HALLUCINATIONS IN NEURAL MACHINE TRANSLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural machine translation (NMT) systems have reached state of the art performance in translating text and are in wide deployment. Yet little is understood about how these systems function or break. Here we show that NMT systems are susceptible to producing highly pathological translations that are completely untethered from the source material, which we term *hallucinations*. Such pathological translations are problematic because they are are deeply disturbing of user trust and easy to find with a simple search. We describe a method to generate hallucinations and show that many common variations of the NMT architecture are susceptible to them. We study a variety of approaches to reduce the frequency of hallucinations, including data augmentation, dynamical systems and regularization techniques, showing that data augmentation significantly reduces hallucination frequency. Finally, we analyze networks that produce hallucinations and show that there are signatures in the attention matrix as well as in the stability measures of the decoder.

1 INTRODUCTION

Neural machine translation (NMT) systems are language translation systems based on deep learning architectures (Cho et al., 2014a; Bahdanau et al., 2014; Sutskever et al., 2014). In the past few years NMT has vastly improved and has been deployed in production systems, for example at Google (Wu et al., 2016), Facebook (Gehring et al., 2017), Microsoft (Hassan et al., 2018) and many others. As NMT systems are built on deep learning methodology, they exhibit both the strengths and weaknesses of the approach. For example, NMT systems are competitive with state of the art performance (Bojar et al., 2017) and scale well to very large datasets (Ott et al., 2018) but like most large deep learning systems, NMT systems are poorly understood. For example, entering repeated words many times occasionally results in strange translations in many commercial translation systems, a phenomenon which has been highly publicized (Christian, 2018). More broadly, recent work shows that NMT systems are highly sensitive to noise in the input tokens (Belinkov & Bisk, 2017) and are also susceptible to adversarial inputs (Cheng et al., 2018). When there is an error in translation, it can be challenging to either understand why the mistake occurred or engineer a fix.

Here we continue the study of noise in the input sequence and describe a type of phenomenon that is particularly pernicious, whereby a single additional input token inserted into the source sequence can completely divorce the translation from the input sentence. For example, here is a German input sentence translated to English (reference) by a small NMT system:

Source: `Caldiero sprach mit E! Nachrichten nach dem hart erkämpften Sieg, noch immer unter dem Schock über den Gewinn des Großen Preises von 1 Million $.`
Reference: `Caldiero spoke with E! News after the hard-fought victory, still in shock about winning the $1 million grand prize.`
NMT Translation: `Caldiero spoke with E, after the hard won victory, still under the shock of the winning of the Grand Prix of 1 million $.`

Simply by adding a single input token (*mit:with*, *werden:to become* and *dass:that*) to the beginning of the input sentence and translating them with the same NMT model yield the following:

Mistranslations:
mit: `It was said to have a lot of fun in the world.`
werden: `Don't hesitate to contact us, if you want to be able to pay for you.`
dass: `I'm looking forward to having a lot of money.`

These mistranslations are completely semantically incorrect and also grammatically viable. They are untethered from the input so we name them *'hallucinations'*. Clearly, even if hallucinations occur only occasionally, the NMT model will be problematic from the perspective of user trust, potentially leading the user to a false sense of confidence in a very incorrect translation. In this work, we show that hallucinations are widespread in the popular NMT system we study. For example, 73% of sentences in our development set can be perturbed to hallucination in our simplified canonical model. If we decode with beam search that number drops, but remains high at 48%.

We systematically explore hallucinations in a variety of NMT variants. We also develop methodologies to help ameliorate this problem, using ideas from data augmentation, dynamical systems theory, and regularization. Finally, we analyze an NMT model that demonstrates hallucinations and show that there are signatures of hallucinations that can be seen in the attention matrix. We focus on NMT systems built using RNNs (Wu et al., 2016), on which many commercial translation systems are based. As RNNs recursively generate translations, we examine the dynamical nature of RNNs. Finally, we study decoder stability when hallucinations are produced and when they are not.

2 Related Work

Since its invention, researchers have been working to better understand NMT. For example, moving from the original Seq2Seq model Sutskever et al. (2014); Cho et al. (2014b) to models that utilize attention mechanisms (Bahdanau et al., 2014), resulted in improved translation quality (Luong et al., 2015) and better interpretability (Ding et al., 2017). Studies identified the most critical components of the LSTM (Greff et al., 2017) and the role of the LSTM in language modeling (Karpathy et al., 2015) more broadly. Followed by explorations in interpretability, recent work has focused on robust NMT, studying the effects of input noise, aiming to reduce variation from typos (Belinkov & Bisk, 2017) and synonym choice (Cheng et al., 2018) in the discrete input set used by NMT systems. Both (Belinkov & Bisk, 2017) and (Cheng et al., 2018) have discovered that NMT systems are highly sensitive to input noise and both used adversarial training to help stabilize NMT systems (either with black-box adversarial training or augmenting an adversarial loss). There has been work in understanding how to handle some of the pathologies in RNNs for language modeling and translation, for example, using scheduled sampling to handle mismatches between training and test sets (Bengio et al., 2015).

In parallel, there has also been work in understanding RNNs through the framework of dynamical systems (Sussillo & Barak, 2013; Sussillo, 2014; Laurent & von Brecht, 2016; Rivkind & Barak, 2017; Beer & Barak, 2018) and understanding their capacity in language tasks (Collins et al., 2017). For example, it is known that continuous time vanilla RNNs exhibit high-dimensional chaos (Sompolinsky et al., 1988), which can be driven out by strong input (Rajan et al., 2010). RNNs exhibiting chaos can be beneficial, for example, RNNs capable of chaos in the absence of input can serve as strong initialization for training RNNs when they are input driven (Sussillo & Abbott, 2009), but caution must be used as unbridled chaos can be viewed as a source of dynamical noise. This has led to efforts to rid RNNs of chaos altogether (Laurent & von Brecht, 2016). Efforts related to improving optimization may also have dynamically regularizing effects, e.g. (Arjovsky et al., 2015). Given the complexities and slow compute times of recurrent systems, there have also been attempts to rid NMT of recurrence (Kalchbrenner et al., 2016; Gehring et al., 2017; Vaswani et al., 2017).

We further expand on these studies by highlighting the specific pathology of hallucinations, systematically studying those hallucinations, and analyzing them from a dynamical systems perspective.

3 NMT Methods

Models: In this paper, we use a standard RNN-based encoder-decoder NMT model with attention. Specifically, we study the NMT model described in (Wu et al., 2016), known as GNMT. We use the GNMT model and its public implementation[1]. Formally, given an input sequence, $\mathbf{x}_{1:S}$ of length S, the NMT model first encodes the input sequence $\mathbf{x}_{1:S}$ into a set of vectors $\mathbf{z}_{1:S} = f_{enc}(\mathbf{x}_{1:S})$ using its encoder f_{enc}. The task of the decoder f_{dec}, is to generate the translation, $\mathbf{y}_{1:T}$, one symbol at a time y_i, given the encoding, $\mathbf{z}_{1:S}$, and previously generated symbols $y_{<i}$. The decoder, f_{dec} is implemented as a conditional sequence model

[1] https://github.com/tensorflow/nmt with *gnmt-v2* architecture.

(Bahdanau et al., 2014), where the distribution over $\mathbf{y}_{1:T}$ is conditioned on $\mathbf{x}_{1:S}$. The decoder internally makes use of an attention mechanism f_{att} to query the encoder, summarizing $\mathbf{z}_{1:S}$ for each output symbol y_i, putting it all together $y_i = f_{dec}(y_{<i}, f_{att}(\mathbf{z}_{1:S}))$ (also see Figure 6 for a detailed model schematic). Finally, the conditional probability of the target sequence is modelled as $p(\mathbf{y}_{1:T}|\mathbf{x}_{1:S}) = \prod_{i=1}^{T} p(y_i|y_{<i}, \mathbf{x}_{1:S})$ and the log of this conditional likelihood is maximized given a set of source-target pairs $(\mathbf{x}, \mathbf{y})$ during training.

We study models that are significantly smaller than those typically used in state-of-the-art or production systems for the sake of research tractability. We use a single layer bidirectional LSTM in the encoder f_{enc} and a two layers unidirectional LSTM in the decoder f_{dec} with an additive attention mechanism as f_{att} (Britz et al., 2017). The word embedding dimensions and each LSTM hidden cell (both in the encoder and decoder) are set to 256. We refer to this model as the *canonical* model. Unless otherwise stated, we used Adam (Kingma & Ba, 2014) optimizer with a learning rate of 0.001, a constant learning rate schedule, and clipped gradients to a maximum norm of 5.0 during training.

Given these hyper-parameter and architectural choices, we trained 10 canonical models with different random seeds to observe how parameter initialization variability played a role in our results. All additional model variants we describe later were also trained 10 times with the same 10 different random seeds. Each model was trained for 1M steps (updates) with a mini-batch of size 128 and the training checkpoint with the best BLEU score on the development set was selected.

The central goal of our study was to understand how various modelling choices affected the frequency of hallucinations. In order to isolate the effects of modeling changes, all model variants we study in this paper were identical to the canonical model except for a single change. This means, for example, that our model with 512 hidden units also is 2 layers deep, etc. We performed a simple hyper-parameter search for the canonical model, and did not perform additional hyper-parameter searches for any additional models. All models we present are well trained with a BLEU score of at least 20.0 on the test set, a reasonable score for 2-layer models with 256 hidden units.

Inference: Generating a translation of the input sequence, or formally finding an output sequence that maximizes the conditional log-probability, $\hat{\mathbf{y}} = \text{argmax}_{\mathbf{y}} \log p(\mathbf{y}|\mathbf{x})$, is a major challenge in NMT since the exact inference (or decoding) is intractable. NMT uses approximate decoding techniques which we also have used in this paper. The simplest approximate decoding technique, *greedy decoding*, chooses the most-likely symbol under the conditional probability $\hat{y}_t = \text{argmax}_i \log p(y_t = i|\hat{y}_{<i}, \mathbf{x}_{1:S})$, outputting a single best local prediction by keeping track of a single hypothesis k, at each time step. Another approximate decoding technique, *beam search*, improves upon greedy decoding by keeping track of multiple hypotheses (*beams*), where $k > 1$ at each time step of the decoding, compared to k=1 in greedy-decoding. To maintain simplicity in our canonical model we used greedy decoding. Note that production systems will often perform beam search to find a more probable translation than one generated by greedy search. We also ran an additional set of experiments with beam search.

Data: We trained all models with the German to English WMT De→En 2016 dataset (4,500,966 examples) (Bojar et al., 2016), validated with the WMT De→En 2015 development set (2,169 examples). We then used the WMT De→En 2016 test set (2,999 examples) to compute the hallucination percentage for each model.

For the input and output of all NMT models in consideration, we used sub-word tokens extracted by Byte-Pair Encoding (BPE) (Sennrich et al., 2015). To construct a vocabulary of unique tokens, we first combined the tokenized source and target corpora[2], and then learned a joint BPE code with an 8k merge operations budget, resulting in 12,564 unique tokens. Further, in order to study the effect of larger vocabulary sizes, for some experiments we repeated the same process with 16,000 and 32,000 BPE codes and ended up with vocabularies having 19,708 and 36,548 unique tokens respectively. Note that we used the same vocabulary for both source and target side languages.

[2]We used Moses tokenizer: https://github.com/moses-smt/mosesdecoder/blob/master/scripts/tokenizer/tokenizer.perl

Algorithm 1: Computing the percentage of hallucinations in a NMT model

```
Select a model; Fix a random seed;
Select a group of subword tokens with the following attributes:;
- Common tokens: 100 most common subword tokens;
- Mid-frequency tokens: random sample of 100 subword tokens between common and rare tokens;
- Rare tokens: 100 least common subword tokens;
- Punctuation: all punctuation tokens;
for every sentence in test corpus (e.g. WMT De→En 2016 test set) do
    if adjusted BLEU between reference sentence and translated sentence > 0.09 then
        for every selected token do
            for every location in (beginning, end, second-to-end, randomly in the middle) do
                put the selected token at the selected location in the byte-pair encoded input sequence;
                translate the perturbed input sequence;
                if adjusted BLEU between the translated, perturbed sentence and the translated,
                  unperturbed sentence< 0.01 then
                    this sentence can be perturbed to produce a hallucination;
```

4 HALLUCINATIONS

Informally, a hallucination is a translation of a perturbed input sentence that has almost no words in common with the translation of the unperturbed sentence. Here, we use the term *perturb* to mean adding a single token to the input source sequence. This is based on an initial observation that adding a rare token to an input sequence reliably caused a model to generate a hallucination, e.g. adding a Chinese character token to a German to English translation. We expanded and systematized this discovery into a brute force search procedure (Algorithm 1) by splitting our tokens into several types: common (100 most common German tokens), rare (100 least common German tokens), mid-frequency tokens (randomly sampled 100 tokens from the remaining German tokens), and punctuation tokens. Additionally, we attempted to perturb each sentence by inserting a token at one of several positions: beginning, end, second to the end, and randomly in the middle. We did this for every sentence in our test set and collected statistics for each model variant.

To define a quantitative threshold for a hallucination we modified the BLEU score, which is used to compare a reference sequence with a translated sequence. Briefly, the BLEU score is a common metric for translation, which measures weighted n-gram overlaps while penalizing short translations. We modified the BLEU score by re-weighting the n-grams in the BLEU score computation to favor having any words in common between the two sentences (1.0 for one-grams and 0.8 for bi-grams and disregarded other n-grams). Then, we call only sentences that have an adjusted BLEU score of less than 0.01 hallucinations. For examples of sentence pairs with different adjusted BLEU scores, see Section 8.2 in the Appendix.

Not all translations are good even before adding a perturbation token to the input sequence. To strengthen our results on hallucinations, we first excluded these poor translations by computing the adjusted BLEU score between the reference translation and the translation produced by the unperturbed input sequence. We kept sentences that had an adjusted BLEU of ≥ 0.09. We chose a value of 0.09 because it seemed to maintain enough context that you could tell the translation and the reference were related.

5 HALLUCINATION FREQUENCY AND REDUCTION

5.1 EXAMPLES

We describe four common hallucination patterns: (1) grammatically correct output that bears no relation to the input text, (2) ungrammatical output with oscillatory structure, (3) output that remains largely in the source language, and finally (4) terse jumps to the end of the sequence. We also observe translations that are ungrammatical nonsense. While still highly undesirable, we note that a user should at least be able spot and reject these additional hallucination patterns. See the Appendix section 8.3 for more examples.

Source: Freundschaft schließen durch Backen.
Reference: Make friends through baking.
Perturbation: Added *ich:I* randomly in sentence.
Grammatically correct hallucination:Should you want to join us?

Source: Monatelang war hinter verschlossenen Türen verhandelt, gegrübelt, debattiert und gezeichnet worden.
Reference: Plans have been negotiated, mulled over, debated, and plotted behind closed doors for months.
Perturbation: Added *uns:we* at the beginning of the sentence.
Oscillatory hallucination: In the month of the month of the month of the month of the month of the month, it was a matter of course.

Source: Neue Verhandlungen mit den Piloten
Reference: New negotiations with pilots
Perturbation: Added *mit:with* randomly in sentence.
Source language hallucination: Neuist e mehr Jahren mit der Piloten d

Source: Für die Fed-Repräsentanten beeinflussen die Marktturbulenzen die komplexe Kalkulation , wann man die Zinsen erhöhen solle.
Reference: For Fed policymakers, the market turmoil adds to the complex calculus of when to raise the interest rate.
Perturbation: Add *uns:we* randomly in sentence.
End of sequence hallucination: For FF.C.

5.2 Frequency of Hallucinations

We show that hallucinations can be easily evoked by inserting tokens in the source sequence. We used Algorithm 1 to quantify how susceptible a given model is to hallucination. In particular, we studied what types of perturbations (location, and token type) are more effective at inducing hallucinations. With this method, we found that, on average, 73% of all sentences in the WMT De→En test set can be perturbed to hallucination in the canonical model.

We studied how beam search, number of hidden units, vocabulary size, and decoding scheme affected hallucination percentages (Figure 1, left). We found that changing the number of hidden units to both 512 and 1024 from 256 and changing the vocabulary size–from 8K to 16K BPE codes did not significantly decrease the hallucination percentage. However, beam search and a vocabulary size increase corresponding to 32K BPE codes did significantly lower the mean percentage of hallucinations. We also studied how different types of perturbations impacted the hallucination percentage of the canonical model (Figure 1, right). By far, adding a perturbing token to the beginning of the input sequence induces the most hallucinations in the canonical model.

5.3 Reducing hallucination frequency

What changes can we make to the model to make it more robust to hallucinations? We investigated the effect of three different methodologies, (1) simple regularizations, (2) data augmentation and (3) regularizations on the dynamics in state space. We tested if a model variation significantly reduces hallucinations by performing a one-sided Mann–Whitney U between the canonical distribution of models and the distribution of models that use the model variant. We use a p-value of 0.05.

Simple Regularizations: We choose dropout, L2 regularization on embeddings (L2E) and L2 regularization on all weights (L2) as straight-forward regularization techniques to be applied. For dropout, we created a model with dropout in all feed-forward layers, with a keep probability of 0.9. Next, we implemented L2 regularization on the token-embeddings and throughout the entire model with weighting hyperparameters of 1×10^{-4} and 1×10^{-5}, respectively.

Data augmentation (DA): We augmented the training data by perturbing all training sentences with a random token (either common, rare, mid-frequency, or punctuation) fat either the beginning, end, second-to-end, or randomly in the middle while keeping the reference translation the same. This doubled our training set.

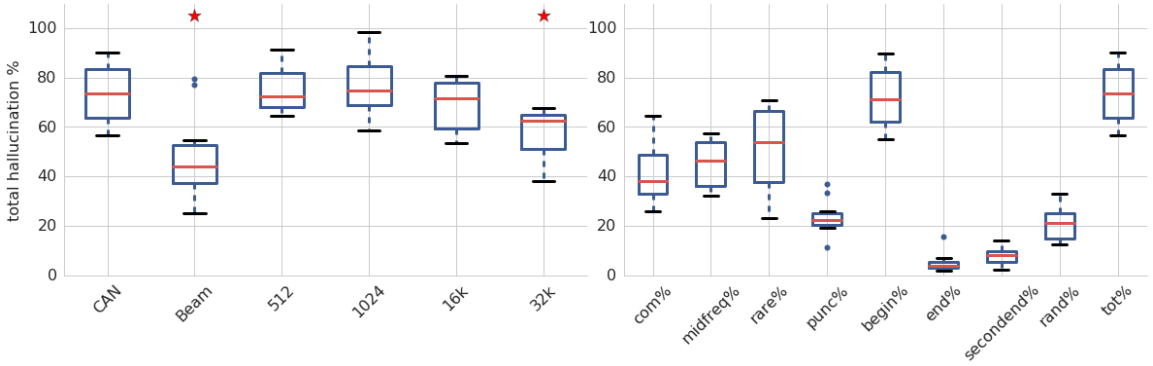


Figure 1: Hallucination percentages. (**left**) Percentage of all test set sentences that can be perturbed to hallucination (using any type of hallucination token at any location) as a function of model variant: canonical, beam search decoding, number of hidden dimensions (512 and 1024), and vocabulary size (16K and 32K BPE codes). Red stars indicate model variant hallucination % is statistically lower than the hallucination % in the canonical model. (**right**) Hallucination percentages as a function of perturbation token type and position inserted in the input sentence over all canonical model random seeds. Any particular statistic, e.g. common% is computed over all perturbation locations, likewise begin% averages over all perturbing token types.

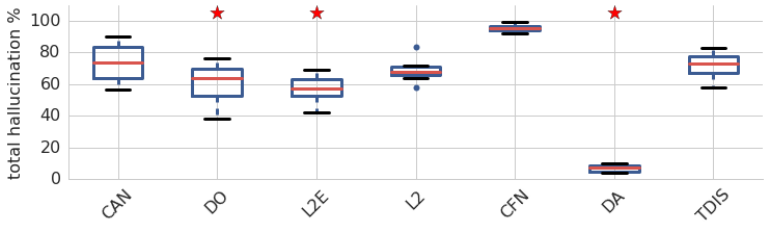


Figure 2: Hallucination percentages for model variants: dropout, L2E, L2, CFN, DA, and tied decoder initial state (TDIS). The red stars denote a statistically significant difference ($p < 0.05$) between the hallucination % of the model variant and the canonical model.

We then trained a canonical model with the augmented training set, and found that data augmentation helps decrease hallucination percentages. We call this model DA.

Dynamical Regularizations: We wondered if hallucinations could be reduced by providing a more exact initial state for the decoder, so we trained additional models where the initial state of the decoder was tied to last step of the encoder (TDIS). Note that the canonical model sets the decoder initial state as a vector of zeros. As a second regularization method that operates on the state space, we used Chaos-free network (CFN) (Laurent & von Brecht, 2016) which by premise cannot produce chaos. We replaced the LSTM cell with the CFN in a set of experiments, again using 256 hidden units.

Dropout, L2E, and DA all resulted in statistically significant decreases in hallucination percentage, with DA being by far the most effective at decreasing hallucinations. On the contrary, switching out LSTM cells for CFN cells resulted in a significant increase in the hallucination percentage. Finally, linking the initial state of the decoder with the final state of the encoder had no statistical effect on the hallucination percentage.

Although data augmentation dramatically reduced hallucinations in the canonical model, it requires knowing the kind of perturbations that one would use to induce a hallucination. To study how fine grained one's knowledge must be, we trained the canonical model on a different training set where we withheld two types of data augmentation: perturbing at the beginning or with common tokens (We call this model DA w/o beginning or common). We then compared this model with the canonical model trained with the full DA training set (Figure 3). We found that DA w/o beginning or common yields much higher hallucination percentages when tested by perturbing at the beginning or with common tokens in comparison to the DA model. However, we also saw a reduction in hallucination percentage for common and beginning tokens when compared to the canonical model. This indicates that DA can still provide some protection against hallucinations, even if exact perturbations are not known.[3]

[3] As of the ICLR submission 7 of 140 simulations had not finished. We do not expect the results to change upon incorporation of that data.

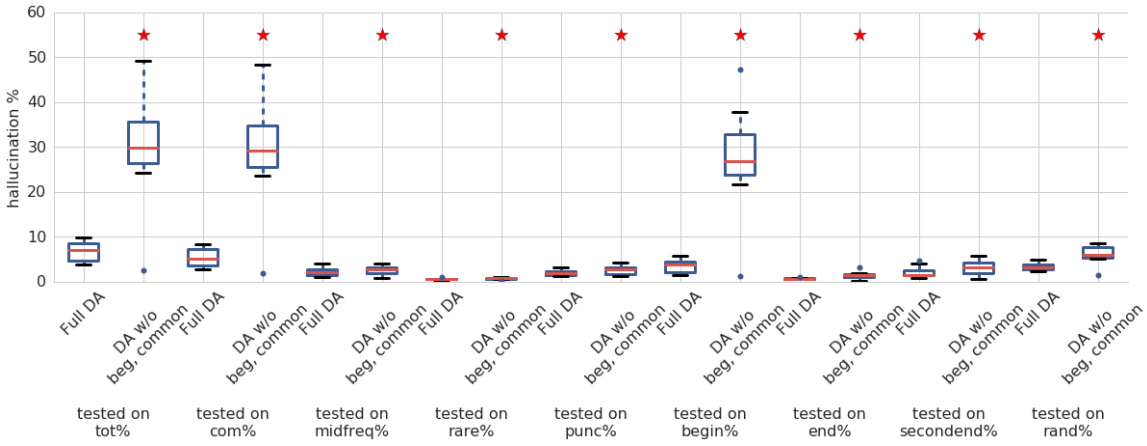


Figure 3: Effects of data augmented (DA) training when including all perturbation types vs excluding common and beginning perturbation types. We trained two models, one including all perturbations types for DA training, and the other excluding common and beginning perturbation types. We then examined the hallucination percentage of each perturbation type for both of these models and studied whether a DA model would be less prone to hallucinate when perturbed with types of tokens or positions it had not been trained against. Red star shows that DA w/o beginning or common had statistically significantly reduced mean compared to the canonical model trained without DA.

6 Analysis of Hallucinations

6.1 Attention Matrices

We observed a large difference between attention matrices of normal translations and of hallucinations. Attention networks in normal translations tend to study the entire input sequence throughout decoding. In French to English and other language pairs that are grammatically aligned (German to English is somewhat aligned), this often results a strong diagonal in the attention matrix. The attention matrix when translating hallucinations, however, shows the model attending to a few tokens. We give an example comparison in Figure 4, top panels. For additional attention matrices, see Section 9.

We wanted to quantify this difference in a way that does not require strong alignment between languages, i.e. one expects English to French to result in a largely diagonal matrix but not English to Turkish, so we used information entropy to compute a statistic that described the difference between attention matrices during a decode that resulted in a hallucination and those that resulted in a normal translation. Specifically, at each output of the decoder, the attention network gives a distribution over input tokens. We averaged these distributions across all decoded output tokens, resulting in a distribution of average attention weight over the input tokens. We treated this as a discrete distribution and computed the entropy, $-\sum_t p(x_t) \log p(x_t)$, where x_t is the input token at time t, for each example, resulting in a distribution of entropy values over all decoded sequences.

We then compared the entropy of average attention distributions between hallucinations and correct translations (Figure 4, bottom panels). This figure shows a significant difference between the entropy values for hallucination sequences. As a control, we show there is no significant difference between original input sequences and perturbed input sequences for sentences that cannot be perturbed to hallucination. Note that, in real world scenarios where a ground truth comparison between a reference translation and model translation is not possible, the measure of entropy of the average attention distribution may be useful to detect hallucinations.

The breakdown of the attention module seems to signal that the encoder and the decoder have been decoupled, and the decoder ignores context from the encoder and samples from its language model. Two possibilities are that broken attention modules are the root cause of decoupling, or they are a symptom of further breakdown in the dynamics of the decoder.

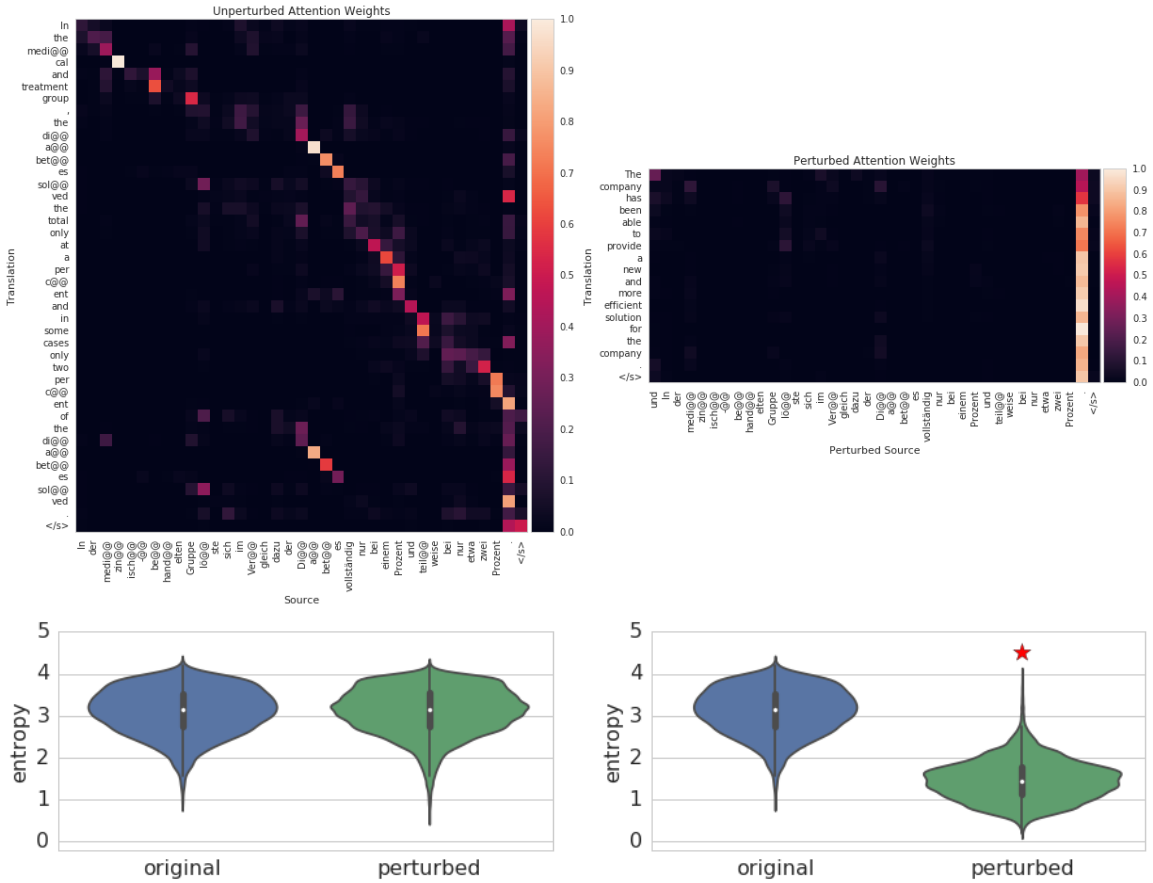


Figure 4: The attention matrix can reveal that a hallucination has occurred. **(upper left)** Normal translations typically produce attention matrices that attend to most tokens in the input source sequence as the decoder produces the translation (x-axis source sequence, y-axis decoded sequence). **(upper right)** The attention network tends to focus on just a few input tokens during hallucinations. **(bottom)** We used an entropy measure to quantify how distributed the attention is over the input source sequence. Shown are the distributions for normal translations **(bottom left)** and hallucinations **(bottom right)** for the original (blue) and perturbed (green) attention entropy values. Mean of the entropy distributions for hallucinations are statistically significantly different (Mann–Whitney U test, $p < 0.05$).

6.2 Stability analysis of the decoder

Examination of the types of hallucinatory text, oscillations and instability, led us to believe that hallucinations result from a dynamical process gone wrong in the decoder (whose cause may yet be further upstream). Further, many of the remedies to hallucinations can also be viewed as being dynamically regularizing. To study this idea more concretely, we defined a spectrum of stability exponents for the decoder and compared them between normal translations and hallucinations (Figure 5).

Concretely, we studied the stability of the decoder as a function of a given input token sequence, $\mathbf{x}_{1:S}$ of length S. The sequence $\mathbf{x}_{1:S}$ is run through the encoder, whose output is processed by the attention network, finally delivering an input to the decoder. For a given input token sequence, the decoder runs until it produces an end-of-sequence token, resulting in an output token sequence $\mathbf{y}_{1:T}$ of length T (or reaches a maximal decoded sequence length $T > 3S$). We were interested in studying $\frac{\partial h_T}{\partial h_0} = \frac{\partial h_T}{\partial h_{T-1}} \cdots \frac{\partial h_1}{\partial h_0}$ as many stability properties can be deduced from it. We note that if one is interested in studying $\frac{\partial h_t}{\partial x_s}$, the iterative process described by $\frac{\partial h_T}{\partial h_0}$ would still be critical to understand due to the chain rule.

We defined our spectrum of stability exponents, in analogy with Lyapunov exponents, but adapted for finite time by studying a finite-time version of Oseledets matrix, typically used in the study of chaotic dynamical

systems. In particular, the ith stability exponent is defined as $\lambda_i(\mathbf{x}_{1:S}) = \frac{1}{2T}\log\left(\alpha_i(\mathbf{x}_{1:S})\right)$, where $\alpha_i(\mathbf{x}_{1:S})$ is the ith eigenvalue of the positive-semidefinite symmetric matrix $\left(\frac{\partial h_T}{\partial h_0}(\mathbf{x}_{1:S})\right)^T\left(\frac{\partial h_T}{\partial h_0}(\mathbf{x}_{1:S})\right)$ and $h(t)$ is the decoder state at time t concatenated across all layers. We used auto-differentiation software to exactly compute the Jacobian $\frac{\partial h_T}{\partial h_0}(\mathbf{x}_{1:S})$ so the complexities of the decoder circuitry were handled naturally (shown in Appendix, Section 8.1[4].

We show the distribution of stability exponents, comparing between all input sequences that could be made to hallucinate, and all those that could not (Figure 5). We show these for both the canonical model and the model trained with data augmentation. There are two observations. First, means of the distribution of the stability exponents for the canonical model, averaged over those sentences that could be perturbed to hallucinate, are statistically different than exponents averaged over sentences that could not be perturbed to hallucinate. Second, the distributions of the model trained with data augmentation show significantly reduced exponents in comparison to the canonical model.

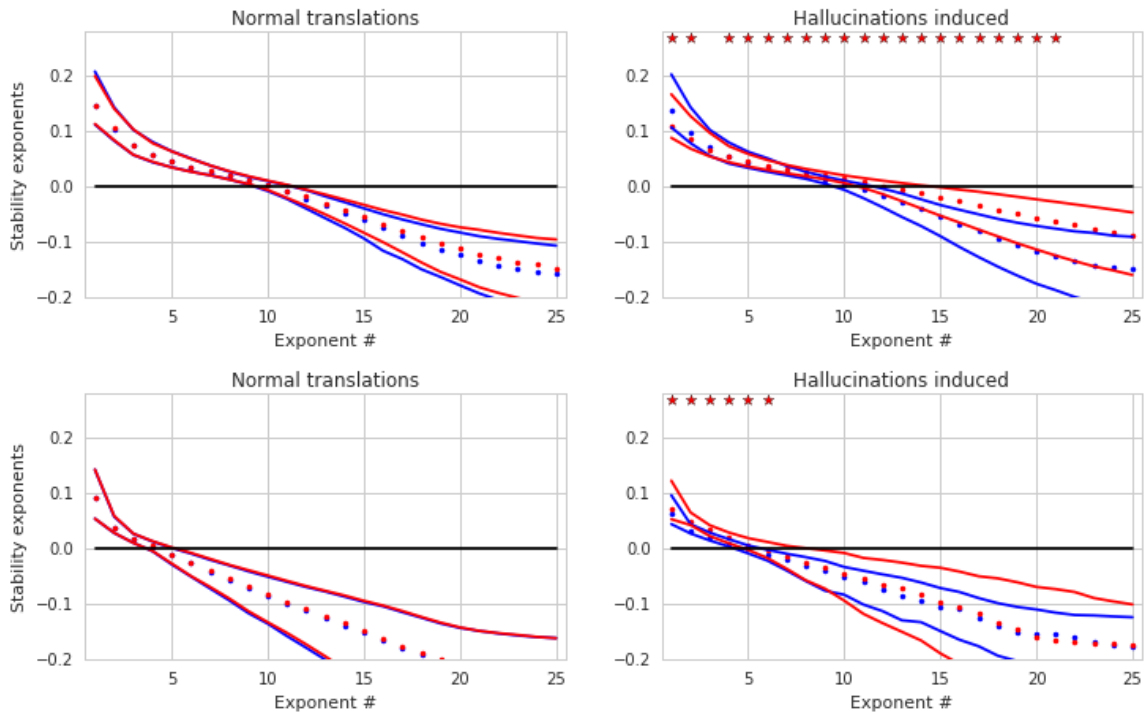


Figure 5: Stability analysis of hallucinations vs. normal translations. **(top left)** Distribution of stability exponents for canonical model for unperturbed (blue) and perturbed, no hallucination (red) input sequences that resulted in a normal translation (Median (dots) and 25% and 75% quartiles (solid lines) shown). **(top right)** Same, but translations that could be perturbed to hallucinate (blue - unperturbed, red - hallucination). Red stars denote statistical significance (U test, $p < 0.001$) in the difference in stability exponents between perturbed and unperturbed compared between normal translations and hallucinations. **(bottom left, right)** Same as top, except for model trained with data augmentation.

[4]Methodologically, we note that it is typical to compute Lyapunov exponents using the algorithm of (Benettin et al., 1980). This is because studying the long-time behavior of dynamical systems typically requires many thousands of system iterations. We studied the numerical stability of our direct approach to computing our stability exponents using random chaotic RNNs (Sompolinsky et al., 1988) and found that the direct approach was more than adequate for short sequences and reasonable condition numbers, so we stayed with the simpler, direct approach.

7 DISCUSSION

In this paper we uncovered and studied a hallucination-like phenomenon whereby adding a single additional token into the input sequence causes complete mistranslation. We showed that hallucinations are common in the NMT architecture we examined, as well as in its variants. We note that hallucinations appear to be model specific. We showed that the attention matrices associated with hallucinations were statistically different on average than those associated with input sentences that could not be perturbed. Finally we proposed a few methods to reduce the occurrence of hallucinations.

Our model has two differences from production systems. For practical reasons we studied a small model and used a limited amount of training data. Given these differences it is likely that our model shows more hallucinations than a quality production model. However, given news reports of strange translations in popular public translation systems (Christian, 2018), the dynamical nature of the phenomenon, the fact that input datasets are noisy and finite, and that our most effective technique for preventing hallucinations is a data augmentation technique that requires knowledge of hallucinations, it would be surprising to discover that hallucinations did not occur in production systems.

While it is not entirely clear what should happen when a perturbing input token is added to an input source sequence, it seems clear that having an utterly incorrect translation is not desirable. This phenomenon appeared to us like a dynamical problem. Here are two speculative hypotheses: perhaps a small problem in the decoder is amplified via iteration into a much larger problem. Alternatively, perhaps the perturbing token places the decoder state in a poorly trained part of state space, the dynamics jump around wildly for while until an essentially random well-trodden stable trajectory is found, producing the remaining intelligible sentence fragment.

Many of our results can be interpreted from the vantage of dynamical systems as well. For example, we note that the NMT networks using CFN recurrent modules were highly susceptible to perturbations in our experiments. This result highlights the difficulty of understanding or fixing problems in recurrent networks. Because the CFN is embedded in a larger graph that contains an auto-regressive loop, there is no guarantee that the chaos-free property of the CFN will transfer to the larger graph. The techniques we used to reduce hallucinations can also be interpreted as dynamical regularization. For example, L2 weight decay is often discussed in the context of generalization. However, for RNNs L2 regularization can also be thought of as dynamically conditioning a network to be more stable. L2 regularization of input embeddings likely means that rare tokens will have optimization pressure to reduce the norm of those embeddings. Thus, when rare tokens are inserted into an input token sequence, the effects may be reduced. Even the data augmentation technique appears to have stability effects, as Figure 5 shows the overall stability exponents are reduced when data augmentation is used.

Given our experimental results, do we have any recommendations for those that engineer and maintain production NMT systems? Our results on reducing hallucinations suggest that standard regularization techniques such as Dropout and L2 weight decay on the embeddings are important. Further, data augmentation seems critical and we recommend inserting randomly chosen perturbative tokens in the input sentence as a part of the standard training regime (while monitoring that the BLEU score does not fall). We note a downside of data augmentation is that, to some extent, it requires knowing the types of the pathological phenomenon one desires to train against.

REFERENCES

Martín Arjovsky, Amar Shah, and Yoshua Bengio. Unitary evolution recurrent neural networks. *CoRR*, abs/1511.06464, 2015. URL http://arxiv.org/abs/1511.06464.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

C. Beer and O. Barak. Dynamics of dynamics: following the formation of a line attractor. *ArXiv e-prints*, May 2018.

Yonatan Belinkov and Yonatan Bisk. Synthetic and natural noise both break neural machine translation. *arXiv preprint arXiv:1711.02173*, 2017.

Giancarlo Benettin, Luigi Galgani, Antonio Giorgilli, and Jean-Marie Strelcyn. Lyapunov characteristic exponents for smooth dynamical systems and for hamiltonian systems; a method for computing all of them. part 1: Theory. *Meccanica*, 15(1):9–20, 1980.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. In *Advances in Neural Information Processing Systems*, pp. 1171–1179, 2015.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Shujian Huang, Matthias Huck, Philipp Koehn, Qun Liu, Varvara Logacheva, et al. Findings of the 2017 conference on machine translation (wmt17). In *Proceedings of the Second Conference on Machine Translation*, pp. 169–214, 2017.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, pp. 131–198, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://www.aclweb.org/anthology/W/W16/W16-2301.

Denny Britz, Anna Goldie, Minh-Thang Luong, and Quoc Le. Massive exploration of neural machine translation architectures. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 1442–1451. Association for Computational Linguistics, 2017. URL http://aclweb.org/anthology/D17-1151.

Yong Cheng, Zhaopeng Tu, Fandong Meng, Junjie Zhai, and Yang Liu. Towards robust neural machine translation. *arXiv preprint arXiv:1805.06130*, 2018.

Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014a.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734. Association for Computational Linguistics, 2014b. doi: 10.3115/v1/D14-1179. URL http://www.aclweb.org/anthology/D14-1179.

Jon Christian. Why is google translate spitting out sinister religious prophecies?, July 2018. URL https://motherboard.vice.com/en_us/article/j5npeg/why-is-google-translate-spitting-out-sinister-religious-prophecies. [Online; posted 20-July-2018].

Jasmine Collins, Jascha Sohl-Dickstein, and David Sussillo. Capacity and trainability in recurrent neural networks. *stat*, 1050:28, 2017.

Yanzhuo Ding, Yang Liu, Huanbo Luan, and Maosong Sun. Visualizing and understanding neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1150–1159. Association for Computational Linguistics, 2017. doi: 10.18653/v1/P17-1106. URL http://www.aclweb.org/anthology/P17-1106.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. Convolutional sequence to sequence learning. *CoRR*, abs/1705.03122, 2017. URL http://arxiv.org/abs/1705.03122.

Klaus Greff, Rupesh K Srivastava, Jan Koutník, Bas R Steunebrink, and Jürgen Schmidhuber. Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, 28(10):2222–2232, 2017.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, Shujie Liu, Tie-Yan Liu, Renqian Luo, Arul Menezes, Tao Qin, Frank Seide, Xu Tan, Fei Tian, Lijun Wu, Shuangzhi Wu, Yingce Xia, Dongdong Zhang, Zhirui Zhang, and Ming Zhou. Achieving human parity on automatic chinese to english news translation. *CoRR*, abs/1803.05567, 2018. URL http://arxiv.org/abs/1803.05567.

Nal Kalchbrenner, Lasse Espeholt, Karen Simonyan, Aäron van den Oord, Alex Graves, and Koray Kavukcuoglu. Neural machine translation in linear time. *CoRR*, abs/1610.10099, 2016. URL http://arxiv.org/abs/1610.10099.

Andrej Karpathy, Justin Johnson, and Li Fei-Fei. Visualizing and understanding recurrent networks. *arXiv preprint arXiv:1506.02078*, 2015.

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

Thomas Laurent and James von Brecht. A recurrent neural network without chaos. *arXiv preprint arXiv:1612.06212*, 2016.

Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1412–1421. Association for Computational Linguistics, 2015. doi: 10.18653/v1/D15-1166. URL http://www.aclweb.org/anthology/D15-1166.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. Scaling neural machine translation. *arXiv preprint arXiv:1806.00187*, 2018.

Kanaka Rajan, LF Abbott, and Haim Sompolinsky. Stimulus-dependent suppression of chaos in recurrent neural networks. *Physical Review E*, 82(1):011903, 2010.

Alexander Rivkind and Omri Barak. Local dynamics in trained recurrent neural networks. *Physical review letters*, 118(25):258101, 2017.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *CoRR*, abs/1508.07909, 2015. URL http://arxiv.org/abs/1508.07909.

Haim Sompolinsky, Andrea Crisanti, and Hans-Jurgen Sommers. Chaos in random neural networks. *Physical review letters*, 61(3):259, 1988.

David Sussillo. Neural circuits as computational dynamical systems. *Current opinion in neurobiology*, 25: 156–163, 2014.

David Sussillo and Larry F Abbott. Generating coherent patterns of activity from chaotic neural networks. *Neuron*, 63(4):544–557, 2009.

David Sussillo and Omri Barak. Opening the black box: low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural computation*, 25(3):626–649, 2013.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112, 2014.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, pp. 6000–6010, 2017.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *CoRR*, abs/1609.08144, 2016. URL http://arxiv.org/abs/1609.08144.

8 APPENDIX

8.1 NMT DECODER SCHEMATIC

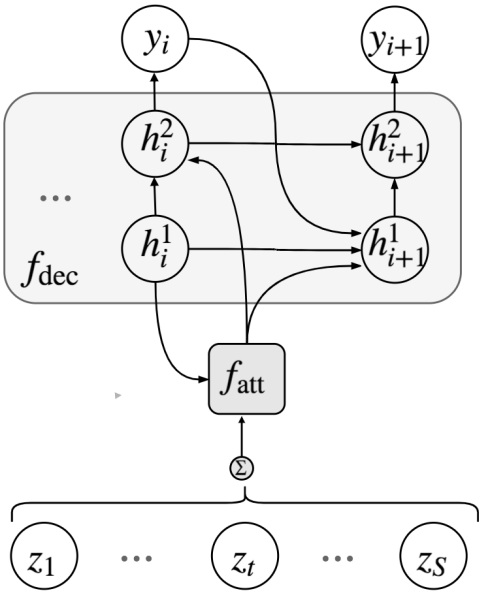


Figure 6: Schematic of the NMT decoder. The input sequence, $\mathbf{x}_{1:S}$, is encoded by a bidirectional encoder (not shown) into a sequence of encodings, $\mathbf{z}_{1:S}$. The attention network, f_{att}, computes a weighted sum of these encodings (computed weights not shown), based on conditioning information from $\mathbf{h}$ and provides the weighted encoding to the 2-layer decoder, f_{dec}, as indicated by the arrows. The decoder proceeds forward in time producing the translation one step at a time. As the decoder proceeds forward, it interacts with both the attention network and also receives as input the decoded output symbol from the previous time step.

8.2 ADJUSTED BLEU SCORE

Examples of pairs of sentences with different adjusted BLEU scores are as follows:

```
BLEU: 0.5
Sent 1: The role you play when creating the news is very important.
Sent 2: The part you play in making the news is very important.

BLEU: 0.09:
Sent 1: At the moment, men are overweighed by men.
Sent 2: Currently the majority of staff are men.

BLEU: 0.05:
Sent 1: Austria has also introduced controls to its southern and eastern boundaries.
Sent 2: The German government has also established the most recent and national
and international government institutions.

BLEU: 0.01:
Sent 1: &quot; But then, suddenly, our small Jumbo was there &quot;, says Brigitte Probst.
Sent 2: Scientists, who are not in the right place, are not in the right place.

BLEU: 0.001:
Sent 1: Mr Corbyn said that Angela Eagle, the shadow of Economic Minister, will be
represented in the economic minister, when he does not participate-for example, when
Mr. Cameron is on the
```

Sent 2: Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr. Mr.

As seen above, an adjusted BLEU score of < 0.01 means the two sentences have very few words in common.

8.3 Additional Examples

Additional examples of perturbed input sentences. For many sentences, there were more than one combination of token and position that caused a hallucination. In those cases, we give several examples of different perturbations and their hallucinations.

'@@' indicates a subword token. For example, perturbing the beginning of the sentence "Vor elf..." with g@@ would give: "gVor elf..."

Source: Vor elf Jahren sitzt Sufjan Stevens auf der Bühne im Kölner Prime Club (heute: Luxor).
Reference: Eleven years ago, Sufjan Stevens sits on the stage in the Prime Club (now Luxor) in Cologne.
NMT Translation: About eleven years ago Sufjan Stevens sits on the stage in Cologne Prime Club (today: Luxor).
Beginning: g@@: EPEFA is the first year of the year.
Beginning: P@@: It is possible to take a look at the location of the hotel.
Randomly: P@@: It is possible to take a look at the location of the hotel.

Source: Die Leichen von 29 Toten werden von Muslimen bei den ersten Begräbnissen in Mekka durch die Straßen getragen
Reference: Bodies of 29 of the dead carried through the streets by Muslims as the first funerals take place in Mecca
NMT Translation: The bodies of 29 dead are carried by Muslims at the first slopes in Mecca by the streets.
Beginning: er@@: I didn 't hesitate to hesitate to contact me.
Beginning: d: EEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEEE

Source: Caldiero sprach mit E! Nachrichten nach dem hart erkämpften Sieg, noch immer unter dem Schock über den Gewinn des Großen Preises von 1 Million $.
Reference: Caldiero spoke with E! News after the hard-fought victory, still in shock about winning the $1 million grand prize.
NMT Translation: Caldiero spoke with E, after the hard won victory, still under the shock of the winning of the Grand Prix of 1 million $.
Beginning: mit: It was said to have a lot of fun in the world.
Beginning: werden: Don't hesitate to contact us, if you want to be able to pay for you.
Beginning: dass: I 'm looking forward to having a lot of money.
Beginning: wird: I 'm looking forward to hearing from you.
Beginning: (: I 'm sure to say that I 'm looking for a lot of fun.
Beginning:): I 'm looking forward to hearing from the band.
Beginning: ich: Thus, we are looking forward to hearing from you.
Beginning: t: If you 're looking for a woman, you 'll be able to get a lot of money.
Beginning: :: Everyone who has been able to do so, is to say that it is not a good thing to do.
Beginning: hat: Everyone who has been able to do so, is the most important thing to do.
Beginning: wie: Two years ago, he was born in the 19th century, and he was born in the 19th century.
Beginning: einen: I 'm looking forward to having a lot of money.
Beginning: Ich: Don't hesitate to contact us, if you have any questions or questions you may have.
Beginning: A@@: If you are looking for a lot of fun, you 'll be able to find out more about it.
Beginning: d@@: AVI said, " I 'm looking forward to having a lot of money.
Beginning: Ver@@: Looking for a lot of fun, we have a lot of fun in the world.
Beginning: nur: I 'm looking forward to having a lot of work.
Beginning: uns: Beautiful Williams, which is the most important thing to do.
Beginning: durch: FAQUEVE THE WILL WITH THE WILL WITH THE WITH THE WITH THE WITH THE WITH THE WITH
Beginning: kann: Don't hesitate to contact us, but you will be able to find out more.
Beginning: P@@: Click here to see how you 're going to do this.
Beginning: Der: This is the reason why I have to say that it is not the case that it is possible to have a look at the fact that it is not possible to find out more.
Second to end: können: I am looking forward to hearing from you.
Second to end: o: We are looking forward to hearing from you.
Second to end: de: We are looking forward to hearing from you.
Randomly: auf: After the day, we have a lot of fun in the world.
Randomly: im: Surrounded by a lot of people, we have a lot of fun.

Source: Wer an einer posttraumatischen Belastungsstörung erkrankt ist, leidet unter anderem an sogenannten Flashbacks mit Atemnot, Schwindel, Herzrasen und Todesängsten.
Reference: Those who suffer from post-traumatic stress disorder suffer among other things from so-called flashbacks, difficulties in breathing, dizziness, palpitations and fear of death.
NMT Translation: Those who are suffering from a post-traumatic stress, suffer from the so-called flashbacks with breathing, darkness, heart, and death.
Beginning: mit: It is one of the oldest in the world.
Beginning: Sie: It is one of the oldest in the world.
Beginning: werden: Don't hesitate to contact us at any time.
Beginning: dass: For example, you will be able to find the right place for you.
Beginning: sind: Amongst other things, you will be able to enjoy a breathtaking view of the city.
Beginning: wird: There are also some of the world's most important tourist attractions.
Beginning:): It is one of the oldest in the world.
Beginning: als: The hotel is located in the heart of the city.
Beginning: e@@: It is also an ideal place for you.
Beginning: Ver@@: Amongst other things, you will be able to find the right place for you.
Beginning: nur: OSA is one of the most popular in the world.
Beginning:!: For example, it is also possible to find the right place for your stay.

Source: Laut den Medienberichten sei jedenfalls in der Früh ein erster Bus mit rund 50 Flüchtlingen in der serbischen Grenzstadt Sid angekommen, im Lauf des Tages würden weitere Busse erwartet.
Reference: According to media reports, at least one bus with around 50 refugees arrived in the Serbian border town of Sid in the early hours, and in the course of the day more buses are expected.
NMT Translation: According to the media report, at least in the morning, a first bus with about 50 refugees arrived in the Serb border town of Sid would expect more buses.
Beginning: werden: Don't hesitate to contact us at any time of your visit.
Beginning: wir: If you are looking for a busy day, then you 'll find the most important thing to do.
Beginning: Ich: Don't hesitate to contact us at any time of your visit.
Beginning: Wir: d Laut den Medienberfüre sei jederzeit einen ersten Bus einem ersten Bus mit rund 50 Rechtlingen im Serbischen Grenzstadt Sid, im Lauf der Tage would der weiteren Busse a</s>
Beginning: kann: Don't hesitate to contact us at any time.

Source: Neue Verhandlungen mit den Piloten
Reference: New negotiations with pilots
NMT Translation: New negotiations with the pilots
Beginning: Die: dass Neue News bis die Piloten
Beginning: ich:
Beginning: ": en Neue Negotiations mit den Piloten d
Beginning: einen: en Neue Negotiations mit den Piloten
Beginning: Das: de Neue News mit den Piloten
Beginning: Wir: d Neue Neue negotiations mit den Piloten d
Beginning: kann: de Neue Neue negotiations mit den Piloten die Piloten
Beginning: c@@: durch Neue Neue negotiations mit den Piloten
Randomly: mit: Neuist e mehr Jahren mit der Piloten d
Randomly: kann: de Neue Neue negotiations mit den Piloten die Piloten

Source: Für die Fed-Repräsentanten beeinflussen die Marktturbulenzen die komplexe Kalkulation, wann man die Zinsen erhöhen solle.
Reference: For Fed policymakers, the market turmoil adds to the complex calculus of when to raise the interest rate.
NMT Translation: For the Fed-Representatives, the market turbos influence the complex calculation, when we increase the interest rate.
Randomly: uns: For FF.C.

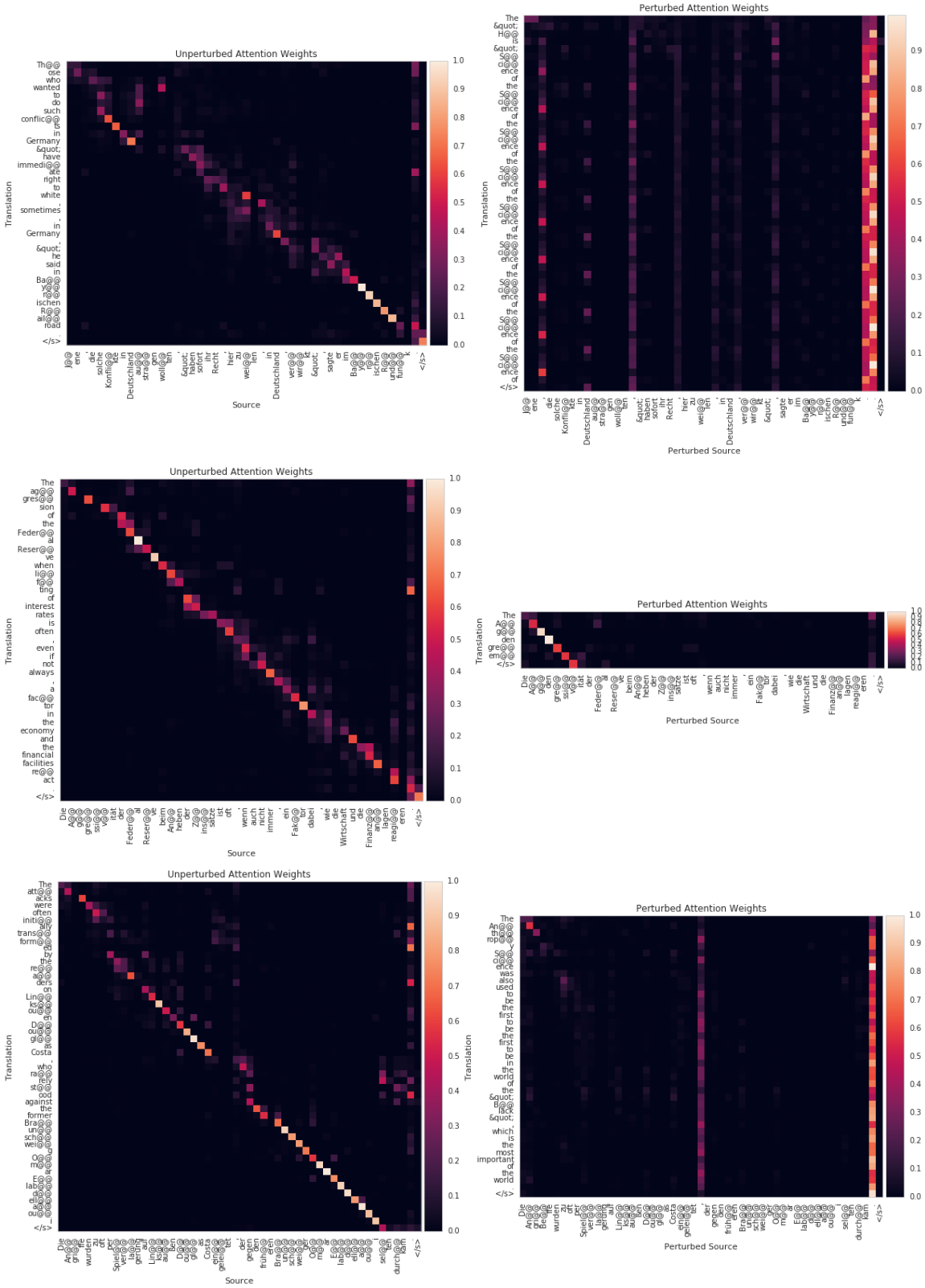
Source: Schon länger war bekannt, dass es unter der Eisdecke von Enceladus flüssiges Wasser geben muss.
Reference: It has long been known that there must be liquid water under Enceladus 's crust of ice.
NMT Translation: It was already well known that there must be fluid water under the ice cup of Enceladus.
Beginning: Ich: I am looking forward to hearing from you.

Source: Freundschaft schließen durch Backen.
Reference: Make friends through baking.
NMT Translation: Friendship by baking by baking.
Randomly: ich: Should you want to join us?

Source: In der Gesellschaft hat sich – auch wegen des Fachkräftemangels – mehrheitlich die Sichtweise durchgesetzt, dass Flüchtlingen schneller der Zugang zum Arbeitsmarkt ermöglicht werden sollte.
Reference: As far as the majority of society is now concerned, particularly in view of skills shortage, refugees should be allowed to gain faster access to the job market.
NMT Translation: Most of the time in the company – even because of the lack of expert labour – a majority of the view that refugees should be made faster from access to the labour market.
Beginning:!: Unfortunately, it is not possible to have a look at the situation.

Source: Monatelang war hinter verschlossenen Türen verhandelt, gegrübelt, debattiert und gezeichnet worden.
Reference: Plans have been negotiated, mulled over, debated, and plotted behind closed doors for months.
NMT Translation: Monatelang has been negotiating, debated and drawn behind closed doors.
Beginning: un@@: In the month of the month of the month of the month of the month of the month, it was a matter of course.
Beginning: uns: During the month of the month of the month, it was a matter of course that it is not possible for us to take a look.
Randomly: a@@: During the month of the month of the month, it was not possible to have the opportunity to take a look at the situation.
Randomly: o@@: During the month of the month of the month of the month, it was a matter of course that it was not possible to have a look at the situation.

9 Additional Attention Matrices


Unperturbed Attention Weights
Perturbed Attention Weights
Translation
Source
Perturbed Source

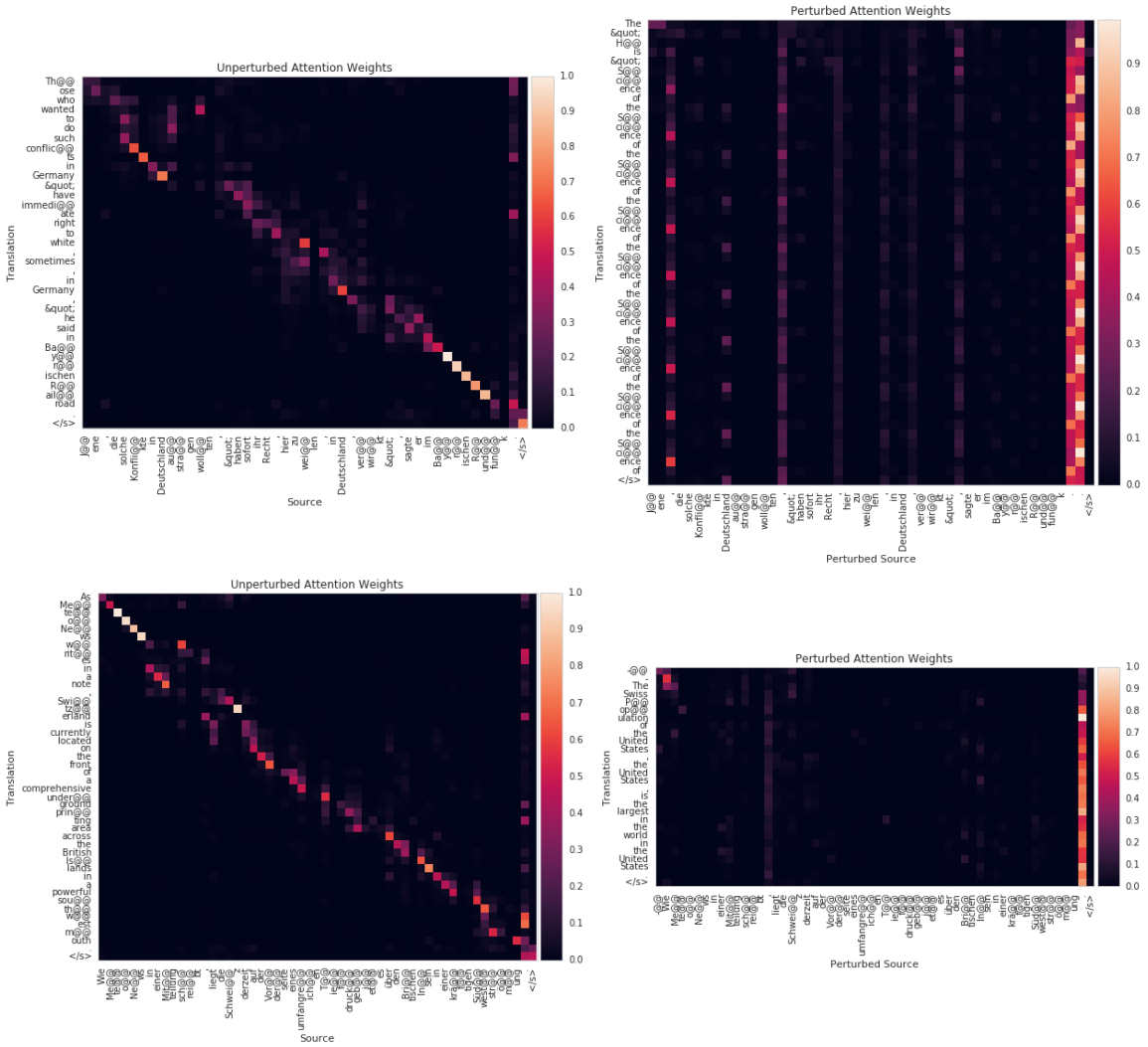


Figure 7: Example attention matrices. **(left)** Attention matrix for the original input sentence. **(right)** Attention matrix for the perturbed input sentence. All translations of the perturbed input sentence shown here are hallucinations. All decoding was done with the canonical model.