SYNTHETIC AND NATURAL NOISE BOTH BREAK NEURAL MACHINE TRANSLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Character-based neural machine translation (NMT) models alleviate out-of-vocabulary issues, learn morphology, and move us closer to completely end-to-end translation systems. Unfortunately, they are also very brittle and easily falter when presented with noisy data. In this paper, we confront NMT models with synthetic and natural sources of noise. We find that state-of-the-art models fail to translate even moderately noisy texts that humans have no trouble comprehending. We explore two approaches to increase model robustness: structure-invariant word representations and robust training on noisy texts. We find that a model based on a character convolutional neural network is able to simultaneously learn representations robust to multiple kinds of noise.

1 INTRODUCTION

Humans have surprisingly robust language processing systems which can easily overcome typos, misspellings, and the complete omission of letters when reading (Rawlinson, 1976). A particularly extreme and comical exploitation of our robustness came years ago in the form of a popular meme:

> "Aoccdrnig to a rscheearch at Cmabrigde Uinervtisy, it deosn't mttaer in waht oredr the ltteers in a wrod are, the olny iprmoetnt tihng is taht the frist and lsat ltteer be at the rghit pclae."

A person's ability to read this text comes as no surprise to the Psychology literature. Saberi & Perrott (1999) found that this robustness extends to audio as well. They experimented with playing parts of audio transcripts backwards and found that it did not effect comprehension. Rayner et al. (2006) found that in noisier settings reading comprehension only slowed by 11%. McCusker et al. (1981) found that the common case of swapping letters could often go unnoticed by the reader. The exact mechanisms and limitations of our understanding system are unknown. There is some evidence that we rely on word shape (Mayall et al., 1997), that we can switch between whole word recognition and piecing together words from letters (Reicher, 1969; Pelli et al., 2003), and there appears to be no evidence that the first and last letter positions are required to stay constant for comprehension.[1]

In stark contrast, neural machine translation (NMT) systems, despite their pervasive use, are immensely brittle. This took an unfortunate turn recently when a mistranslation led to someone's wrongful arrest. The Arabic word يصبحهم (*ySbHhm*), roughly translating to a blessing for Good Morning, was confused for يذبحهم (*y*bHhm*), meaning to hurt or slaughter someone, and interpreted as a call to terrorism.[2] Facebook's MT system mistakenly confused two words that only differ by one character, a situation that is challenging for a character-based NMT system. While typos and noise are not new to NLP, our systems are rarely trained to explicitly address them, as we instead hope that the relevant noise will occur in the training data.

Despite these weaknesses the move to character-based NMT is important. It helps us tackle the long tailed distribution of out-of-vocabulary words in natural language, as well as reduce computation load of dealing with large word embedding matrices. NMT models based on characters and other sub-word units are able to extract stem and morphological information to generalize to unseen words

[1] One caveat we feel is important to note is that most of the literature in Psychology has focused on English.

[2] https://www.theguardian.com/technology/2017/oct/24/facebook-palestine-israel-translates-good-morning-attack-them-arrest

and conjugations. They perform very well in practice on a range of languages (Sennrich et al., 2016a; Wu et al., 2016). In many cases, these models actually discover an impressive amount of morphological information about a language (Belinkov et al., 2017a). Unfortunately, training (and testing) on clean data makes models brittle and, arguably, unfit for broad deployment.

Figure 1 shows how the performance of a state-of-the-art NMT system degrades when translating German to English as a function of the percent of German words modified. Here we show two types of noise: 1) Random permutation of the word, 2) Swapping a pair of adjacent letters in the center of words. We discuss these types of noise and others in depth in section 4.2. The important thing to note is that even small amounts of noise lead to substantial drops in performance.

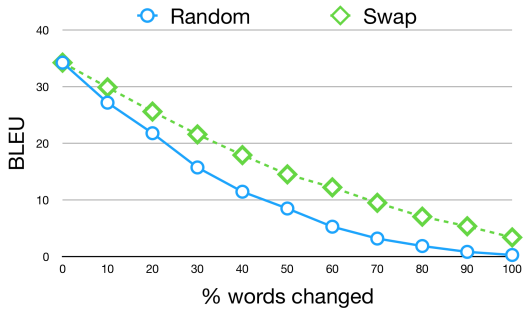

Figure 1: Degradation of `Nematus` (Sennrich et al., 2017) performance as noise increases.

To address these trends and investigate the effects of noise on NMT, we explore two simple strategies for increasing model robustness: using structure-invariant representations and robust training on noisy data, a form of adversarial training (Szegedy et al., 2014; Goodfellow et al., 2015). We find that a character CNN representation trained on an ensemble of noise types is robust to all kinds of noise. We shed some light on the model ability to learn robust representations to multiple types of noise, and point to remaining difficulties in handling natural noise. Our goal is two fold: 1) initiate a conversation on robust training and modeling techniques in NMT, and 2) promote the creation of better and more linguistically accurate artificial noise to be applied to new languages and tasks.

2 Adversarial Examples

The growing literature on adversarial examples has demonstrated how dangerous it can be to have brittle machine learning systems being used so pervasively in the real world (Biggio et al., 2012; Szegedy et al., 2014; Goodfellow et al., 2015; Mei & Zhu, 2015). Small changes to the input can lead to dramatic failures of deep learning models (Szegedy et al., 2014; Goodfellow et al., 2015). In the machine vision field, imperceptible changes to the input image that are indistinguishable by humans can lead to model misclassification. This leads to potential for malicious attacks using adversarial examples. An important distinction is often drawn between white-box attacks, where adversarial examples are generated with access to the model parameters, and black-box attacks, where examples are generated without such access (Papernot et al., 2016; 2017; Liu et al., 2017; Narodytska & Kasiviswanathan, 2017). While more common in the vision domain, recent work has used black-box adversarial examples for NLP evaluation (Jia & Liang, 2017). Here we devise simple methods for generating adversarial examples for NMT. We do not assume any access to the NMT models' gradients, instead relying on cognitively-informed and naturally occurring language errors to generate noise.

The other side of the coin is to improve models' robustness to adversarial examples (Globerson & Roweis, 2006; Cretu et al., 2008; Rubinstein et al., 2009; Chan et al., 2017). To some extent, adversarial training – including adversarial examples in the training data – can improve the model ability to cope with such examples at test time (Szegedy et al., 2014; Goodfellow et al., 2015). While this kind of defense is sensitive to the type of adversarial examples seen in training, it can be made more robust by *ensemble adversarial training* – training on examples transfered from multiple pre-trained models (Tramèr et al., 2017) – which was applied to image classification. We explore ensemble training by combining multiple types of noise at training time, and observe similar increased robustness in the machine translation scenario.

3 MT SYSTEMS

The rise of end-to-end models in neural machine translation has led to recent interest in understanding how these models operate. Several studies investigated the ability of such models to learn linguistic properties at morphological (Vylomova et al., 2016; Belinkov et al., 2017a; Dalvi et al., 2017), syntactic (Shi et al., 2016; Sennrich, 2017), and semantic levels (Belinkov et al., 2017b). The use of characters or other sub-word units emerges as an important component in these models. Our work complements previous studies by presenting such NMT systems with noisy examples and exploring methods for increasing their robustness.

We experiment with three different NMT systems with access to character information at different levels. First, we use the fully character-level model of Lee et al. (2017). This is a sequence-to-sequence model with attention (Sutskever et al., 2014; Bahdanau et al., 2014) that is trained on characters to characters (`char2char`). It has a complex encoder with convolutional, highway, and recurrent layers, and a standard recurrent decoder. See Lee et al. (2017) for architecture details. This model was shown to have excellent performance on the German→English and Czech→English language pairs. We use the pre-trained German/Czech→English models.

Second, we use `Nematus` (Sennrich et al., 2017), a popular NMT toolkit that was used in top-performing contributions in shared MT tasks in last year's WMT (Sennrich et al., 2016b) and IWSLT (Junczys-Dowmunt & Birch, 2016). It is another sequence-to-sequence model with several architecture modifications, especially operating on sub-word units using byte-pair encoding (BPE) (Sennrich et al., 2016a). We experimented with both their single best and ensemble BPE models, but saw no significant difference in their performance under noise, so we report results with their single best WMT models for German/Czech→English.

Finally, we train an attentional sequence-to-sequence model with a word representation based on a character convolutional neural network (`charCNN`). This model retains the notion of a word but learns a character-dependent representation of words. It was shown to perform well on morphologically-rich languages (Costa-jussà & Fonollosa, 2016), thanks to its ability to learn morphologically-informative representations (Belinkov et al., 2017a).

The `charCNN` model has two long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) layers in the encoder and the decoder. A CNN over characters in each word replaces the word embeddings on the encoder side (for simplicity, the decoder side is word-based). We use 1000 filters with a width of 6 characters. The character embedding size is set to 25. The convolutions are followed by a Tanh nonlinearity and max-pooling over the length over the word (Kim et al., 2015). We train the `charCNN` model with the NMT implementation in Kim (2016); all other settings are kept to their default values.

4 DATA

4.1 MT DATA

We use the TED talks parallel corpus prepared for IWSLT 2016 (Cettolo et al., 2012) for testing all of the NMT systems, as well as for training the `charCNN` models. We follow the official training/development/test splits. All texts are tokenized with the Moses tokenizer. Table 1 summarizes statistics on the TED talks corpus.

Table 1: Statistics for the source-side of French/German/Czech→English parallel corpora.

	French			German			Czech		
	Train	Dev	Test	Train	Dev	Test	Train	Dev	Test
Sentences	235K	2.5K	0.8K	210K	2.5K	1.4K	122K	20K	1K
Words	5.2M	55K	16K	4M	50K	26K	2.1M	35K	15K

4.2 NOISE: NATURAL AND ARTIFICIAL

We insert noise into the source-side of the parallel MT data by utilizing naturally occurring errors and generating synthetic ones.

Table 2: Average number of available edits per word in natural noise datasets and the corresponding token recall of those edits on the training and test splits.

FR				DE				CS			
Words	Errors	Train	Test	Words	Errors	Train	Test	Words	Errors	Train	Test
65,156	2.7	40%	41%	1,344	2.5	37%	40%	6,036	2.6	46%	51%

4.2.1 NATURAL NOISE

Since we do not have access to a parallel corpus with natural noise, we instead harvest naturally occurring errors (typos, misspellings, etc.) from available corpora of edits to build a look-up table of possible lexical replacements. In this work, we restrict ourselves to single word replacements, but several of the corpora below also provide access to phrase replacements.

French Max & Wisniewski (2010) collected Wikipedia edit histories to form the Wikipedia Correction and Paraphrase Corpus (WiCoPaCo). For our purposes we only extracted the single word corrections, but future work should incorporate more complicated paraphrases. In the construction of their corpus they found the bulk of edits were due to incorrect diacritics, choosing the wrong homophone, and incorrect grammatical conjugation.

German Our German data combines two projects: RWSE Wikipedia Revision Dataset (Zesch, 2012) and The MERLIN corpus of language learners (Wisniewski et al., 2013). These corpora were created to measure spelling difficulty and test models of contextual fitness. Unfortunately, the datasets are quite small so we have combined them here.

Czech Our Czech errors come from manually annotated essays written by non-native speakers (Šebesta et al., 2017). Here, the authors found an incredibly diverse set of errors, and therefore phenomena of interest: capitalization, incorrectly replacing voiced and voiceless consonants (e.g. z/s, g/k), missing palatalization (matke/matce), error in valence, pronominal reference, inflection, colloquial forms, and so forth. Their analysis gives us the best insight into how difficult it would be to synthetically generate truly natural errors.

We insert these errors into the source-side of the parallel data by replacing every word in the corpus with an error if one exists in our dataset. When there is more than one possible replacement to choose we sample uniformly. Words for which there is no error are kept as is. Table 2 shows the number of words for which we were able to collect errors in each language, and the average number of errors per word. Despite the small size of the German and Czech datasets, we are able to replace up to half of the words in the corpus with errors. Due to the small size of the German and Czech datasets these percentages decrease for longer words (> 4 characters) to 25% and 32%, respectively.

4.2.2 SYNTHETIC NOISE

In addition to naturally collected sources of error, we also experiment with four types of synthetic noise: Swap, Middle Random, Fully Random, and Key Typo.

Swap : `Swap` The first and simplest source of noise is swapping two letters. This is a common occurrence when typing quickly and is easily implemented. We perform one swap per word, but do not alter the first or last letters. For this reason, this noise is only applied to words of length ≥ 4.

Middle Random : `Mid` Following the claims of the previously discussed meme, we randomize the order of all the letters in a word except for the first and last. Again, by necessity, this means we do not alter words shorter than four characters.

Fully Random : `Rand` As we are unaware of any strong results on the importance of the first and last letters we also include completely randomized words. This is a particularly extreme case, but we include it for completeness. This type of noise is applied to all words.

Keyboard Typo : `Key` Finally, using the traditional keyboards for our languages, we randomly replace one letter in each word with an adjacent key. This type of error should be much easier than the random settings as most of the word is left intact, but does introduce a completely new character which will often break the templates a system has otherwise learned to rely on.

Table 3: The effect of Natural (Nat) and synthetic noise (Swap swap, Middle Random Mid, Fully Random Rand, and Keyboard Typo Key) on models trained on clean (Vanilla) texts.

			Synthetic				
		Vanilla	Swap	Mid	Rand	Key	Nat
French	charCNN	42.54	10.52	9.71	1.71	8.26	17.42
German	charCNN	34.79	9.25	8.37	1.02	6.40	14.02
	char2char	29.97	5.68	5.46	0.28	2.96	12.68
	Nematus	34.22	3.39	5.16	0.29	0.61	10.68
Czech	charCNN	25.99	6.56	6.67	1.50	7.13	10.20
	char2char	25.71	3.90	4.24	0.25	2.88	11.42
	Nematus	29.65	2.94	4.09	0.66	1.41	11.88

Table 4: An example noisy text with human and machine translations. All NMT systems fail on this noisy text.

Input	Luat eienr Stduie der Cambrdige Unievrstit speilt es kenie Rlloe in welcehr Reiehnfogle die Buhcstbaen in eniem Wrot vorkmomen, die eingzie whctige Sahce ist, dsas der ertse und der lettze Buhcstbaen stmimt .
Human	According to a study from Cambridge university, it doesn't matter which order letters in a word are, the only important thing is that the first and the last letter appear in their correct place
char2char	Cambridge Universttte is one of the most important features of the Cambridge Universttten , which is one of the most important features of the Cambridge Universttten .
Nematus	Luat eienr Stduie der Cambrant Unievrstilt splashed it kenie Rlloe in welcehr Reiehnfogle the Buhcstbaen in eniem Wred vorkmomen, die eingzie whcene Sahce ist, DSAs der ertse und der lettze Buhcstbaen stmimt .
charCNN	According to the <unk> of the Cambridge University , it 's a little bit of crude oil in a little bit of recycling , which is a little bit of a cool cap , which is a little bit of a strong cap , that the fat and the <unk> bites is consistent .

5 Failures to Translate Noisy Texts

Table 3 shows BLEU scores of models trained on clean (Vanilla) texts and tested on clean and noisy texts. All models suffer a significant drop in BLEU when evaluated on noisy texts. This is true for both natural noise and all kinds of synthetic noise. The more noise in the text, the worse the translation quality, with random scrambling producing the lowest BLEU scores.

The degradation in translation quality is especially severe in light of humans' ability to understand noisy texts. To illustrate this, consider the noisy text in Table 4, circulated as a meme several years ago. Humans are quite good at understanding such scrambled texts in a variety of languages.[3] We also verified this by obtaining a translation from a German native-speaker, unfamiliar with the meme. As shown in the table, the speaker had no trouble understanding and translating the sentence properly. In contrast, the state-of-the-art systems (char2char and Nematus) fail miserably on this text.

6 Dealing with Noise

6.1 Structure Invariant Representations

The three NMT models are all sensitive to word structure. The char2char and charCNN models both have convolutional layers on character sequences, designed to capture character n-grams. The model in Nematus is based on sub-word units obtained with BPE. It thus relies on character order

[3]Matt Davis has a wide collection of translations of this text in multiple languages: https://www.mrc-cbu.cam.ac.uk/personal/matt.davis/Cmabrigde/.

Table 5: Results of meanChar models trained and tested on different noise conditions: Scrambled (Scr), Keyboard Typo (Key), and Natural (Nat).

Train \ Test	French			German			Czech		
	Scr	Key	Nat	Scr	Key	Nat	Scr	Key	Nat
Vanilla	34.26	4.27	12.58	27.53	3.34	9.41	3.73	2.06	3.25
Key	31.88	29.75	13.16	10.04	8.84	4.45	2.03	1.9	1.42
Nat	26.94	5.30	27.49	15.65	3.06	26.26	1.66	1.52	1.58
Rand + Key	13.60	11.09	6.12	26.59	22.41	11.07	9.97	7.48	4.21
Rand + Nat	28.28	5.10	20.40	13.87	3.73	12.74	4.89	2.82	3.42
Key + Nat	31.30	26.94	24.24	6.62	5.41	5.75	1.62	1.68	1.58
Rand + Key + Nat	3.10	3.28	2.76	8.02	5.79	6.36	1.73	1.74	1.66

within and across sub-word units. All these models are therefore sensitive to types of noise generated by character scrambling (Swap, Mid, and Rand). Can we improve model robustness by adding invariance to these kinds of noise? Perhaps the simplest such model is to take the average character embeddings as a word representation. This model, referred to as meanChar, first generates a word representation by averaging character embeddings, and then proceeds with a word-level encoder similar to the charCNN model. The meanChar model is by definition insensitive to scrambling, although it is still sensitive to other kinds of noise (Key and Nat).

Table 5 (first row) shows the results of meanChar models trained on vanilla texts and tested on noisy texts (the results on vanilla texts are by definition equal to those on scrambled texts). Overall, the average character embedding proves to be a pretty good representation for translating scrambled texts, with performance of about 7 BLEU points below charCNN on French and German. The results on Czech are much worse, possibly due to its more complex morphology.

However, the meanChar model performance degrades quickly on other kinds of noise as the model trained on vanilla texts was not designed to handle Nat and Key types of noise.

6.2 Black-Box adversarial training

To increase model robustness we follow a black-box adversarial training scenario, where the model is presented with adversarial examples that are generated without direct access to the model (Papernot et al., 2016; 2017; Liu et al., 2017; Narodytska & Kasiviswanathan, 2017; Jia & Liang, 2017).

As shown in Table 5 (second block), training on noisy text can lead to improved performance. The models trained on Key perform well on Key in French, but not in the other languages. The models trained on Nat perform well in French and German, but not in Czech. Overall, training the meanChar model on noisy text does not appear to consistently increase its robustness to different kinds of noise. The meanChar model however was not expected to perform well on non-scrambling types of noise. Next we test whether the more complicated charCNN model is more robust to different kinds of noise, by training on noisy texts.

The results are shown in Table 6. In general, charCNN models that are trained on a specific kind of noise perform well on the same kind of noise at test time (results in **bold**). All models also maintain a fairly good quality on vanilla texts, although most perform slightly worse than models trained on vanilla texts. The robust training is sensitive to the kind of noise. Among the scrambling noise methods (Wwap/Mid/Rand), more noise helps in training: models trained on random noise can still translate Swap/Mid noise, but not vice versa. The three broad classes of noise (scrambling, Key, Nat) are not mutually-beneficial. Models trained on one do not perform well on the others. In particular, only models trained on natural noise can reasonably translate natural noise at test time. We find this result indicates an important difference between computational models and human performance, given that humans are not explicitly exposed to noisy examples when learning a language.

Next, we test whether we can increase training robustness by exposing the model to multiple types of noise during training. Our motivation is to see if models can perform well on more than one

Table 6: Results of charCNN models trained and tested on different noise conditions.

	Train \ Test	Vanilla	Swap	Mid	Rand	Key	Nat	Ave
French	Swap	39.01	**42.56**	33.64	2.72	4.85	16.43	23.20
	Mid	**42.46**	42.19	**42.17**	3.36	6.20	18.22	25.77
	Rand	39.53	39.46	39.13	**39.73**	3.11	16.63	29.60
	Key	38.49	10.56	8.69	1.08	**38.88**	16.86	19.10
	Nat	28.77	12.45	8.39	1.03	6.61	**36.00**	15.54
	Rand + Key	39.23	38.85	38.89	39.13	38.22	18.71	35.51
	Rand + Nat	36.86	38.95	38.44	38.63	6.67	33.89	32.24
	Key + Nat	38.47	17.33	10.54	1.52	38.62	34.66	23.52
	Rand + Key + Nat	36.97	36.92	36.65	36.64	35.25	31.77	**35.70**
German	Swap	32.66	**34.76**	29.03	2.19	4.78	13.37	19.47
	Mid	**34.32**	34.26	**34.27**	3.50	5.08	14.43	20.98
	Rand	33.65	39.46	39.13	**39.73**	3.00	14.47	28.24
	Key	32.87	10.13	8.39	1.16	**33.28**	13.88	16.62
	Nat	25.79	8.20	5.73	0.93	4.80	**34.59**	13.34
	Rand + Key	32.03	31.57	31.32	31.58	31.23	15.59	28.89
	Rand + Nat	32.37	32.40	31.91	32.11	4.77	33.00	27.76
	Key + Nat	30.39	13.51	8.99	1.53	32.23	33.46	20.02
	Rand + Key + Nat	31.29	30.93	30.54	30.04	29.81	31.60	**30.70**
Czech	Swap	**24.22**	**42.56**	33.64	2.72	6.00	9.03	19.69
	Mid	23.81	42.19	**42.17**	3.36	6.34	9.54	21.23
	Rand	23.44	33.44	33.75	**33.56**	3.70	8.10	**22.66**
	Key	23.15	7.06	6.04	1.56	**22.80**	10.16	11.79
	Nat	18.04	5.36	4.48	1.47	6.71	**21.64**	9.62
	Rand + Key	21.46	20.81	20.90	20.59	19.48	8.72	18.66
	Rand + Nat	20.59	21.56	20.49	20.53	5.89	18.39	17.91
	Key + Nat	19.55	6.59	5.72	1.40	21.31	19.54	12.35
	Rand + Key + Nat	21.30	21.33	20.38	19.94	19.25	18.38	20.10

kind of noise. We therefore mix up to three kinds of noise by sampling a noise method uniformly at random for each sentence. We then train a model on the mixed noisy training set and test it on both vanilla and (unmixed) noisy versions of the test set. We find that models trained on mixed noise are slightly worse than models trained on unmixed noise. However, the models trained on mixed noise are robust to the specific types of noise they were trained on. In particular, the model trained on a mix of Rand, Key, and Nat noise is robust to all noise kinds. Even though it's not the best on any one kind of noise, it achieves the best result on average.

This model is also able to translate the scrambled meme reasonably well:

> "According to a study of Cambridge University, it doesn't matter which technology in a word is going to get the letters in a word that is the only important thing for the first and last letter."

7 Analysis

7.1 Learning multiple kinds of noise in charCNN

The charCNN model was able to perform well on all kinds of noise by training on a mix of noise types. In particular, it performed well on scrambled characters even though its convolutions should be sensitive to the character order, as opposed to meanChar which is by definition invariant to character order. How then can charCNN learn to be robust to multiple kinds of noise at the same time? We speculate that different convolutional filters learn to be robust to different kinds of noise. A convolutional filter can in principle capture a mean (or sum) operation by employing equal or close to equal weights.

To test this, we analyze the weights learned by charCNN models trained on two kinds of input: completely scrambled words (Rand) without other kinds of noise, and a mix of Rand+Key+Nat

kinds of noise. For each model, we compute the variance across the filter dimension for each one of the 1000 filters and for each one out of 25 character embedding dimensions. The we average the variances across the 1000 filters. This yields 25 averaged variances, one for each character embedding dimension.

Figure 2 shows a box plot of these averages. Clearly, the variances of the weights learned by the `Rand` model are much smaller than those of the weights learned by the mixed noise model. In other words, the ensemble model trained on mixed noise learns more diverse weights, which can help it capture morphological templates in addition to average representations. Moreover, the variance of variances (size of the box) is larger in the mixed noise model, indicating that filters for different character embedding dimensions are more different from one another. In contrast, in the `Rand` model, the variance of variances is close to zero, indicating that in all character embedding dimensions the learned weights are of small variance; they do similar things, that is, the model learned to reproduce a representation similar to the `meanChar` model.

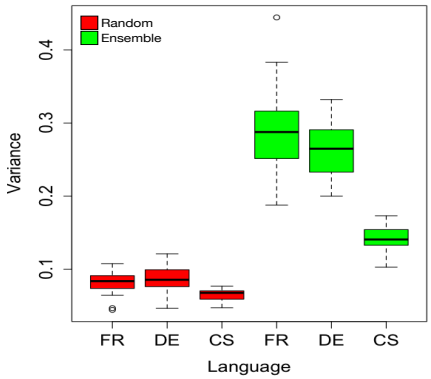

Figure 2: Variance of `charCNN` weights when trained on only `Rand` noise and on a mix of `Rand`, `Key`, and `Nat` noise.

7.2 Richness of Natural Noise

Natural noise appears to be very different from synthetic noise. None of the models that were trained only on synthetic noise were able to perform well on natural noise. We manually analyzed a small sample (~40 examples) of natural noise from the German dataset. We found that the most common sources of noise are phonetic or phonological phenomena in the language (34%) and character omissions (32%). The rest are incorrect morphological conjugations of verbs, key swaps, character insertions, orthographic variants, and other errors. Table 7 shows examples of these kinds of noise.

The most common types of natural noise – phonological and omissions – are not directly captured by our synthetic noise generation, and demonstrate that good synthetic errors will likely require more explicit phonemic and linguistic knowledge. This discrepancy helps explain why the models trained on synthetic noise were not particularly successful in translating natural noise.

Table 7: Examples of natural noise from the German errors dataset.

Error type	Examples
Phonetic	Tut/Tud (devoicing of final stops), sieht/zieht (s = /z/ before vowel), Trotzdem/Trozdem (tz = /z/), gekriegt/gekrigt (vowel length), Natürlich/Naturlich/Näturlich (diacritics)
Omission	erfahren/erfaren, Babysitter/Babysiter, selbst/sebst, Hausschuhe/Hausschue
Morphological	wohnt/wonnen, fortsetzt/forzusetzen, wünsche/wünchen
Key swap	Eltern/Eltren, Deine/Diene, nichts/nichst, Bahn/Bhan
Other	Detailhandelsfachfrau/Deitellhandfachfrau, Thema/Temer, Hausaufgabe/Hausausgabe

8 Conclusion

In this work, we have shown that character-based NMT models are extremely brittle and tend to break when presented with both natural and synthetic kinds of noise. We investigated methods for increasing their robustness by using a structure-invariant word representation and by ensemble training on adversarial examples of different kinds. We found that a character-based CNN can learn to address multiple types of errors that are seen in training. However, we observed rich characteristics of natural human errors that cannot be easily captured by existing models.

We believe that more work is necessary in order to immune NMT models against natural noise. As corpora with natural noise are limited, future work will have to design better NMT architectures that would be robust to noise without seeing it in the training data. New psychology results on how humans cope with natural noise might point to possible solutions to this problem.

REFERENCES

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural Machine Translation by Jointly Learning to Align and Translate. *arXiv preprint arXiv:1409.0473*, 2014.

Yonatan Belinkov, Nadir Durrani, Fahim Dalvi, Hassan Sajjad, and James Glass. What do Neural Machine Translation Models Learn about Morphology? In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 861–872, Vancouver, Canada, July 2017a.

Yonatan Belinkov, Lluís Màrquez, Hassan Sajjad, Fahim Dalvi, Nadir Durrani, and James Glass. Evaluating Layers of Representation in Neural Machine Translation on Part-of-Speech and Semantic Tagging Tasks. In *Proceedings of the 8th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Taipei, Taiwan, November 2017b. Association for Computational Linguistics.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning Attacks Against Support Vector Machines. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pp. 1467–1474, USA, 2012. Omnipress. ISBN 978-1-4503-1285-1. URL http://dl.acm.org/citation.cfm?id=3042573.3042761.

Mauro Cettolo, Christian Girardi, and Marcello Federico. WIT^3: Web Inventory of Transcribed and Translated Talks. In *Proceedings of the 16th Conference of the European Association for Machine Translation (EAMT)*, pp. 261–268, Trento, Italy, May 2012.

Patrick P. K. Chan, Zhi-Min He, Hongjiang Li, and Chien-Chang Hsu. Data sanitization against adversarial label contamination based on data complexity. *International Journal of Machine Learning and Cybernetics*, Jan 2017. ISSN 1868-808X. doi: 10.1007/s13042-016-0629-5. URL https://doi.org/10.1007/s13042-016-0629-5.

Marta R. Costa-jussà and José A. R. Fonollosa. Character-based Neural Machine Translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 357–361, Berlin, Germany, August 2016. Association for Computational Linguistics. URL http://anthology.aclweb.org/P16-2058.

Gabriela F. Cretu, Angelos Stavrou, Michael E. Locasto, Salvatore J. Stolfo, and Angelos D. Keromytis. Casting out Demons: Sanitizing Training Data for Anomaly Sensors. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy*, SP '08, pp. 81–95, Washington, DC, USA, 2008. IEEE Computer Society. ISBN 978-0-7695-3168-7. doi: 10.1109/SP.2008.11. URL https://doi.org/10.1109/SP.2008.11.

Fahim Dalvi, Nadir Durrani, Hassan Sajjad, Yonatan Belinkov, and Stephan Vogel. Understanding and Improving Morphological Learning in the Neural Machine Translation Decoder. In *Proceedings of the 8th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, Taipei, Taiwan, November 2017. Association for Computational Linguistics.

Amir Globerson and Sam Roweis. Nightmare at Test Time: Robust Learning by Feature Deletion. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pp. 353–360, New York, NY, USA, 2006. ACM. ISBN 1-59593-383-2. doi: 10.1145/1143844.1143889. URL http://doi.acm.org/10.1145/1143844.1143889.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations (ICLR)*, 2015.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8): 1735–1780, 1997.

Robin Jia and Percy Liang. Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2011–2021, Copenhagen, Denmark, September 2017.

Marcin Junczys-Dowmunt and Alexandra Birch. The University of Edinburgh's systems submission to the MT task at IWSLT. In *Proceedings of the First Conference on Machine Translation*, Seattle, USA, 2016.

Yoon Kim. Seq2seq-attn. https://github.com/harvardnlp/seq2seq-attn, 2016.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware Neural Language Models. *arXiv preprint arXiv:1508.06615*, 2015.

Jason Lee, Kyunghyun Cho, and Thomas Hofmann. Fully Character-Level Neural Machine Translation without Explicit Segmentation. *Transactions of the Association for Computational Linguistics (TACL)*, 2017.

Yanpei Liu, Xinyun Chen, Chang Liu, and Dawn Song. Delving into Transferable Adversarial Examples and Black-box Attacks. 2017.

Aurlien Max and Guillaume Wisniewski. Mining Naturally-occurring Corrections and Paraphrases from Wikipedias Revision History. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*, Valletta, Malta, may 2010. European Language Resources Association (ELRA). ISBN 2-9517408-6-7. URL https://wicopaco.limsi.fr.

K. Mayall, G.W. Humphreys, and A. Olson. Disruption to word or letter processing? the origins of case-mixing effects. *Journal of Experimental Psychology: Learning, Memory, & Cognition*, 23: 1275 – 1286, 1997.

L. X. McCusker, P. B. Gough, and R. G. Bias. Word recognition inside out and outside in. *Journal of Experimental Psychology: Human Perception and Performance*, 7(3):538 – 551, 1981.

Shike Mei and Xiaojin Zhu. Using Machine Teaching to Identify Optimal Training-set Attacks on Machine Learners. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, AAAI'15, pp. 2871–2877. AAAI Press, 2015. ISBN 0-262-51129-0. URL http://dl.acm.org/citation.cfm?id=2886521.2886721.

N. Narodytska and S. Kasiviswanathan. Simple Black-Box Adversarial Attacks on Deep Neural Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 1310–1318, July 2017. doi: 10.1109/CVPRW.2017.172.

Nicolas Papernot, Patrick McDaniel, and Ian Goodfellow. Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples. *arXiv preprint arXiv:1605.07277*, 2016.

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z. Berkay Celik, and Ananthram Swami. Practical Black-Box Attacks Against Machine Learning. In *Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security*, ASIA CCS '17, pp. 506–519, New York, NY, USA, 2017. ACM. ISBN 978-1-4503-4944-4. doi: 10.1145/3052973.3053009. URL http://doi.acm.org/10.1145/3052973.3053009.

D. G. Pelli, B. Farell, and D.C. Moore. The remarkable inefficiency of word recognition. *Nature*, 423:752 – 756, 2003.

G. E. Rawlinson. *The significance of letter position in word recognition*. PhD thesis, 1976.

Keith Rayner, Sarah J. White, Rebecca L. Johnson, and Simon P. Liversedge. Raeding Wrods With Jubmled Lettres There Is a Cost. *Psychological Science*, 17(3):192 – 193, 2006.

G. M. Reicher. Perceptual recognition as a function of meaningfulness of stimulus material. *Journal of Experimental Psychology*, 81(2):275 – 280, 1969.

Benjamin I.P. Rubinstein, Blaine Nelson, Ling Huang, Anthony D. Joseph, Shing-hon Lau, Satish Rao, Nina Taft, and J. D. Tygar. ANTIDOTE: Understanding and Defending Against Poisoning of Anomaly Detectors. In *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement*, IMC '09, pp. 1–14, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-771-4. doi: 10.1145/1644893.1644895. URL http://doi.acm.org/10.1145/1644893.1644895.

Kourosh Saberi and David R. Perrott. Cognitive restoration of reversed speech. *Nature*, 398(760), April 1999.

Karel Šebesta, Zuzanna Bedřichová, Kateřina Šormová, Barbora Štindlová, Milan Hrdlička, Tereza Hrdličková, Jiří Hana, Vladimír Petkevič, Tomáš Jelínek, Svatava Škodová, Petr Janeš, Kateřina Lundáková, Hana Skoumalová, Šimon Sládek, Piotr Pierscieniak, Dagmar Toufarová, Milan Straka, Alexandr Rosen, Jakub Náplava, and Marie Poláčková. CzeSL grammatical error correction dataset (CzeSL-GEC). Technical report, LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics, Charles University, 2017. URL https://lindat.mff.cuni.cz/repository/xmlui/handle/11234/1-2143.

Rico Sennrich. How Grammatical is Character-level Neural Machine Translation? Assessing MT Quality with Contrastive Translation Pairs. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pp. 376–382. Association for Computational Linguistics, 2017. URL http://aclweb.org/anthology/E17-2060.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 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)*, pp. 1715–1725. Association for Computational Linguistics, 2016a. doi: 10.18653/v1/P16-1162. URL http://aclanthology.coli.uni-saarland.de/pdf/P/P16/P16-1162.pdf.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Edinburgh Neural Machine Translation Systems for WMT 16. In *Proceedings of the First Conference on Machine Translation*, pp. 371–376, Berlin, Germany, August 2016b. Association for Computational Linguistics.

Rico Sennrich, Orhan Firat, Kyunghyun Cho, Alexandra Birch, Barry Haddow, Julian Hitschler, Marcin Junczys-Dowmunt, Samuel Läubli, Antonio Valerio Miceli Barone, Jozef Mokry, and Maria Nadejde. Nematus: a Toolkit for Neural Machine Translation. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 65–68, Valencia, Spain, April 2017. Association for Computational Linguistics. URL http://aclweb.org/anthology/E17-3017.

Xing Shi, Inkit Padhi, and Kevin Knight. Does String-Based Neural MT Learn Source Syntax? In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 1526–1534, Austin, Texas, November 2016. Association for Computational Linguistics. URL https://aclweb.org/anthology/D16-1159.

Ilya Sutskever, Oriol Vinyals, and Quoc VV Le. Sequence to Sequence Learning with Neural Networks. In *Advances in neural information processing systems*, pp. 3104–3112, 2014.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In *International Conference on Learning Representations (ICLR)*, 2014.

Florian Tramèr, Alexey Kurakin, Nicolas Papernot, Dan Boneh, and Patrick McDaniel. Ensemble Adversarial Training: Attacks and Defenses. *arXiv preprint arXiv:1705.07204*, 2017.

Ekaterina Vylomova, Trevor Cohn, Xuanli He, and Gholamreza Haffari. Word Representation Models for Morphologically Rich Languages in Neural Machine Translation. *arXiv preprint arXiv:1606.04217*, 2016.

Katrin Wisniewski, Karin Schne, Lionel Nicolas, Chiara Vettori, Adriane Boyd, Detmar Meurers, Andrea Abel, and Jirka Hana. MERLIN: An online trilingual learner corpus empirically grounding the European Reference Levels in authentic learner data, 10 2013. URL https://www.ukp.tu-darmstadt.de/data/spelling-correction/rwse-datasets.

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

Torsten Zesch. Measuring Contextual Fitness Using Error Contexts Extracted from the Wikipedia Revision History. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pp. 529–538, Avignon, France, April 2012. Association for Computational Linguistics.