Addressing the Vulnerability of NMT in Input Perturbations

Weiwen Xu[1,2] *, **Ai Ti Aw**[1] †, **Yang Ding**[1], **Kui Wu**[1], **Shafiq Joty**[3]

[1]Institute for Infocomm Research, A*STAR
[2]The Chinese University of Hong Kong
[3]Nanyang Technological University
`wwxu@se.cuhk.edu.hk`
`{aaiti, ding_yang, wuk}@i2r.a-star.edu.sg`
`srjoty@ntu.edu.sg`

Abstract

Neural Machine Translation (NMT) has achieved significant breakthrough in performance but is known to suffer vulnerability to input perturbations. As real input noise is difficult to predict during training, robustness is a big issue for system deployment. In this paper, we improve the robustness of NMT models by reducing the effect of noisy words through a Context-Enhanced Reconstruction (CER) approach. CER trains the model to resist noise in two steps: (1) perturbation step that breaks the naturalness of input sequence with made-up words; (2) reconstruction step that defends the noise propagation by generating better and more robust contextual representation. Experimental results on Chinese-English (ZH-EN) and French-English (FR-EN) translation tasks demonstrate robustness improvement on both news and social media text. Further fine-tuning experiments on social media text show our approach can converge at a higher position and provide a better adaptation.

1 Introduction

Recent techniques (Bahdanau et al., 2014; Wu et al., 2016; Vaswani et al., 2017) in NMT have gained remarkable improvement in translation quality. However, robust NMT that is immune to real input noise remains a big challenge for NMT researchers. Real input noises can exhibit in many forms such as spelling and grammatical errors, homophones replacement, Internet slang, new words or even a valid word used in an unfamiliar or a new context. Unlike humans who can easily comprehend and translate such texts, most NMT models are not robust to generate appropriate and meaningful translations in the presence of such noises, challenging the deployment of NMT system in real scenarios.

---

*Work was done when the author was a staff in Institute for Infocomm Research, A*STAR.

†Corresponding Author

Input	通宵打游戏上分贼快
Ref.	It's super-fast to gain scores when playing games over the night.
MT	Play the game all night and take points thief fast.
CER	Play games all night to score points quickly.
Input	我已剪短了我的发,剪断了惩罚,剪一地伤透我的尴尬。。。。
Ref.	I have cut my hair, i cut off the punishment, i away the awkwardness that hurt me.
MT	I got my punishment, got rid of my embarrassment.
CER	I cut short my hair , cut off punishment , and cut off my embarrassment that hurts me.

Table 1: Examples of NMT's vulnerability in translating text containing noisy words ("zei" → "thief", "chengfa" → "punishment"). CER mitigates the effect of noisy words.

Noisy words have long been discussed in previous work. Aw et al. (2006) proposed the normalization approach to reduce the noise before translation. Tan et al. (2020a,b) addressed the character-level noise directly in the NMT model. Though these approaches addressed the effect of noisy words to some extent, they are limited to spelling errors, inflectional variations, and other noises definable during training. In addition, strong external supervision like a parallel corpus of noisy text translation or dictionary containing the translation of those noisy words are hard and expensive to obtain; they are also not practical in handling real noises as noisy words can exhibit in random forms and cannot be fully anticipated during training.

Belinkov and Bisk (2018) pointed out NMT models are sensitive to small input perturbations and if this issue is not addressed, it will continue to bottleneck the translation quality. In such cases, not only the word embeddings of perturbations may cause irregularities with the local context, the contextual representation of other words may also get affected by such perturbations (Liu et al., 2019). This phenomenon applies to valid words in unfamiliar context as well, which will also cause the translation to fail as illustrated in Table 1 (case 2).

In this paper, we define "noisy word" as a valid or invalid word that is uncommonly used in the context or not observed frequently enough in the training data. When encoding a sentence with such a noisy word, the contextual representation of other words in the sentence are affected by the "less jointly trained" noisy word embeddings. We refer this process as "noise propagation". Noise propagation can extend to the decoder and finally distort the overall translation.

The main intuition of our proposed method is to minimize this noise propagation and reduce the irregularities in contextual representation due to these words via a *Context-Enhanced Reconstruction* (CER) approach. To reduce the sensitivity of contextual towards noisy words in the encoder, we inject made-up words randomly to the source side of the training data to break the text naturalness. We then use a *Noise Adaptation Layer* (NAL) to enable a more stable contextual representation by minimizing the reconstruction loss. In the decoder, we add perturbations with a semantic constraint and apply the same reconstruction loss. Unlike adversarial examples which are crafted to cause the target model to fail, our perturbation process does not have such constraint and does not rely on a target model. Our input perturbations are randomly generated, representing any types of noises that can be observed in real-world usage. This makes the perturbation process generic, easy and fast. Following (Cheng et al., 2018), we generate semantically related perturbations in the decoder to increase the diversity of the translations.

Together with NAL, our model shows its ability to resist noises in the input and produce more robust translations. Results on ZH-EN and FR-EN translation significantly improve over the baseline by +1.24 (MT03) and +1.4 (N15) BLEU on news domain, and +1.63 (*Social*), +1.3 (*mtnt18*) on social media domain respectively. Further fine-tuning experiments on FR-EN social media text even witness an average improvement of +1.25 BLEU over the best approach.

2   Related Work

Robust Training: Robust training has shown to be effective to improve the robustness of the models in computer vision (Szegedy et al., 2013). In Natural Language Processing, it involves augmenting the training data with carefully crafted noisy examples: semantically equivalent word substitu-

tions (Alzantot et al., 2018), paraphrasing (Iyyer et al., 2018; Ribeiro et al., 2018), character-level noise (Ebrahimi et al., 2018b; Tan et al., 2020a,b), or perturbations at embedding space (Miyato et al., 2016; Liang et al., 2020). Inspired by Lei et al. (2017) that nicely captures the semantic interactions in discourse relation, we regard noise as a disruptor to break semantic interactions and propose our CER approach to mitigate this phenomenon. We make up "noisy" words randomly to act as random noise in the input to break the text naturalness. Our experiment demonstrates its superiority in multiple dimensions.

Robust Neural Machine Translation: Methods have been proposed to make NMT models resilient not only to adequacy errors (Lei et al., 2019) but also to both natural and synthetic noise. Incorporating monolingual data into NMT has the capacity to improve the robustness (Sennrich et al., 2016a; Edunov et al., 2018; Cheng et al., 2016). Some non data-driven approaches that specifically designed to address the robustness problem of NMT (Sperber et al., 2017; Ebrahimi et al., 2018a; Wang et al., 2018; Karpukhin et al., 2019; Cheng et al., 2019, 2020) explored effective ways to synthesize adversarial examples into the training data. Belinkov and Bisk (2018) showed a structure-invariant word representation capable of addressing multiple typo noise. Cheng et al. (2018) used adversarial stability training strategy to make NMT resilient to arbitrary noise. Liu et al. (2019) added an additional phonetic embedding to overcome homophone noise.

Meanwhile, Michel and Neubig (2018) released a dataset for evaluating NMT on social media text. This dataset was used as a benchmark for WMT 19 Robustness shared task (Li et al., 2019) to improve the robustness of NMT models on noisy text. We show our approach also benefits the fine-tuning process using additional social media data.

3   Approaches

We propose a Context-Enhanced Reconstruction (CER) approach to learn robust contextual representation in the presence of noisy words through a perturbation step and a reconstruction step in both encoder and decoder during model training. Figure 1 shows the architecture.

The perturbation step automatically inserts made-up words in the input sequence $\mathbf{x}$ to generate a noisy example $\mathbf{x}'$. The noisy example mimics input where text naturalness is broken due to
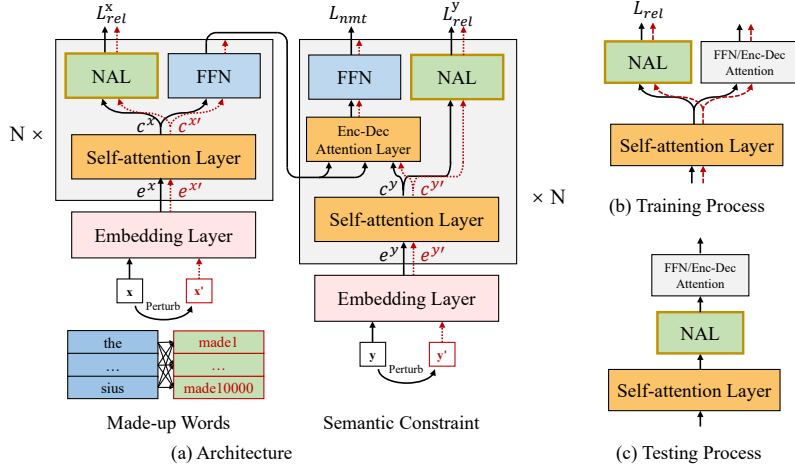
Figure 1: The architecture of CER (a), and the use of NAL in training (b) and testing (c). The solid lines indicate the flow for original input, while the dotted lines for noisy input, generated in the perturbation step.

the noisy words. Similarly, we perturb the output sequence **y** to **y**′ using a semantic constraint to generate noisy examples for the decoder to have more diversity in the translations.

The reconstruction step in the model aims to restore the contextual representation $\mathbf{c}^{\mathbf{x}'}$ of **x**′ to be similar to its corresponding original contextual representation $\mathbf{c}^{\mathbf{x}}$ in the encoder. Specifically, under the Transformer architecture (Figure 1), the reconstruction step aims to stabilize and minimize the disruption of attention distribution for a word over the whole input in the presence of inserted noise. The stabilization is needed for both clean and noisy words as both of their contextual representations are affected. For a noisy word, reconstruction reduces the attention to itself and encourages the construction of the contextual representation to leverage more on its clean neighbors. For clean words, reconstruction works as a denoise module to mitigate the interference of noisy words. For $\mathbf{c}^{\mathbf{y}'}$ in the decoder, the aim is to generate more examples with similar context as $\mathbf{c}^{\mathbf{y}}$. The reconstruction helps to normalize the contextual representation of semantically similar words.

3.1 Perturbing Input Text with Noise

We insert made-up words, representing any kinds of noise, to disturb the contextual representation during training. To create those words, we build a made-up dictionary $\mathcal{D}_x^-$ with M made-up words. As shown in Figure 1(a), made-up words are simply indexed slots in $\mathcal{D}_x^-$, whose embeddings are randomly initialized with no prior restriction and updated during training just as valid words. During the perturbation step, we randomly select multiple

positions in each input sequence based on probability σ_x and replace the words with any arbitrary made-up words in $\mathcal{D}_x^-$.

For the decoder, as the aim is not to insert noise but to increase the diversity of translation, we add small perturbations with a semantic constraint to make the model robust. Specifically, we randomly select multiple positions in each target sequence with a probability σ_y and perturb the corresponding words. For the word y_i chosen to be perturbed, we create a dynamic set $\mathcal{V}_{y_i}$ consisting of m words having the highest cosine similarity with it (excluding y_i). We average the embeddings of the words in $\mathcal{V}_{y_i}$ as the perturbation for y_i.

$$\mathcal{V}_{y_i} = \underset{y_j \in \mathcal{D}_y, j \neq i}{top\_m} (cos(e^{y_i}, e^{y_j})) \qquad (1)$$

$$e^{y_i'} = \frac{1}{m} \sum_{y_j \in \mathcal{V}_{y_i}} e^{y_j} \qquad (2)$$

Where $\mathcal{D}_y$ is the target dictionary, e^{y_j} is the target word embedding for y_j and $e^{y_i'}$ is the perturbed embedding for y_i.

3.2 Reconstructing Contextual Representation

As the injected noise in **x**′ affects the self-attention mechanism in producing correct contextual representation, we regularize the contextual representation using a Noise Adaptation Layer (NAL) immediately after the self-attention layer as depicted in Figure 1(a). This NAL is trained together with the NMT model and used as a reconstruction module during testing (See Figure 1(b),(c)).

Formally, let $\mathbf{c}_l^{\mathbf{x}}$ and $\mathbf{c}_l^{\mathbf{x}'}$ be the outputs of the self-attention in the l-th encoder layer for **x** and **x**′

respectively. We train the NAL by:

$$\mathcal{L}_{nal}^x(\boldsymbol{\theta}_{nal}^x) = \frac{1}{|S|} \sum_{(\mathbf{x},\mathbf{y}) \in S} \sum_{l=1}^{N} ||\mathbf{c}_l^{\mathbf{x}} - \text{NAL}(\mathbf{c}_l^{\mathbf{x}'})||^2 \quad (3)$$

Where $\boldsymbol{\theta}_{nal}^x$ are parameters of NAL, S is the training corpus and N is the encoder layer size. Given $\mathbf{c}^{\mathbf{x}'}$, NAL attempts to output a more correct contextual representation guided by $\mathbf{c}^{\mathbf{x}}$. We use a single layer feed-forward network (FFN) in (Vaswani et al., 2017) as our NAL implementation. Similarly, the reconstruction loss for decoder is:

$$\mathcal{L}_{nal}^y(\boldsymbol{\theta}_{nal}^y) = \frac{1}{|S|} \sum_{(\mathbf{x},\mathbf{y}) \in S} \sum_{l=1}^{N} ||\mathbf{c}_l^{\mathbf{y}} - \text{NAL}(\mathbf{c}_l^{\mathbf{y}'})||^2 \quad (4)$$

3.3 Model Training

We apply the perturbation step at the embedding layer, see Figure 1. The inserted noise in $\mathbf{x}'$ and $\mathbf{y}'$ would also receive gradient from the final loss function and update just like other clean words. NAL is added at each Transformer layer where the outputs are only used to calculate the reconstruction loss and not passed to the next layer. On the other hand, the output of FFN is propagated to the next layer as usual. The reconstruction step mainly serves as a stabilizer to prevent the noise from propagating.

The final training objective $\mathcal{L}$ is the combination of the above three loss functions, the original translation loss, the reconstruction loss for the encoder and the reconstruction loss for the decoder. Both λ_x and λ_y are set empirically to count for the relative importance.

$$\mathcal{L} = \mathcal{L}_{nmt}(\boldsymbol{\theta}_{nmt}) + \lambda_x \mathcal{L}_{nal}^x(\boldsymbol{\theta}_{nal}^x) + \lambda_y \mathcal{L}_{nal}^y(\boldsymbol{\theta}_{nal}^y) \quad (5)$$

4 Experiment Settings

Experiments are conducted on ZH-EN and FR-EN translation tasks for both news and social media domains. We also use social media text to fine-tune the NMT systems on FR-EN.

4.1 Data

ZH-EN: The training data consists of 1.25M sentence pairs extracted from LDC. For news domain, we use NIST MT02 as the development set and select the best model to test MT03, MT04, MT05, MT06 and MT08 news test sets. For social media domain, we create a test set (*Social*) consisting of 2000 sentences with three human annotated references. The source sentences are collected from public social media platforms in four Chinese-speaking

regions: Mainland China, Hong Kong, Taiwan and Singapore [1].

FR-EN: We use the same datasets as Michel and Neubig (2018). The training set consists of 2.16M sentence pairs extracted from *europarl-v7* and *news-commentary-v10*. We use the *newsdiscuss-dev2015* as development set and evaluate the model on two news test sets, *newstest2014* (N14) and *newsdiscusstest2015* (N15). We also evaluate on two social media test sets: *mtnt18* (Michel and Neubig, 2018) and *mtnt19* (Li et al., 2019).

FR-EN Fine-Tuning: We use the noisy training set (*mtnttrain*) provided by Michel and Neubig (2018) to fine-tune the FR-EN model.

We use fairseq's implementation of Transformer (Ott et al., 2019). In evaluation, we report case-insensitive tokenized BLEU for ZH-EN (Papineni et al., 2002) and sacre-BLEU (Post, 2018) for FR-EN. Following Michel and Neubig (2018), we do not use development set but only report best results on three social media test sets.

We segment the Chinese words using THU-LAC (Li and Sun, 2009) and tokenize both French and English words using tokenize.perl[2]. We apply BPE (Sennrich et al., 2016b) to get sub-word vocabularies for the encoder and decoder, both with 20K merge operations.

The hyper-parameters setting is the same as transformer-base in (Vaswani et al., 2017) except that we set dropout rate as 0.4 in all our experiments. Our proposed models are trained on top of Transformer baseline for efficiency purpose, where additional parameters from the embeddings of $\mathcal{D}_x^-$ and ReL are uniformly initialized. The madeup dictionary size M is set to 10,000. The size of dynamic set m is set to 3. The probability σ_x and σ_y are both set to 0.1 and balance coefficient λ_x and λ_y are both set to 1.

4.2 Baseline Models

We use Transformer as our baseline.
ZH-EN: We compare with Wang et al. (2018); Cheng et al. (2018, 2019). Wang et al. (2018) use a data augmentation approach by randomly replacing words in source and target sentences with other in-dictionary words. Cheng et al. (2018) use adversarial stability training to make NMT resilient to noise. Cheng et al. (2019) employ a white-box approach to synthesize adversarial examples.

---

[1]Available at https://github.com/wwxu21/CER-MT.

[2]https://github.com/moses-smt/mosesdecoder

Model	MT02 (DEV)	MT03	MT04	MT05	MT06	MT08	News Ave.	*Social*
		Existing systems						
Wang et al. (2018)	47.13	46.68	47.41	46.66	46.62	38.46	45.17	23.20
Cheng et al. (2018)	46.10	44.07	45.61	43.45	44.44	34.94	42.50	21.27
Cheng et al. (2019)	47.06	46.48	47.39	46.58	46.95	37.38	44.96	22.74
		Our systems						
Transformer	46.98	46.35	47.27	46.35	46.77	38.20	45.00	22.41
+ CER-Enc	47.65	46.72	47.53	47.06	47.04	38.53	45.38	23.81
+ CER	**48.34**	**47.59**	**48.21**	**47.29**	**47.64**	**39.33**	**46.01**	**24.04**

Table 2: Case-insensitive BLEU scores (%) on ZH-EN translation. MT02 is our development set.

Model	N14	N15	*mtnt18*	*mtnt19*
	Exising systems			
Wang et al.	29.2	31.1	25.0	28.1
Michel and Neubig	28.9	30.8	23.3	26.2
Zhou et al.*	N.A.	N.A.	24.5	**30.3**
	Our systems			
Transformer	29.7	31.0	25.2	28.0
+ CER-Enc	30.4	31.7	26.1	28.7
+ CER	**30.7**	**32.4**	**26.5**	29.1

Table 3: `sacreBLEU` (%) on FR-EN translation task. *Zhou et al. use more data to train their model.

Figure 2: BLEU improvements compared to Transformer baseline shown in Table 2 and Table 3 when applying noise-insertion methods.

FR-EN: In addition to Wang et al. (2018), we compare with Michel and Neubig (2018); Zhou et al. (2019); Vaibhav et al. (2019) on FR-EN or FR-EN Fine-Tuning tasks. Michel and Neubig (2018) do the first benchmark of the noisy text translation tasks in three languages. Vaibhav et al. (2019) leverage effective synthetic noise to make NMT resilient to noisy text. We implement their approach on Transformer backbone. For a fair comparison, we limit the data to train back-translation models only with *mtnttrain*. Zhou et al. (2019) adopt a multitask transformer architecture with two decoders, where the first decoder learns to denoise and the second decoder learns to translate from the denoised text. They adopt the approach proposed by Vaibhav et al. (2019) to synthesize the noisy text for their first decoder.

We do not compare our model with (Berard et al., 2019; Helcl et al., 2019) as they use much more out-domain data, a great number of monolingual data and a bigger Transformer model, and hence not comparable with our experimental settings.

5 Results and Analysis

5.1 Comparison with Baseline Models

Table 2 and Table 3 show the performance on ZH-EN and FR-EN tasks. We show the results of applying CER only to the encoder (+ CER-Enc), and to both the encoder and decoder (+ CER).
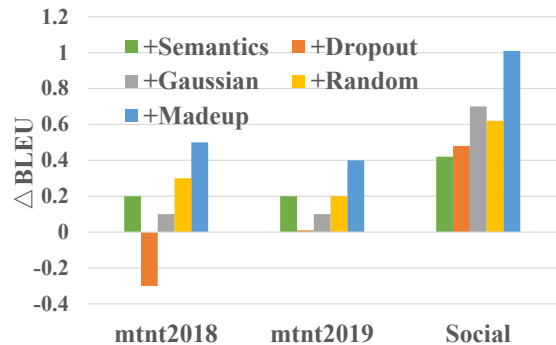
As illustrated, our approach improves the news

text translations on all test sets for both ZH-EN and FR-EN and outperforms the Transformer baseline in terms of average BLEU by +1.01 and +1.2 on ZH-EN and FR-EN respectively, illustrating the superiority of our approach.

The performance on social media test sets shows significant improvement with up to +1.63 BLEU over Transformer and +0.84 BLEU over the best approach (Wang et al., 2018) on ZH-EN. For FR-EN, our model outperforms Wang et al. (2018) by +1.5 and +1.0 BLEU on *mtnt18* and *mtnt19* respectively. Zhou et al. (2019) use *mtnttrain* and TED (Qi et al., 2018) to synthesize noisy sentences for their first decoder, hence effectively they are exploiting in-domain data during training and thus not quite a fair comparison in the evaluation. Nevertheless, CER still significantly outperforms Zhou et al. (2019) by +2.0 BLEU on *mtnt18*.

5.2 Effect of Noise

We investigate the effect of different noise-insertion methods by dynamically inserting noise into the source side of the original training set using different strategies with a same probability σ_x.

Madeup: Our approach to add made-up words.

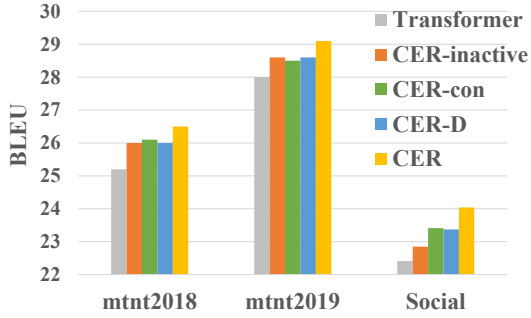*Semantics*: We test our semantic constraint in the decoder to assess if it benefits the encoder.

Figure 3: BLEU scores of CER variants.

Model	mtnt18	mtnt19
Existing systems		
Michel and Neubig	30.3	N/A
Wang et al.	35.1	36.7
Zhou et al.	31.7	32.8
Vaibhav et al.	36.0	37.5
Our systems		
Transformer (Base)	25.2	28.1
+FT	35.2	37.4
+FT w/ CER	**37.3**	**38.7**

Table 4: `sacreBLEU` on FR-EN fine-tuning task.

Dropout: We replace word embeddings with all-0 vectors, similar to enlarging the dropout rate.
Gaussian: Following the feature-level perturbations of Cheng et al. (2018), we add the Gaussian noise to a word embedding to simulate the noise.
Random: We replace a word with an arbitrary word in the dictionary. This would result in a valid word being placed in an unreasonable context.

Figure 2 shows the BLEU improvement of various noise-insertion methods on social media test sets. We find that nearly all kinds of noise-insertion methods improve the robustness of MT with the exception of *Dropout*. Since we have already set the dropout rate to an optimal rate, inserting additional *Dropout* noise does not increase but decreases the performance. As shown, *Madeup* improves the performance nearly twice than the rest of the noise-insertion methods. We conjecture *Semantics*, *Dropout* and *Gaussian* may be small and not diverse enough to simulate the real noisy words. Both *Random* and *Madeup* can break the text coherence. However, *Random* uses a random in-dictionary word, which can place a valid word in an unreasonable context and cause its embedding to update in a wrong direction. In fact, this method improves the robustness of NMT models at the cost of those replaced words. Our *Madeup* can entirely avoid this cost as we use made-up words to work as noisy words and does not cause any context change of all in-dictionary words.

5.3 Effect of NAL

To further gain insights on how NAL helps improve the robustness of NMT models. We create three variants to aid our analysis:

CER-inactive: We do not activate NAL at testing time. The contextual representation is feed directly into later FFN. This variant is to test the effectiveness of NAL.
CER-con: We remove NAL but only add a con-

straint to ensure $\{\mathbf{c^x}, \mathbf{c^{x'}}\}$ and $\{\mathbf{c^y}, \mathbf{c^{y'}}\}$ to be close respectively at training time. This forces the self-attention layer to reconstruct the correct contextual representation itself. This variant is to demonstrate the necessity to set apart the context generation module (self-attention layer) and the reconstruction module (NAL).
CER-D: We borrow the adversarial stability training strategy proposed in Cheng et al. (2018) here. In this variant, NAL is replaced by a discriminator and $\boldsymbol{\theta}_{nal}^x$ and $\boldsymbol{\theta}_{nal}^y$ are changed to the adversarial learning loss in Cheng et al. (2018). The purpose is to assess the effectiveness of NAL and the discriminator in context reconstruction.

Figure 3 shows the results of the three variants on three social media test sets. From the figure, we make the following observations.

NAL is effective at Test Time. The activation of NAL at test time helps to produce more reliable contextual representation. Notably, NAL gains +1.19 BLEU on *Social*.

NAL needs to be learnt separately. As shown in CER-con, by forcing self-attention layer to do both tasks (context generation and reconstruction), the performance improvement gets affected by at least 0.4 BLEU.

NAL is more effective than a discriminator to guide reconstruction. The improvements are less significant in all test sets when using a discriminator (CER-D) comparing to CER. Therefore, we can conclude that NAL is more effective than a discriminator to reconstruct the perturbed contextual representation and CER outperforms all variants.

5.4 FR-EN Fine-Tuning on Social Media Text

We fine-tune the same Transformer model in Table 3 with the social media data *mtnttrain* (*+FT*) and further include CER in the fine-tuning (+FT w/ CER). Table 4 shows our performance (*+FT w/ CER*) with other four fine-tuning approaches on *mtnttrain*. It shows that our CER also bene-

Model		*Social*
Google Translate		38.59
Ours	Baseline	39.01
	+FT	40.56 (+3.97%)
	+FT w/ CER	**40.82 (+4.64%)**

Table 5: Case-insensitive BLEU scores (relative improvement) on large-scale ZH-EN translation system.

fits the fine-tuning process and outperforms all the approaches in two noisy test sets. Specifically, it gains +2.1 and +1.3 BLEU over *+FT* on *mtnt18* and *mtnt19* and outperforms Vaibhav et al. (2019) by +1.3 and +1.2 BLEU respectively.

5.5 Experiments on Large-Scale Datasets

We first train a ZH-EN baseline model using 25M sentence pairs, which are mainly in news domain. Similar to the setting in Table 4, we apply both simple finetuning (+FT) and our CER (+ FT w/ CER) approach using 125K social media training data. We evaluate those models on *Social*. We also include the performance of Google Translate [3] here to show the competitiveness of our baseline model.

As shown in Table 5, our CER approach can still benefit the fine-tuning process even on the strong baseline. It should be noted that the baseline has already maintained high robustness with large-scale training data where improvement in such a model is hard to obtain. In fact, 125K in-domain data can only contribute to 1.55 BLEU improvement. Under this circumstance, the 0.26 BLEU improvement brought by CER should be highly valued considered no additional fine-tuning data is used.

6 Conclusions

In this work, we propose an approach to reduce the vulnerability of NMT models to input perturbations. Our input perturbation is easy, fast and not specific to a target victim model. Experimental results show our proposed approach improves the robustness on both news and social media text and helped to improve the translation of real input.

7 Acknowledgments

---

[3]https://translate.google.com/

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. 2018. Generating natural language adversarial examples. *arXiv preprint arXiv:1804.07998*.

AiTi Aw, Min Zhang, Juan Xiao, and Jian Su. 2006. A phrase-based statistical model for SMS text normalization. In *Proceedings of the COLING/ACL 2006 Main Conference Poster Sessions*, pages 33–40, Sydney, Australia. Association for Computational Linguistics.

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

Yonatan Belinkov and Yonatan Bisk. 2018. Synthetic and natural noise both break neural machine translation. In *International Conference on Learning Representations*.

Alexandre Berard, Ioan Calapodescu, and Claude Roux. 2019. Naver labs europe's systems for the wmt19 machine translation robustness task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 526–532, Florence, Italy. Association for Computational Linguistics.

Yong Cheng, Lu Jiang, and Wolfgang Macherey. 2019. Robust neural machine translation with doubly adversarial inputs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4324–4333, Florence, Italy. Association for Computational Linguistics.

Yong Cheng, Lu Jiang, Wolfgang Macherey, and Jacob Eisenstein. 2020. AdvAug: Robust adversarial augmentation for neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics.

Yong Cheng, Zhaopeng Tu, Fandong Meng, Junjie Zhai, and Yang Liu. 2018. Towards robust neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1756–1766, Melbourne, Australia. Association for Computational Linguistics.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1965–1974, Berlin, Germany. Association for Computational Linguistics.

Javid Ebrahimi, Daniel Lowd, and Dejing Dou. 2018a. On adversarial examples for character-level neural

machine translation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 653–663, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. 2018b. HotFlip: White-box adversarial examples for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 31–36, Melbourne, Australia. Association for Computational Linguistics.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, Brussels, Belgium. Association for Computational Linguistics.

Jindich Helcl, Jindich Libovick, and Martin Popel. 2019. Cuni system for the wmt19 robustness task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 539–543, Florence, Italy. Association for Computational Linguistics.

Mohit Iyyer, John Wieting, Kevin Gimpel, and Luke Zettlemoyer. 2018. Adversarial example generation with syntactically controlled paraphrase networks. 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 1875–1885, New Orleans, Louisiana. Association for Computational Linguistics.

Vladimir Karpukhin, Omer Levy, Jacob Eisenstein, and Marjan Ghazvininejad. 2019. Training on synthetic noise improves robustness to natural noise in machine translation. *arXiv preprint arXiv:1902.01509*.

Wenqiang Lei, Xuancong Wang, Meichun Liu, Ilija Ilievski, Xiangnan He, and Min-Yen Kan. 2017. Swim: A simple word interaction model for implicit discourse relation recognition. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4026–4032.

Wenqiang Lei, Weiwen Xu, Ai Ti Aw, Yuanxin Xiang, and Tat Seng Chua. 2019. Revisit automatic error detection for wrong and missing translation – a supervised approach. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 942–952, Hong Kong, China. Association for Computational Linguistics.

Xian Li, Paul Michel, Antonios Anastasopoulos, Yonatan Belinkov, Nadir Durrani, Orhan Firat, Philipp Koehn, Graham Neubig, Juan Pino, and Hassan Sajjad. 2019. Findings of the first shared task on machine translation robustness. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 91–102,

Florence, Italy. Association for Computational Linguistics.

Zhongguo Li and Maosong Sun. 2009. Punctuation as implicit annotations for Chinese word segmentation. *Computational Linguistics*, 35(4):505–512.

Hongru Liang, Wenqiang Lei, Paul Yaozhu Chan, Zhenglu Yang, Maosong Sun, and Tat-Seng Chua. 2020. Pirhdy: Learning pitch-, rhythm-, and dynamics-aware embeddings for symbolic music. In *MM '20: The 28th ACM International Conference on Multimedia, Virtual Event / Seattle, WA, USA, October 12-16, 2020*, pages 574–582. ACM.

Hairong Liu, Mingbo Ma, Liang Huang, Hao Xiong, and Zhongjun He. 2019. Robust neural machine translation with joint textual and phonetic embedding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3044–3049, Florence, Italy. Association for Computational Linguistics.

Paul Michel and Graham Neubig. 2018. MTNT: A testbed for machine translation of noisy text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 543–553, Brussels, Belgium. Association for Computational Linguistics.

Takeru Miyato, Andrew M Dai, and Ian Goodfellow. 2016. Adversarial training methods for semi-supervised text classification. *arXiv preprint arXiv:1605.07725*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting bleu scores. *arXiv preprint arXiv:1804.08771*.

Ye Qi, Devendra Sachan, Matthieu Felix, Sarguna Padmanabhan, and Graham Neubig. 2018. When and why are pre-trained word embeddings useful for neural machine translation? In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 529–535, New Orleans, Louisiana. Association for Computational Linguistics.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically equivalent adversarial rules for debugging NLP models. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*,

pages 856–865, Melbourne, Australia. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Matthias Sperber, Jan Niehues, and Alex Waibel. 2017. Toward robust neural machine translation for noisy input sequences. In *International Workshop on Spoken Language Translation (IWSLT)*.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Samson Tan, Shafiq Joty, Min-Yen Kan, and Richard Socher. 2020a. It's morphin' time! Combating linguistic discrimination with inflectional perturbations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2920–2935, Online. Association for Computational Linguistics.

Samson Tan, Shafiq Joty, Lav Varshney, and Min-Yen Kan. 2020b. Mind your inflections! Improving NLP for non-standard Englishes with Base-Inflection Encoding. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5647–5663, Online. Association for Computational Linguistics.

Vaibhav Vaibhav, Sumeet Singh, Craig Stewart, and Graham Neubig. 2019. Improving robustness of machine translation with synthetic noise. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1916–1920, Minneapolis, Minnesota. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. 2018. SwitchOut: an efficient data augmentation algorithm for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 856–861, Brussels, Belgium. Association for Computational Linguistics.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Shuyan Zhou, Xiangkai Zeng, Yingqi Zhou, Antonios Anastasopoulos, and Graham Neubig. 2019. Improving robustness of neural machine translation with multi-task learning. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 565–571, Florence, Italy. Association for Computational Linguistics.