic and in a human evaluation.

1 Introduction

Deep neural network-based (DNN) models have demonstrated remarkable performance on a multitude of text-to-text (Bahdanau et al., 2015; Rothe et al., 2019; Narayan et al., 2018; Rush et al., 2015, inter alia) as well as data-to-text generation tasks (Wiseman et al., 2017; Puduppully et al., 2019, inter alia). To reach high performance, DNN models require a large training corpus which is normally not readily available. Indeed, it is rare to have a sufficiently large human-curated corpus of parallel data (Koehn, 2005), and researchers have come up with heuristic rules to mine input-output pairs on a large scale (Hermann et al., 2015; Rush et al., 2015; Narayan et al., 2018). No matter how powerful, DNN models are known to be sensitive to data artifacts (Kaushik and Lipton, 2018) and pick on the noise in the training data.

While hallucinations have not been defined formally, the term is standardly used to refer to the generated content which is either unfaithful to the input, or nonsensical (Maynez et al., 2020). In our work we are concerned with the former hallucination kind which is primarily caused by imperfect quality of the training data. If the data are noisy, how can one reduce the chances of hallucinating? One may try to improve the quality of a dataset and clean it from phrases for which a clear support in the input is missing, or augment the input with information found only in the output. The former path is risky as it easily results in ungrammatical targets. The latter approach of enforcing a stronger alignment between inputs and outputs has been tried previously but it assumes a moderate amount of noise in the data (Nie et al., 2019; Dušek et al., 2019). Alternatively, one can leave the data as is and try to put more pressure on the decoder to pay attention to the input at every generation step (Tian et al., 2019). This requires significant modifications to the model and may make it harder for the decoder to generate fluent and diverse text as found in the targets.

Figure 1: Two WikiBio sources and targets with example attributes: tense and length can be read-off the target directly. When added to the input, the model gets a knob to control for length and tense. We propose to estimate the noise degree by comparing the source with the target thus obtaining a hallucination knob.
In contrast to the described approaches, our proposal is to train the model on the data as is without modifying the decoding (and encoding) architecture but instead introduce a handle on the input side to control the degree of hallucination (Fig. 1). With this "hallucination knob" one can minimize (or maximize) the amount of unsupported information in the output during generation (Fig. 2). The hallucination or noise degree of every training instance is estimated separately and converted into a categorical value which becomes part of the input, like in a controlled generation setting (Ficler and Goldberg, 2017; Raffel et al., 2019). We introduce a simple technique to measure the amount of noise in every training example which is based on the intuition that whenever a language model (LM) has a smaller loss than a conditional generator during forced-path decoding, it is a good signal that the next token cannot be explained by the input.

We consider a particularly noisy dataset, WikiBio (Lebret et al., 2016), which has been found to have extra information in 62% of the references (Dhingra et al., 2019) and where 1:1 correspondence between the input and the output never holds (Perez-Beltrachini and Gardent, 2017). Our models demonstrate superior performance over the model of Liu et al. (2018) which reports SoTA BLEU results on WikiBio. In sum, our contributions are (1) a novel idea of controlling hallucinations which requires no modification to the model, (2) a data-and task-independent technique of implementing this idea and (3) three-way evaluation with human raters which confirms that faithfulness does not need to be traded for coverage.

### 3 Detecting Hallucinations in the Training Data

To detect hallucinations in the training data targets, we consider (3.1) an overlap-based technique, which has a clear foundation but cannot be applied to any seq2seq task, and (3.2) a simple procedure applicable in any setting. Both methods give us a hallucination score \( \text{hal} \in [0, 1] \) for every source-target pair. The scores are converted into categorical values with quantiles: five intervals, each covering 20% of the full range, are introduced and a special tag is used for every interval. During training, the data2text model learns an embedding for each of the five tags and during inference the tag with the lowest hallucination value, \( \text{hal}_0 \), is used (Fig. 2).

#### 3.1 Word Overlap

When the source and the target are similar on the token level, one can use word overlap between them to estimate how many words unsupported by the source are present in the target. More formally we define \( \text{hal} \) as a function of a source-target pair \((x, y)\):

\[
\text{hal}_{WO}(x, y) = 1 - \frac{|W_y \cap W_x|}{|W_y|}
\]  

(1)
where \( W \) is the set of words (in the source or the target). Note that this overlap technique only makes sense when the source and the target are in the same language and are known to be very similar. The second condition may hold to different degrees even within a dataset: for example, news publishers differ in whether they tend to write more abstractive or extractive headlines (Zhang et al., 2018).

3.2 When a LM Knows Better

It has often been observed that hallucinations can be partially explained by a strong LM component in the decoder which tends to select the next token as a likely continuation of the sequence generated so far (Rohrbach et al., 2018; Dušek et al., 2019, inter alia). This observation motivates our second method of detecting hallucinations.

Given a source and a target, how can one know if a target token \( w_{y_t} \) is unsupported by the source? Consider two generation models with an identical architecture trained on the same dataset:

- \( \text{LM} \): an unconditional LM which generates the next token based on the decoded prefix and which is trained only on the targets,

- \( \text{LM}_x \): a conditional LM which is also trained to generate targets but which is additionally informed about the source.

On the task of generating targets from the source, during forced-path decoding, we expect \( \text{LM}_x \) to perform better as long as the target is supported by the source because, unlike \( \text{LM} \), it anticipates what may come next. For example, \( \text{LM} \) will assign roughly the same probability to every month of the year while \( \text{LM}_x \) will put the mass on one month, provided that the birth month is listed in the source table. On the contrary, whenever the next token is unexpected, it is \( \text{LM} \) which reserves a small probability for it because it has been trained to predict whatever is likely to continue a given prefix, while \( \text{LM}_x \) puts more probability mass on tokens related to the source. The more faithful \( \text{LM}_x \), the more pronounced this difference is.

Based on this intuition, to compute a single \( \text{hal}_{LM} \) value for a source-target pair, we compute the ratio of tokens predicted incorrectly by \( \text{LM}_x \) for which \( \text{LM} \) got a smaller loss than \( \text{LM}_x \) to the total target length \( |y| \) (\( w_{y_t} \) denotes the \( t \)'th token in the target \( y \); \( \tilde{w}_{y_t} \) denotes the token predicted by \( \text{LM}_x \) at position \( t \)):

\[
\text{hal}_{LM}(x, y) = \frac{1}{|y|} \sum_{t=1}^{|y|} \left[ w_{y_t, \text{LM}} \neq w_{y_t, \text{LM}_x} \land p_{\text{LM}}(w_{y_t}) > p_{\text{LM}_x}(w_{y_t}) \right]
\]

(2)

For example, given a prefix \( \text{first-name last-name is a, target first-name last-name is a french writer} \) and a source mentioning the profession (\textit{writer}) but not the nationality (\textit{french}), \( \text{LM}_x \) will assign a high probability on the next token being the profession while \( \text{LM} \) will have a small probability for any continuation, including a nationality. The smaller loss of \( \text{LM} \) on the next token (\textit{french}) will signalize the presence of a hallucination.

4 Experiments

The primary goal of the experiments is to verify whether hallucinations can indeed be controlled for: we compare a seq2seq model trained on the WikiBio data as is with the same model trained with the noise attribute annotated (by the Word Overlap and LM-based methods). We also evaluate the model of Liu et al. (2018), which reported SoTA BLEU results, and the model of Tian et al. (2019), which was designed to generate hallucination-free output.

In our automatic evaluation, we measure BLEU (Papineni et al., 2002) as well as the recently introduced PARENT metric designed specifically for data2text tasks and verified on WikiBio (Dhingra et al., 2019). Unlike BLEU, it compares the output not only with the reference but also measures how much of it is entailed by the input table.

While PARENT is much more appropriate than BLEU for data2text evaluation, in its standard implementation it may miss a paraphrase of a table field in the target sentence (e.g., \textit{spouse} hardly ever occurs on the target side). It may also assign points for a match with the reference which is unsupported by the table. Thus, it can give a wrong estimate of both precision and recall and should be complemented with a human evaluation if two similar performing models are compared.

To this end, in our experiments with human raters we measure fluency and faithfulness of generated sentences as well as coverage: we need all three as we do not want to favor models which generate fluent and faithful but short sentences because fluency and faithfulness can be trivially achieved with a handful of templates.
Fluent sentences are natural and grammatically correct (Fluent, Mostly fluent and Not fluent). We report the percentage of fluent sentences.

Faithful sentences express information supported by the table or by non-expert background knowledge (Faithful, Mostly faithful and Not faithful). Since there is a grey area of what can be inferred from the table without expert knowledge\(^1\), we report the percentage of Faithful and Mostly faithful sentences to the total.

Coverage counts table cells with the information expressed in the generated sentence.

Faithfulness and coverage can be seen as precision and recall metrics, respectively. We randomly selected 200 examples from the test set and collected three ratings for every input table and a generated output.

### 4.1 Model

We train a bi-LSTM encoder-decoder model\(^2\) on WikiBio tokenized into SentencePieces (Kudo and Richardson, 2018). The input table is converted into a string with `<row>` and `<col>` special tags indicating fields and values. We use the standard train-development-test split and do no pretraining. The same model architecture is used for \(LM\) and \(LM_x\). That is, the default seq2seq model which we compare against is also used as \(LM_x\). It differs from \(LM\) in that the latter takes no input and the only difference to the controlled models is that they prepend the input with a single hallucination tag.

### 4.2 Removing Noisy Examples

The first question we address is whether a data cleaning procedure would already result in good quality sentences. As Table 1 indicates, the cleanest 20% of the data with the smallest \(hal_{WO}\) is not sufficient to train a competitive model. The predictions are more precise than those of the default model but the PARENT-recall and also the BLEU scores are low. Given a big gap to all other models, we do not evaluate this variant of the seq2seq model with humans.

\(^1\)For example, place of birth: Paris suggests that the person is French although an exception is thinkable. position: midfielder and club: Juventus imply that the person is a soccer player. We observed that mostly faithful is often used for such inferences.

\(^2\)Model details: two encoder and a single decoder layers; 256 dimensions for token embeddings, the size of the hidden cell is 128; Adam optimizer and attention; learning rate of 0.001 with a decay factor; 16,000 tokens in the vocabulary.

### 4.3 Results

All the models perform similar in terms of PARENT-F, the differences are in PARENT precision and recall. Liu-et-al. gets the best PARENT-F score but it comes at the cost of much lower precision than any other model which is exactly the problem we are trying to battle: unfaithful generations are arguably more harmful than missing information. Hence we turn to the human evaluation to draw final conclusions.

As perfect coverage and faithfulness can be achieved by concatenating the fields of an input table, we first verify that the generated sentences sound natural to humans. On this dimension, all the models designed to reduce hallucinations perform comparably well (93-96%) and better than the models which do not address this problem (Liu-et-al., seq2seq).

Supporting the main hypothesis of our work, the two controlled versions of the seq2seq model produce significantly more faithful sentences than both Liu-et-al. and the default seq2seq: the gap to the default SEQ2SEQ version is 15-25 points (13-15, if mostly faithful is included). Compared with Tian-et-al., our techniques are comparable or better if only faithful ratings are considered and worse if also mostly-faithful results are included. However, Tian-et-al. requires significant modifications to the model (e.g., using the variational Bayes objective) which may not always be implementable. More importantly, Tian-et-al. is the model with the significantly smaller coverage than any other model (4.1 vs. 4.5 for \(hal_{LM}\)). In terms of coverage, the LM-based version of the controlled generator achieves higher coverage than the overlap-based one, equalling the default seq2seq.

The last point is the main result of our work: it is possible to keep the recall of the default model (SEQ2SEQ) while dramatically improving precision. Moreover, no assumptions about the similarity between the sources and targets in the training data are needed as the \(hal_{WO}\) method demonstrates.

| Model                                      | BLEU-4 | PARENT (P / R / F) |
|--------------------------------------------|--------|--------------------|
| seq2seq (clean data)                      | 31.9   | 76.3/37.7/48.1     |
| Liu-et-al.                                 | 45.4   | 74.0/44.0/52.8     |
| Tian-et-al.                                | 38.1   | 79.5/40.6/51.4     |
| seq2seq                                   | 41.0   | 75.9/42.0/51.8     |
| seq2seq + \(hal_{WO}\)                    | 36.5   | 79.5/40.9/51.7     |
| seq2seq + \(hal_{LM}\)                    | 36.1   | 78.5/40.3/50.9     |

Table 1: Automatically computed metrics.
Table 2: Human evaluation results.

| Method                  | Fluent | Faithful (F+MF) | Coverage |
|-------------------------|--------|-----------------|----------|
| Liu-et-al.              | 89%    | 41 (55%) %      | 4.7      |
| Tian-et-al.             | 95%    | 68 (92%) %      | 4.1      |
| seq2seq                 | 90%    | 51 (67%) %      | 4.5      |
| seq2seq + halWO         | 93%    | 76 (82%) %      | 4.3      |
| seq2seq + halLM         | 96%    | 66 (80%) %      | 4.5      |

5 Discussion

Comparing the two methods of estimating the amount of hallucinations in a target, for applications where the input and the output use the same vocabulary with a comparable term distribution the overlap method may be better as it has a clear foundation. The LM-based method that we proposed has an important advantage that it makes no assumptions about the data. In our WikiBio experiment it also produced better results in the human evaluation, presumably because it allowed for paraphrasing and straightforward inferences. For example, the target "ozren nedoklan was a yugoslav footballer and manager" has a high halWO score because the source table has no occupation field and does not mention yugoslav. The halLM score of that example is zero because footballer and manager can be inferred from the names of the clubs and the manageryears fields in the source.

Possible extensions It should be emphasized that alternative methods of detecting noise can be explored and may perform better in the controlled-hallucination framework. For example, it is possible to measuring target-source similarity in an embedded space or use word alignment tools to find unsupported information.

While here we have focused on eliminating hallucinations, one can think of applications where one is interested in generating adversarial sentences which sound fluent but are guaranteed to include unsupported information. Figure 2 shows how the amount of hallucinations in the output increases following the value of the hallucination knob.

Why is BLUE so different? It is striking that while all the models tested outperform Liu et al. (2018) in terms of PARENT and human evaluation scores, none could approach its BLEU performance. We do not have an explanation of why this is so but note that our results are in line with the review by Reiter (2018) who concludes that BLEU is an inappropriate metric for generation tasks other than MT.

Can we measure length instead of noise? One may wonder whether an even simpler approach of controlling for length would deliver a similar reduction in hallucinations. Indeed, hallucinations and length are expected to correlate, and shorter length should result in fewer hallucinations. However, as pointed out in Sec. 4, drastically reducing hallucinations may be possible without any control mechanism and can be achieved, at least on WikiBio, with templates. The main challenge lies in doing so without a big drop in informativeness, that is, in coverage of input fields. Comparing the outputs of halLM with those of halWO, and both with those of Tian et al. (2019), we note that the ranking in terms of average sentence length (in sentencepiece tokens) coincides with the ranking in terms of coverage (Table 2): 17.2, 17.8, 18.7. While halWO may associate the special hal_0 token with the shortest 20% of the training data, for halLM this token is apparently associated with a different selection of 20% of the data points.

6 Conclusions

We presented a simple but powerful idea of controlling hallucinations which are caused by the noise in the training data and proposed two ways of detecting such noise. We demonstrated that it is possible to reduce the amount of hallucinations at no coverage cost by informing the model about how noisy every source-target example is and without changing the model architecture. Importantly, this was done without making any assumptions about the data. In an evaluation with humans we showed that the faithfulness of generated sentences can be significantly improved at no loss in fluency or coverage. The results we reported on the noisy WikiBio dataset improve upon the prior work.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In Proceedings of the 3rd International Conference on Learning Representations.

Bhuwan Dhingra, Manaal Faruqui, Ankur Parikh, Ming-Wei Chang, Dipanjan Das, and William Cohen. 2019. Handling divergent reference texts when evaluating table-to-text generation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics.

Ondřej Dušek, David M. Howcroft, and Verena Rieser. 2019. Semantic noise matters for neural natural lan-
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.

Ye Zhang, Nan Ding, and Radu Soricut. 2018. SHAPED: Shared-private encoder-decoder for text style adaptation. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1528–1538.