On the Understanding of Back-translation

Wenxiang Jiao
2019-12-19

Investigating Backtranslation in Neural Machine Translation (EAMT 2018)

Alberto Poncelas, Dimitar Shterionov, Andy Way, Gideon Maillette de Buy Wenniger and Peyman Passban
School of Computing, DCU, ADAPT Centre

Storyline

There are many unknown factors regarding the actual effects of back-translated data on the translation capabilities of an NMT model.

RQ: How will using back-translated data as a training corpus affect the performance of an NMT model?

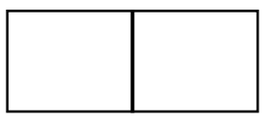
- Authentic vs. Hybrid
- Authentic vs. Synthetic

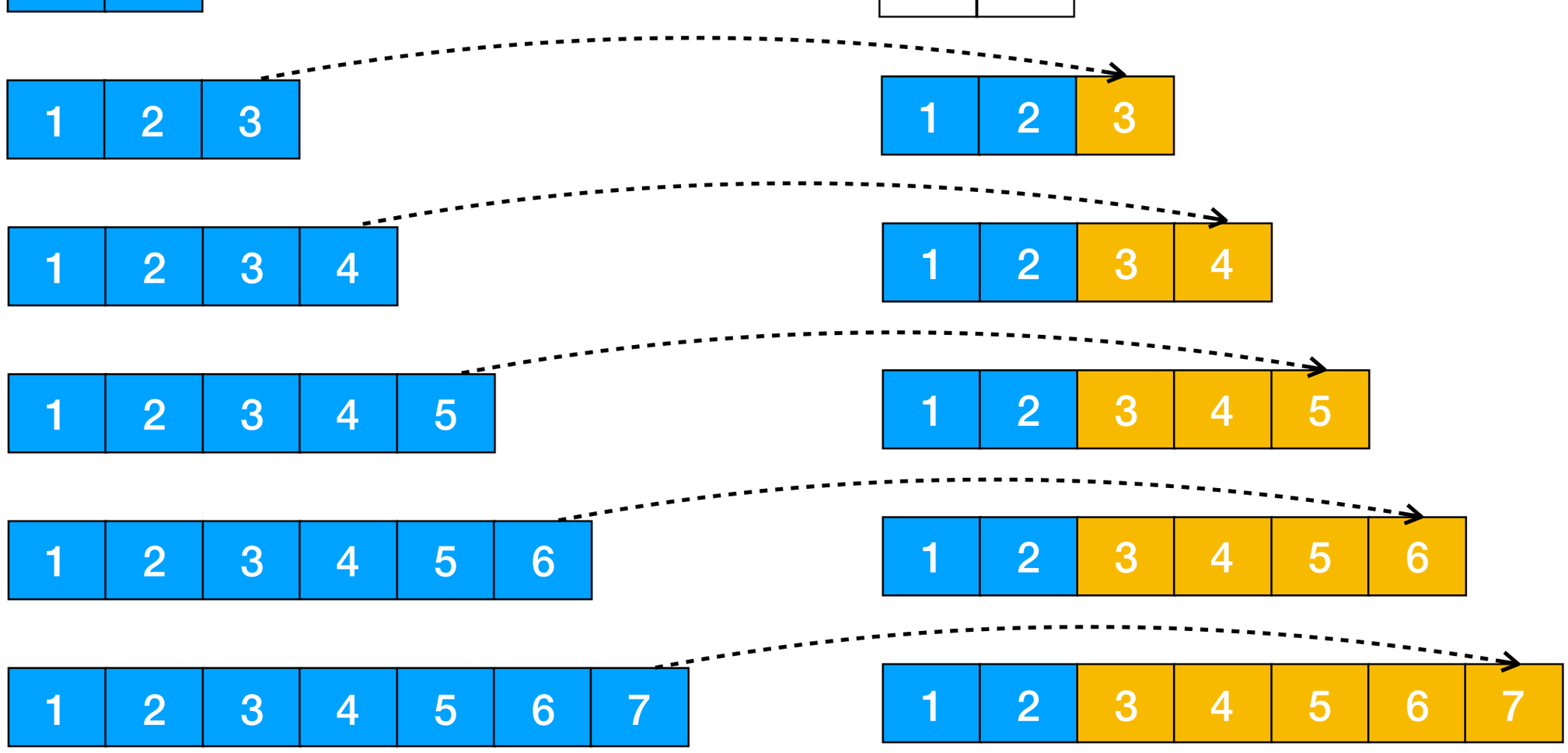
Data Preparation

WMT14 En-De (~4.5M)

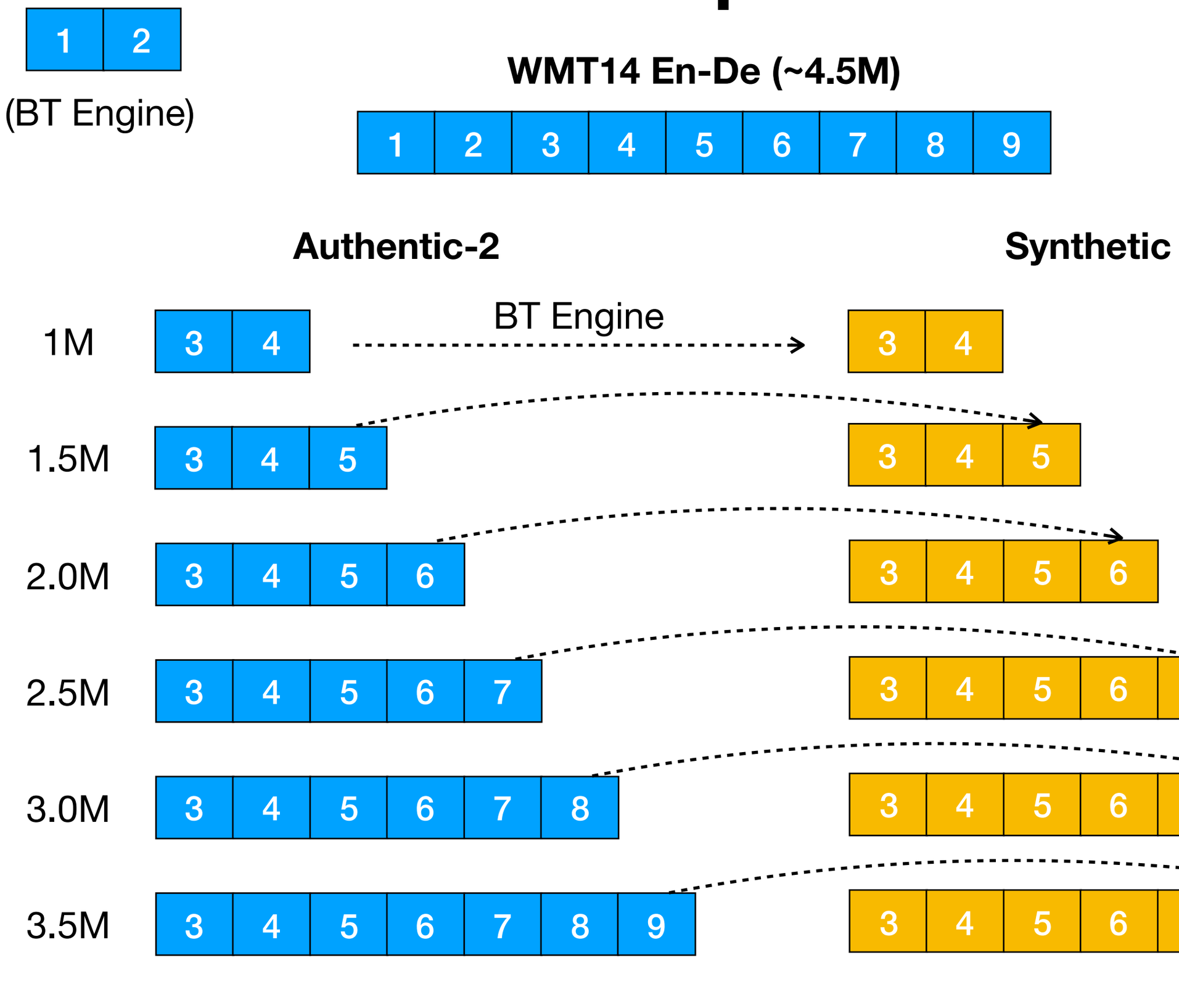
Authentic-1 **Hybrid**

1M 1 2 (BT Engine)





Data Preparation



Results

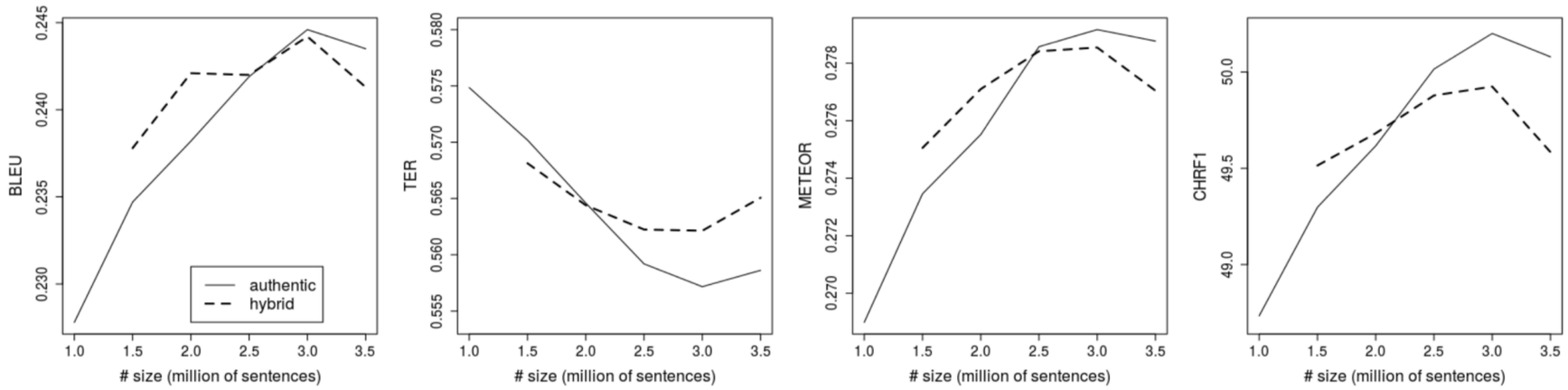


Figure 1: Quality scores of NMT systems trained with different sizes of training data from the $auth_{0+}$ and *hybr* sets.

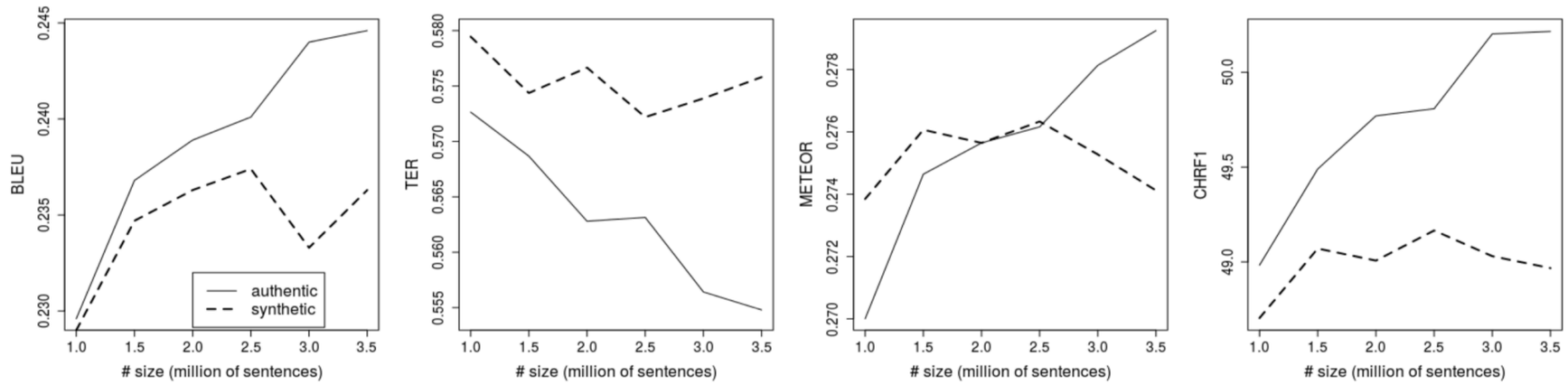


Figure 2: Quality scores of NMT systems trained with different sizes of training data from the $auth_{1+}$ and *synth* sets.

Conclusion:

1. Authentic data: adding more data boosts the performance of the NMT model;
2. Hybrid data: outperforms authentic data at small sizes; degrades when synthetic data exceeds authentic data involved (2M).
3. Synthetic data: performs close to authentic data; no explicit rise when adding more synthetic data; learns well the knowledge encoded (also restricted) by the BT engine.

Vocabulary coverage

data size	$auth_{0+}$	*hybr*	$auth_{1+}$	*synthetic*
1M	67.03%	-	66.35%	60.81%
1.5M	67.15%	66.14%	66.44%	60.93%
2M	67.11%	65.10%	66.41%	60.97%
2.5M	67.25%	64.60%	66.36%	61.03%
3M	67.30%	64.15%	66.47%	60.98%
3.5M	67.25%	63.77%	66.55%	61.01%

Table 1: Coverage of the vocabularies (the top-50000 words) on the tokens in the test set.

Conclusion:

1. Authentic data: slowly increases when adding data; covers more tokens in test set;
2. Hybrid data: decreases when adding more synthetic data; pushes out words that are more likely to appear in real parallel data;
3. Synthetic data: no increase when adding more synthetic data; much lower than authentic data; the vocabulary is restricted by the BT engine.

Tagged Back-Translation (WMT 2019)

AlbertoIsaac Caswell, Ciprian Chelba, David Grangier
Google Research

Storyline

Recent work in Neural Machine Translation (NMT) has shown significant quality gains from noised-beam decoding during back-translation, a method to generate synthetic parallel data.

RQ: What is the real role of added synthetic noise? Is it really to diversify the source side? (Result: simply to indicate to the model that the given source is synthetic)

- Noising En-De bitext sources does not seriously impact the translation quality of the transformer-base baseline.
- A simpler alternative to noising techniques, consisting of tagging back-translated source sentences with an extra token.
- More comparison between noised BT and Tagged BT.

Noising and Tagging

Noise type	Example sentence
[no noise]	Raise the child, love the child.
P3BT	child Raise the, love child the.
NoisedBT	Raise child ___ love child, the.
TaggedBT	<BT> Raise the child, love the child.
TaggedNoisedBT	<BT> Raise, the child the ___ love.

Table 1: Examples of the five noising settings examined in this paper

Noise:

- Word dropout
- Word blanking
- 3-constrained permutation

Tag:

- Prepend a <BT> tag before BT output

Noising Parallel Bitext Sources

% noised	SacreBLEU Newstest '12	Newstest '17
0%	22.4	28.1
20%	22.4	27.9
80%	21.5	27.0
100%	21.2	25.6

Table 2: SacreBLEU degradation as a function of the proportion of bitext data that is noised.

Reasoning about the role of noise in BT:

- By itself, noising does not add meaningful signal, otherwise it would improve performance;
- It also does not damage the signal much;
- In the context of back-translation, the noise could therefore signal whether a sentence were back-translated, without significantly degrading performance.

Tagged Back-Translation

a. Results on 24M BT Set

Model	AVG 13-18	2010	2011	2012	2013	2014	2015	2016	2017	2018
Bitext	32.05	24.8	22.6	23.2	26.8	28.5	31.1	34.7	29.1	42.1
BT	33.12	24.7	22.6	23.5	26.8	30.8	30.9	36.1	30.6	43.5
NoisedBT	34.70	26.2	**23.7**	**24.7**	**28.5**	31.3	33.1	37.7	**31.7**	**45.9**
P3BT	34.57	26.1	23.6	24.5	28.1	31.8	33.0	37.4	31.5	45.6
TaggedBT	**34.83**	**26.4**	23.6	24.5	28.1	**32.1**	**33.4**	**37.8**	**31.7**	**45.9**
TaggedNoisedBT	34.52	26.3	23.4	24.6	27.9	31.4	33.1	37.4	**31.7**	45.6
BT alone	31.20	23.5	21.2	22.7	25.2	29.3	29.4	33.7	29.1	40.5
NoisedBT alone	30.28	23.2	21.0	22.1	24.6	28.4	28.2	33.0	28.1	39.4
Noised(BT + Bitext)	32.07	24.2	22.1	23.5	26.2	29.7	30.1	35.1	29.4	41.9
+ Tag on BT	33.53	25.5	22.8	24.5	27.6	30.3	31.9	36.9	30.4	44.1

b. Results on 216M BT Set

Model	AVG 13-18	2010	2011	2012	2013	2014	2015	2016	2017	2018
Edunov et al. (2018)	35.28			25.0	**29.0**	**33.8**	34.4	37.5	**32.4**	44.6
NoisedBT	35.17	**26.7**	24.0	**25.2**	28.6	32.6	33.9	38.0	32.2	45.7
TaggedBT	**35.42**	26.5	**24.2**	**25.2**	28.7	32.8	**34.5**	**38.1**	**32.4**	**46.0**

Table 3: SacreBLEU on Newstest EnDe for different types of noise, with back-translated data either sampled down to 24M or using the full set of 216M sentence pairs.

Conclusions:

- TaggedNoisedBT does not improve over either tagging or noising alone, supporting the conclusion that tagging and noising are not orthogonal signals but rather different means to the same end;
- If noising in fact increases the quality or diversity of the data, NoisedBT alone should outperform BT;
- NoisedBT on bitext with noise, does not work; noise is not suitable for marking BT samples; TaggedBT performs better as expected.

Tagged Back-Translation

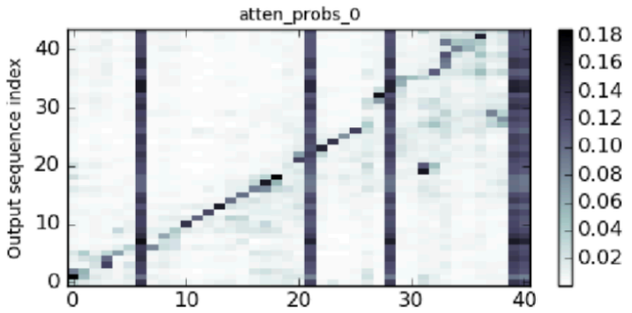
a. Forward models (EnRo)

Model	dev	test
Gehring et al. (2017)		*29.9*
Sennrich 2016 (BT)	*29.3*	*28.1*
bitext	26.5	28.3
BT	31.6	32.6
NoisedBT	29.9	32.0
TaggedBT	30.5	33.0
It.-3 BT	31.3	32.8
It.-3 NoisedBT	31.2	32.6
It.-3 TaggedBT	31.4	**33.4**

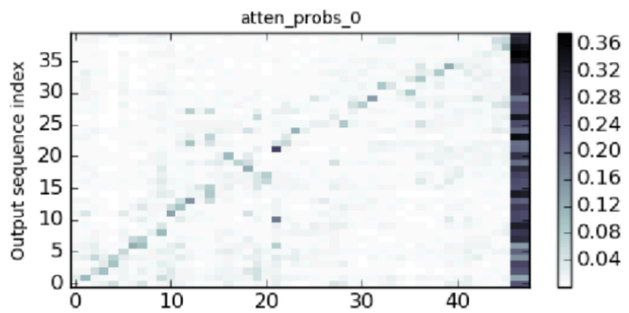
Conclusions:

- NoisedBT is harmful; TaggedBT closes the gap;
- NoisedBT and TaggedBT benefits from iterative BT operation;
- The separation of the synthetic and natural domains allows the model to bootstrap more effectively from the increasing quality of the back-translated data.

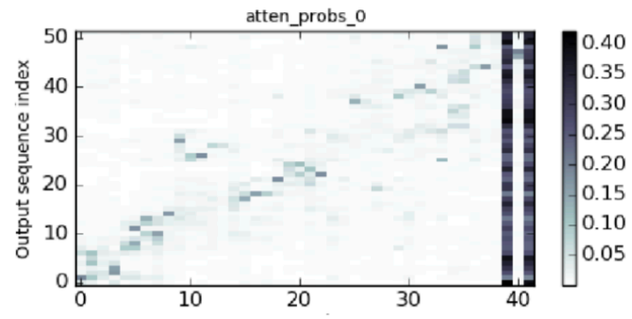
Attention Heatmap



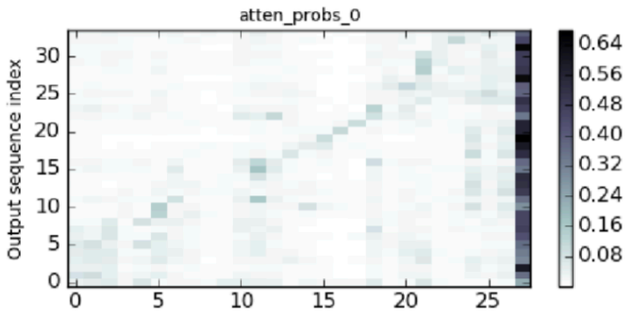
(a) EnDe BT



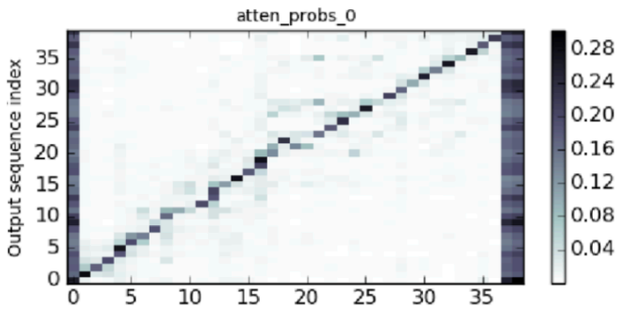
(d) EnRo BT



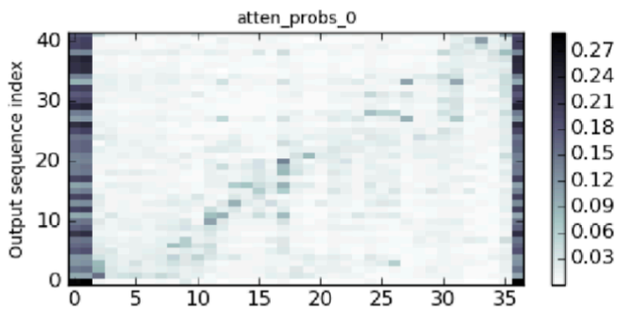
(b) EnDe NoisedBT



(e) EnRo NoisedBT



(c) EnDe TaggedBT



(f) EnRo TaggedBT

Figure 1: Comparison of attention maps at the first encoder layer for a random training example for BT (row 1), NoisedBT (row 2), and TaggedBT (row 3), for both EnDe (col 1) and EnRo (col 2). Note the heavy attention on the tag (position 0 in row 3), and the diffuse attention map learned by the NoiseBT models. These are the models from Table 3.a

Conclusions:

- Heavy attention on tag, indicating the model relies on the information signaled by tag.
- Word-by-word translation in BT, harmful part from BT.
- NoisedBT undoes the word-by-word biases from BT.

Understanding Back-Translation at Scale

Sergey Edunov, Myle Ott, Michael Auli, David Grangier

Facebook AI Research

Presenter: Shilin HE

12/31/19

Motivation and Approach

To broadens the understanding of back-translation (BT), this paper empirically investigated the BT for NMT at a large scale (8*16 GPUs).

Some findings:

1. Sampling or noising beam search outperforms argmax inference
2. Synthetic data can sometimes match the accuracy of real bitext.
3. Others…

Back-Translation

Generating Synthetic Sources:

1. **MAP (Maximum a-posteriori)**:
    1. Beam Search
    2. Greedy
2. **Sampling**:
    1. Unrestricted Sampling
    2. Restricted Sampling: selected top k and then sampling (a middle ground between MAP and Unrestricted Sampling)
3. **Noising beam search:**
    1. delete/swap/replace

	news2013	news2014	news2015	news2016	news2017	Average
bitext	27.84	30.88	31.82	34.98	29.46	31.00
+ beam	27.82	32.33	32.20	35.43	31.11	31.78
+ greedy	27.67	32.55	32.57	35.74	31.25	31.96
+ top10	28.25	33.94	34.00	36.45	32.08	32.94
+ sampling	28.81	34.46	34.87	37.08	32.35	33.51
+ beam+noise	29.28	33.53	33.79	37.89	32.66	33.43

Table 1: Tokenized BLEU on various test sets of WMT English-German when adding 24M synthetic sentence pairs obtained by various generation methods to a 5.2M sentence-pair bitext (cf. Figure 1).

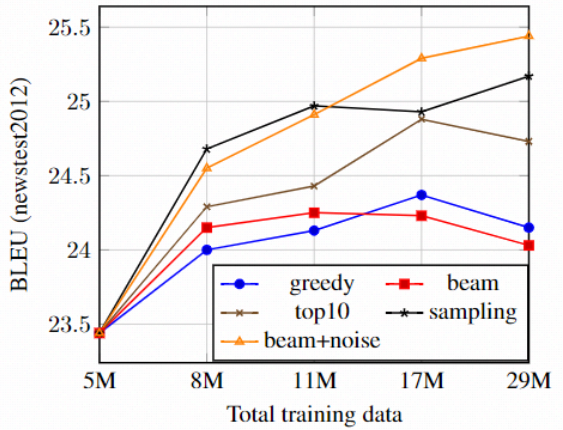


Figure 1: Accuracy of models trained on different amounts of back-translated data obtained with greedy search, beam search ($k = 5$), randomly sampling from the model distribution, restricting sampling over the ten most likely words (top10), and by adding noise to the beam outputs (beam+noise). Results based on newstest2012 of WMT English-German translation.

Findings: Sampling and noising beam search perform significantly better than pure MAP methods

Analysis of Generation methods

Why? Conjecture:

Beam Search/Greedy reduces the diversity and richness of the generated source translations.

In other words,

1. Noisy source sentences make it harder to predict the target translations which may help learning.
2. Sampling provides a richer training signal than argmax, because sampling can better approximate the data distribution.

To verify it,

1) compare the loss on the training data (sampled) by each method:

Training loss is low -> model can easily fit the training data without extracting much learning signal.

Richer training signal

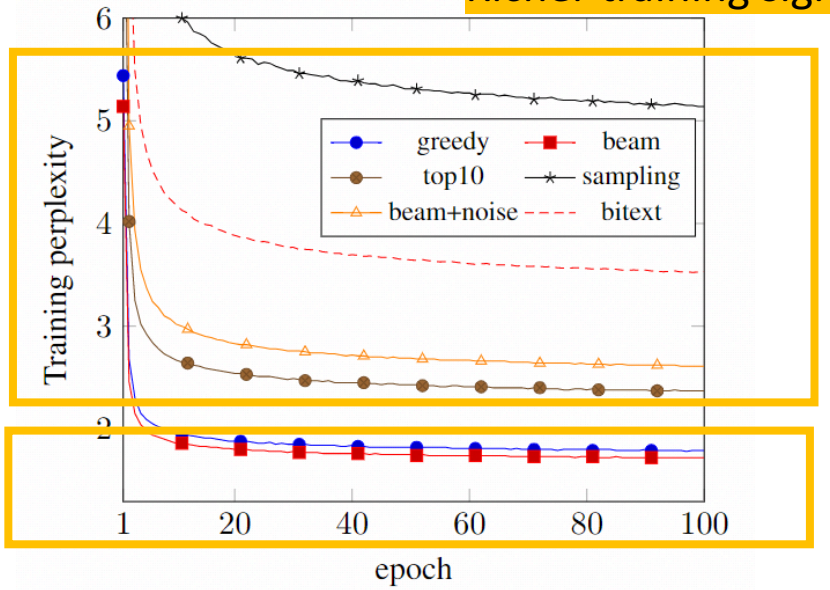


Figure 2: Training perplexity (PPL) per epoch for different synthetic data. We separately report PPL on the synthetic data and the bitext. Bitext PPL is averaged over all generation methods.

Analysis of Generation methods

2) LM score on the generated source data (richness)

Regular data should be more predictable by the LM and has low perplexity.

	Perplexity
human data	75.34
beam	72.42
sampling	500.17
top10	87.15
beam+noise	2823.73

Table 2: Perplexity of source data as assigned by a language model (5-gram Kneser–Ney). Data generated by beam search is most predictable.

source	Diese gegenstzlichen Auffassungen von Fairness liegen nicht nur der politischen Debatte zugrunde.
reference	These competing principles of fairness underlie not only the political debate.
beam	These conflicting interpretations of fairness are not solely based on the political debate.
sample	*Mr President,* these contradictory interpretations of fairness are not based solely on the political debate.
top10	Those conflicting interpretations of fairness are not solely at the heart of the political debate.
beam+noise	conflicting BLANK interpretations BLANK are of not BLANK based on the political debate.

Table 3: Example where sampling produces inadequate outputs. "Mr President," is not in the source. BLANK means that a word has been replaced by a filler token.

Sampling often introduce target words that have no counterpart in the source

Low Resource and Real Bitext

- Sampling is not better than beam in low-resource setting

  Possible reason: low resource back-translations are low-quality while MAP outputs provide more useful training signals

- Synthetic data performs remarkably well (2.2BLEU, 83%), close to the real bitext (2.6BLEU)

- Large scale results on WMT En-De can be 35.0

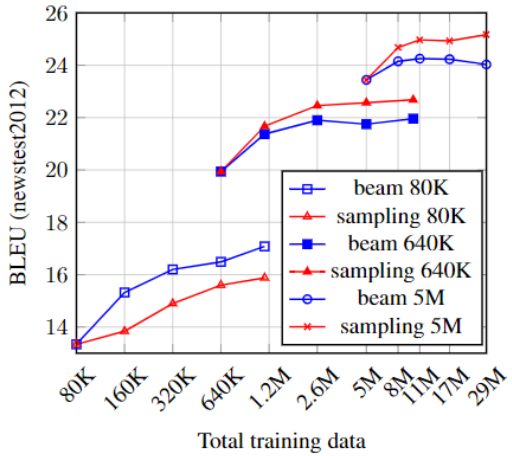


Figure 3: BLEU when adding synthetic data from beam and sampling to bitext systems with 80K, 640K and 5M sentence pairs.

	En–De	En–Fr
a. Gehring et al. (2017)	25.2	40.5
b. Vaswani et al. (2017)	28.4	41.0
c. Ahmed et al. (2017)	28.9	41.4
d. Shaw et al. (2018)	29.2	41.5
DeepL	33.3	**45.9**
Our result	**35.0**	45.6
detok. sacreBLEU[3]	*33.8*	*43.8*

Table 6: BLEU on newstest2014 for WMT English-German (En–De) and English-French (En–Fr). The first four results use only WMT bitext (WMT'14, except for b, c, d in En–De which train on WMT'16). DeepL uses proprietary high-quality bitext and our result relies on back-translation with 226M newscrawl sentences for En–De and 31M for En–Fr. We also show detokenized BLEU (SacreBLEU).

Back-Translation Sampling by Targeting Difficult Words in Neural Machine Translation

Marzieh Fadaee and Christof Monz

University of Amsterdam

Motivation

Back-translation can mitigate the problem of overfitting and fluency. How to select the monolingual data to optimally benefit translation quality?

Analysis Findings:

1. Quality of the synthetic data has a small impact on the effectiveness of back-translation, but the ratio of synthetic to real training data is more important

2. Mostly words that are difficult to predict benefit from the back-translated data

3. words with high pre-diction losses in the target language benefit most from additional back-translated data

4. proposed targeted sampling and specifically targeted words that are difficult to predict

BT settings

- Size

translation quality does not improve linearly with the size of the synthetic data

	Size	2014	2015	2016	2017
Baseline	4.5M	26.7	27.6	32.5	28.1
+ synthetic (1:1)	9M	28.7	29.7	36.3	30.8
+ synthetic (1:4)	23M	29.1	30.0	36.9	31.1
+ synthetic (1:10)	50M	22.8	23.6	29.2	23.9

Table 1: German→English translation quality (BLEU) of systems with different ratios of *real:syn* data.

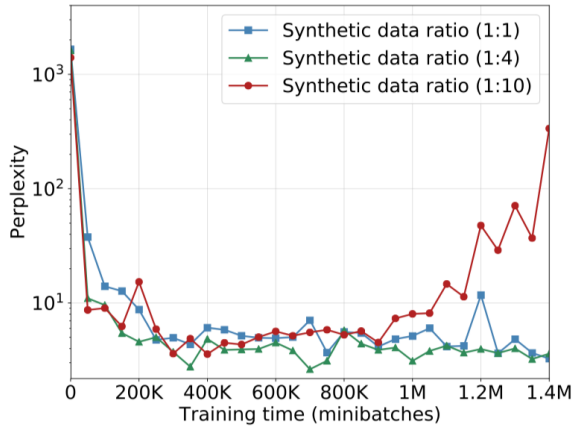


Figure 1: Training plots for systems with different ratios of $(real : syn)$ training data, showing perplexity on development set.

- Direction

Synthetic source is better than synthetic target

	Size	2014	2015	2016	2017
Baseline	4.5M	21.2	23.3	28.0	22.4
+ synthetic tgt	9M	22.4	25.3	29.8	23.7
+ synthetic src	9M	24.0	26.0	30.7	24.8

Table 2: English→German translation quality (BLEU) of systems using forward and reverse models for generating synthetic data.

- Quality

back-translation with a good model achieves results that are close to a system that uses additional manually translated data.

	Size	2014	2015	2016	2017
Baseline	2.25M	24.3	24.9	29.5	25.6
+ synthetic	4.5M	26.0	26.9	32.2	27.5
+ ground truth	4.5M	26.7	27.6	32.5	28.1

Table 3: German→English translation quality (BLEU).

Token Prediction

- Mean prediction loss at token level

Prediction loss increases slightly for most tokens (red)
Decrease dramatically for tokens with large prediction loss (blue)

Difficult to predict tokens

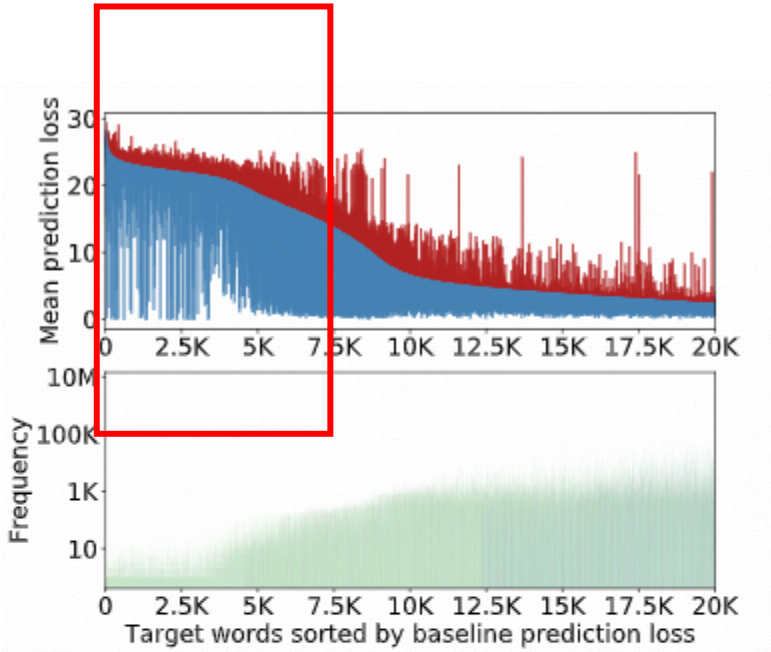


Figure 2: Top: Changes in mean token prediction loss after re-training with synthetic data sorted by mean prediction loss of the baseline system. Decreases and increases in values are marked blue and red, respectively. Bottom: Frequencies (log) of target tokens in the baseline training data.

Next step:
By over-sampling sentences containing difficult-to-predict tokens we can maximize the impact of using the monolingual data

use the target token prediction loss to identify the most rewarding sentences for back-translating and re-training the translation model

Difficult Words

How to determine the “difficult words”?

- Low-frequency tokens
  ⇒ Frequency less than a threshold

- High Mean Prediction Losses
  => Averaged by all occurrence of a token *y*
  Select sentences in monolingual data that contain difficult words

- Skewed Prediction Losses
  ⇒ The above does not discriminate the context, consider the mean and std

- Preserve the sample ratio of difficult occurrences
  ⇒ Consider the sentence occurrence
  y is difficult to predict in 2 contexts, z is difficult to predict in 4 context
  => Sentences contain z should be double the sentences contain y

	De-En				En-De			
System	test2014	test2015	test2016	test2017	test2014	test2015	test2016	test2017
BASELINE†	26.7	27.6	32.5	28.1	21.2	23.3	28.0	22.4
RANDOM†	28.7	29.7	36.3	30.8	24.0	26.0	30.7	24.8
FREQ	29.7	30.5	37.5	31.4	24.2	27.0	31.7	25.2
MEANPREDLOSS†	29.9	**30.9**	**37.8**	**32.1**	**24.7**	26.8	31.5	**25.5**
MEANPREDLOSS + STDPREDLOSS	**30.0**	30.9	37.7	31.9	24.1	26.9	31.0	25.3
PRESERVE PREDLOSS RATIO	29.8	30.9	37.4	31.6	24.5	**27.2**	**31.8**	25.5

Table 4: German↔English translation quality (BLEU). Experiments marked † are averaged over 3 runs. MEANPREDLOSS and FREQ are difficulty criteria based on mean token prediction loss and token frequency respectively. MEANPREDLOSS + STDPREDLOSS is experiments favoring tokens with skewed prediction losses. PRESERVE PREDLOSS RATIO preserves the ratio of the distribution of difficult contexts.

Context-Aware Targeted Sampling

Consider the local context

$$context(S, i) = [S^{i-w}, \ldots, S^{i-1}, S^{i+1}, \ldots, S^{i+w}]$$

where S^j is the token at index j in sentence S.

Compare the context of difficult words in both monolingual data and bitext, evaluate the similarity:

- Matching: word by word, token-level matching
- Word Representation: cosine similarity based on embedding

			De-En				En-De			
System			test2014	test2015	test2016	test2017	test2014	test2015	test2016	test2017
BASELINE †			26.7	27.6	32.5	28.1	21.2	23.3	28.0	22.4
RANDOM †			28.7	29.7	36.3	30.8	24.0	26.0	30.7	24.8
Difficulty criterion	**Context**	**Similarity**								
FREQ	TOKENS	EMB	30.0	30.8	37.6	31.7	24.4	26.3	31.5	25.6
PREDLOSS	SWORDS	MATCH	29.1	30.1	36.9	31.0	23.8	26.2	28.8	23.2
PREDLOSS	TOKENS	MATCH	29.7	30.6	37.6	31.8	24.3	27.4	31.6	25.5
PREDLOSS	TOKENS	EMB	29.9	30.8	37.7	31.9	24.5	**27.5**	31.7	25.6
PREDLOSS	SENTENCE	EMB	24.9	25.5	30.1	26.2	22.0	24.6	27.9	22.5
MEANPREDLOSS	TOKENS	EMB	**30.2**	**31.4**	**37.9**	**32.2**	**24.4**	27.2	**31.8**	**25.6**

Table 8: German↔English translation quality (BLEU). Experiments marked † are averaged over 3 runs. PREDLOSS is the contextual prediction loss and MEANPREDLOSS is the average loss. TOKEN and SWORD are context selection definitions from neighboring tokens and subword units respectively. Note that token includes both subword units and full words. EMB is computing context similarities with token embeddings and MATCH is comparing the context tokens.

- Iterative Back-Translation for Neural Machine Translation WNMT2018
- Exploiting Monolingual Data at Scale for Neural Machine Translation EMNLP2019
- Data Diversification: An Elegant Strategy For Neural Machine Translation arxiv 2019
- ~~On The Evaluation of Machine Translation Systems Trained With Back-Translation arxiv 2019~~

Iterative Back-Translation for Neural Machine Translation

Motivation: a better back-translation system will lead to a better synthetic corpus, hence producing a better final system.

Algorithm 1 Iterative Back-Translation

Input: parallel data D^p, monolingual source, D^s, and target D^t text

Let $T_{\leftarrow} = D^p$
repeat
    Train target-to-source model $\Theta_{\leftarrow}$ on $T_{\leftarrow}$
    Use $\Theta_{\leftarrow}$ to create $S = \{(\hat{s}, t)\}$, for $t \in D^t$
    Let $T_{\rightarrow} = D^p \cup S$
    Train source-to-target model $\Theta_{\rightarrow}$ on $T_{\rightarrow}$
    Use $\Theta_{\rightarrow}$ to create $S' = \{(s, \hat{t})\}$, for $s \in D^s$
    Let $T_{\leftarrow} = D^p \cup S'$
until convergence condition reached

Output: newly-updated models $\Theta_{\leftarrow}$ and $\Theta_{\rightarrow}$

Iterative Back-Translation for Neural Machine Translation

- Impact of Back-Translation Quality

German–English	Back	Final
no back-translation	-	29.6
10k iterations	10.6	29.6 (+0.0)
100k iterations	21.0	31.1 (+1.5)
convergence	23.7	32.5 (+2.9)

English–German	Back	Final
no back-translation	-	23.7
10k iterations	14.5	23.7 (+0.0)
100k iterations	26.2	25.2 (+1.5)
convergence	29.1	25.9 (+2.2)

Iterative Back-Translation for Neural Machine Translation

- Iterative Back-Translation

German–English	Back*	Shallow	Deep	Ensemble
back-translation	23.7	32.5	35.0	35.6
re-back-translation	27.9	33.6	36.1	36.5
Best WMT 2017	-	-	-	35.1

English–German	Back*	Shallow	Deep	Ensemble
back-translation	29.1	25.9	28.3	28.8
re-back-translation	34.8	27.0	29.0	29.3
Best WMT 2017	-	-	-	28.3

Exploiting Monolingual Data at Scale for Neural Machine Translation

- Motivation:
- While target-side monolingual data has been proven to be very useful to improve neural machine translation (briefly, NMT) through back translation, source-side monolingual data is not well investigated.

Exploiting Monolingual Data at Scale for Neural Machine Translation

- We propose a simple yet effective strategy to leverage two-side monolingual data for NMT, which consists of three steps:
- **Step-1: Preparation.**

$$\bar{\mathcal{B}}_s = \{(x, f_b(x))|x \in \mathcal{M}_x\},$$
$$\bar{\mathcal{B}}_t = \{(g_b(y), y)|y \in \mathcal{M}_y\},$$

- **Step-2: Large-scale noised training.**

$$\bar{\mathcal{B}}_s^n = \{(\sigma(x), y)|(x, y) \in \bar{\mathcal{B}}_s\},$$
$$\bar{\mathcal{B}}_t^n = \{(\sigma(x), y)|(x, y) \in \bar{\mathcal{B}}_t\},$$

- **Step-3: Clean data tuning.**

$$\min \sum_{(x,y)\in\mathcal{B}\cup\bar{\mathcal{B}}_s^s\cup\bar{\mathcal{B}}_t^s} -\log P(y|x; f),$$

Exploiting Monolingual Data at Scale for Neural Machine Translation

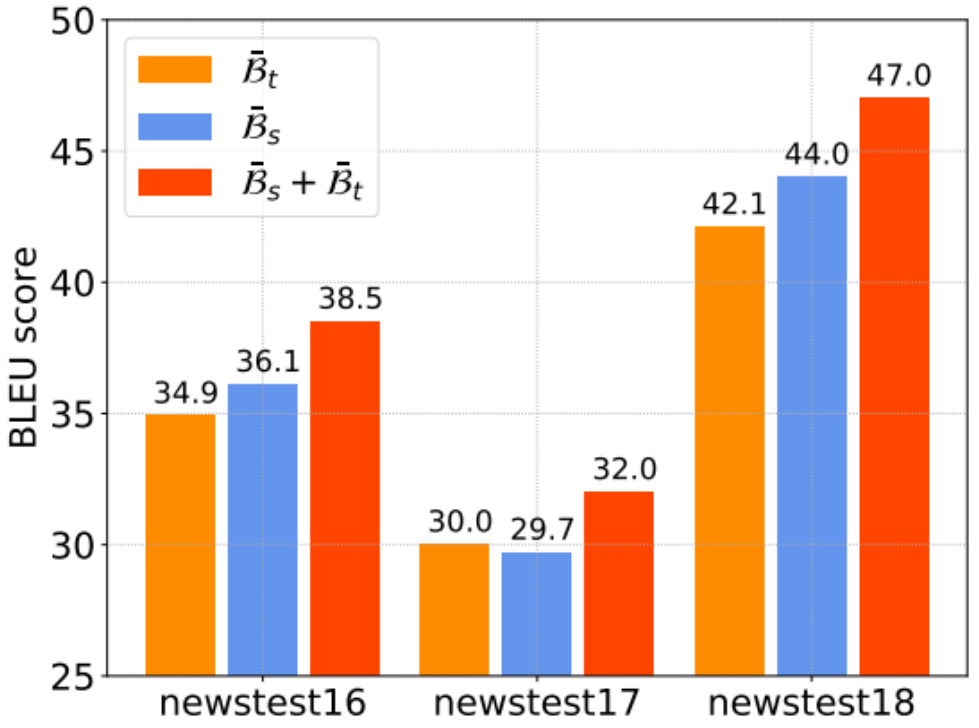
- Experimental results：

	En→De					De→En				
Model	**2016**	**2017**	**2018**	**2019**	**Avg**	**2016**	**2017**	**2018**	**2019**	**Avg**
WMT	34.0	28.0	41.3	37.3	35.15	38.6	34.3	41.1	34.5	37.13
WMTPC	37.1	30.5	45.6	40.3	38.38	41.9	37.5	45.4	40.1	41.23
+Noised Training	39.3	32.0	47.5	41.2	40.00	46.1	39.8	47.7	40.2	43.45
+Clean Tuning	**40.9**	**32.9**	**49.2**	**43.8**	**41.70**	**47.5**	**41.0**	**49.5**	**41.9**	**44.98**
WMTPC+BT	38.7	31.8	46.0	39.8	39.08	45.8	39.8	47.2	38.6	42.90

Table 1: De-tokenized case-sensitive SacreBLEU on WMT En↔De newstest2016, newstest2017, newstest2018, newstest2019 and the average score. "Avg" means the average BLEU score. "+" is conducted upon WMTPC dataset.

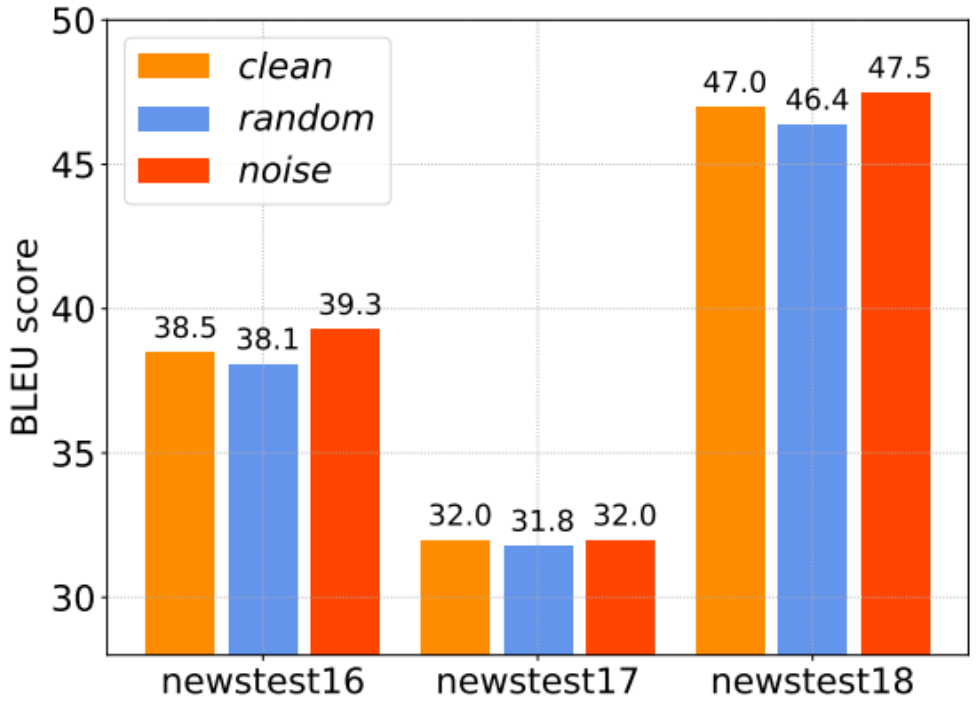
Exploiting Monolingual Data at Scale for Neural Machine Translation

- Source or Target Monolingual Data



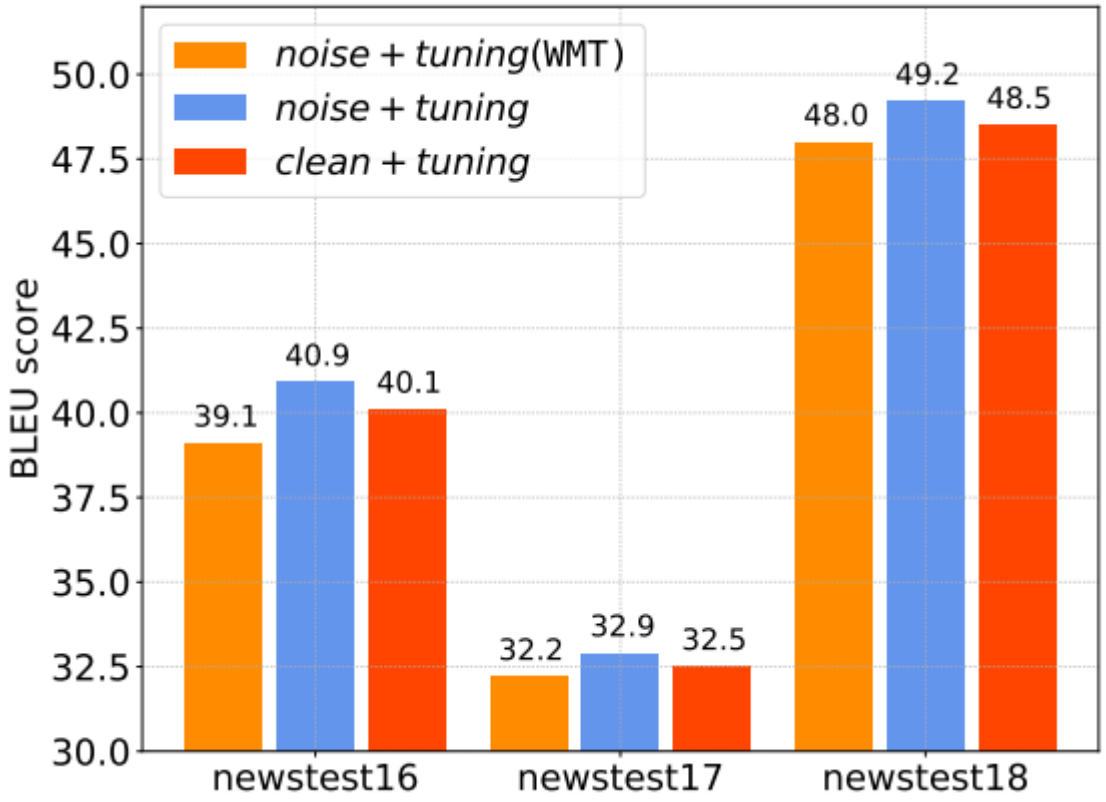
Exploiting Monolingual Data at Scale for Neural Machine Translation

- Synthetic Data Generation



Exploiting Monolingual Data at Scale for Neural Machine Translation

- Synthetic Tuning



Exploiting Monolingual Data at Scale for Neural Machine Translation

- We propose a simple yet effective strategy to leverage two-side monolingual data for NMT, which consists of three steps:
- **Step-1: Preparation.**

$$\bar{\mathcal{B}}_s = \{(x, f_b(x))|x \in \mathcal{M}_x\},$$
$$\bar{\mathcal{B}}_t = \{(g_b(y), y)|y \in \mathcal{M}_y\},$$

- **Step-2: Large-scale noised training.**

$$\bar{\mathcal{B}}_s^n = \{(\sigma(x), y)|(x, y) \in \bar{\mathcal{B}}_s\},$$
$$\bar{\mathcal{B}}_t^n = \{(\sigma(x), y)|(x, y) \in \bar{\mathcal{B}}_t\},$$

- **Step-3: Clean data tuning.**

$$\min \sum_{(x,y)\in\mathcal{B}\cup\bar{\mathcal{B}}_s^s\cup\bar{\mathcal{B}}_t^s} -\log P(y|x; f),$$

Diversification: An Elegant Strategy For Neural Machine Translation

- Motivation:
- … is resort to huge extra monolingual data to conduct semi-supervised training, like back-translation. But extra monolingual data is not always available, especially for low resource languages.

Diversification: An Elegant Strategy For Neural Machine Translation

- Data Diversification
  - the forward models are used to translate the source-side original corpus *S* to synthetic target-side corpora as:

$$T_1^1 = M_{S\to T,1}^1(S), ..., T_1^k = M_{S\to T,1}^k(S)$$

  - the backward models are used to translate the target-side original corpus *T* to synthetic source-side corpora as:

$$S_1^1 = M_{T\to S,1}^1(T), ..., S_1^k = M_{T\to S,1}^k(T)$$

  - we augment the original data with the newly generated synthetic data

$$\mathcal{D}_1 = (S,T) + \cup_{i=1}^k (S,T_1^i) + \cup_{i=1}^k (S_1^i,T)$$

Diversification: An Elegant Strategy For Neural Machine Translation

- Experimental results:

Method	WMT'14 En-De
Transformer (Vaswani et al., 2017)	28.4
Transformer + Rel. Pos (Shaw et al., 2018)	29.2
Scale Transformer (Ott et al., 2018)	29.3
Dynamic Convolution (Wu et al., 2019)	29.7
Our Data Diversification with	
Scale Transformer (Ott et al., 2018)	**30.7**

Method	IWSLT			
	En-De	En-Fr	De-En	Fr-En
No Data Diversification				
Transformer	28.6	44.0	34.7	43.3
Dynamic Conv	28.7	43.8	35.0	43.5
Our Data Diversification				
Transformer	**30.4**	**45.3**	**36.8**	**44.5**

Table 3: Performances in BLEU scores on IWSLT'14 English-German, German-English, and IWSLT'13 English-French and French-English translation tasks.

Diversification: An Elegant Strategy For Neural Machine Translation

- Hypothesis
- Experiments and Explanation

5.4 Forward-Translation is Important

Hypothesis *Forward-translation is as vital as back-translation.*

Experiments and Explanation We separate our method into forward and backward diversification, in which we only train the final model ($\hat{M}_{S\to T}$) with the original data augmented by either the translations of the forward models ($M^i_{S\to T,n}$) or those of the backward models ($M^i_{T\to S,n}$) separately. We compare those variants with the bidirectionally diversified model and the single-model baseline. Experiments were conducted on the IWSLT'14 English-German and German-English tasks.

As shown in Table 6, both the forward and backward diversification methods perform worse than the bidirectional counterpart but still better than the baseline. However, it is interesting that diversification with forward models outperforms the ones with backward models as recent research has focused mainly on back-translation, where they use a backward model to translate target monolingual data to source language (Sennrich et al., 2016a;

Diversification: An Elegant Strategy For Neural Machine Translation

- Ensemble Effects:
- **Hypothesis:** *Data diversification exhibits a strong correlation with ensemble of models*.

	$\|\Theta\|$/FLOPS	IWSLT'14		WMT'14
		En-De	De-En	En-De
Baseline	1x	28.6	34.7	29.3
Ensemble	**7x**	30.2	36.5	30.3
Ours	1x	**30.4**	**36.8**	**30.7**

Table 5: Data diversification preserves the effects of ensembling, while it keeps the number of parameters constant.

Diversification: An Elegant Strategy For Neural Machine Translation

- Perplexity vs. BLEU Score
- **Hypothesis** *Data diversification sacrifices perplexity for better BLEU score*.

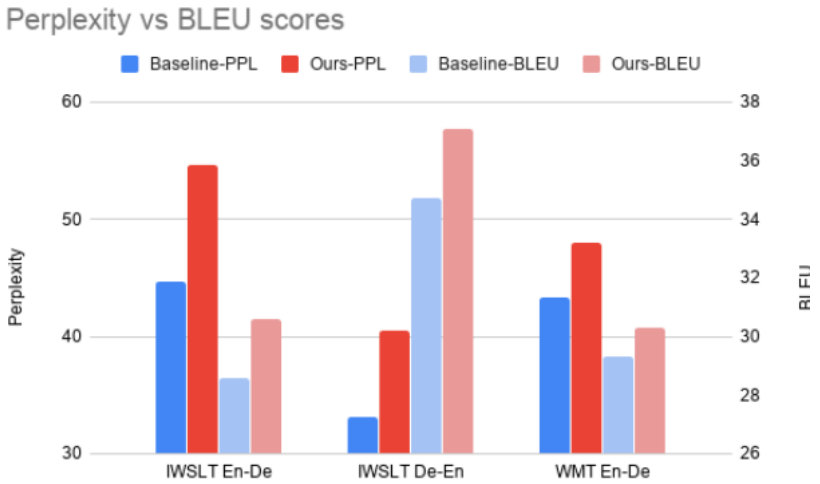


Figure 1: Relationship between validation perplexity and the BLEU scores for the Transformer baseline and data diversification, in the IWSLT'14 English-German, German-English and WMT'14 English-German tasks.

Diversification: An Elegant Strategy For Neural Machine Translation

- Initial Parameters vs. Diversity
- **Hypothesis** *Models with different initial parameters increase diversity in the augmented data, while the ones with fixed initial parameters decrease it.*

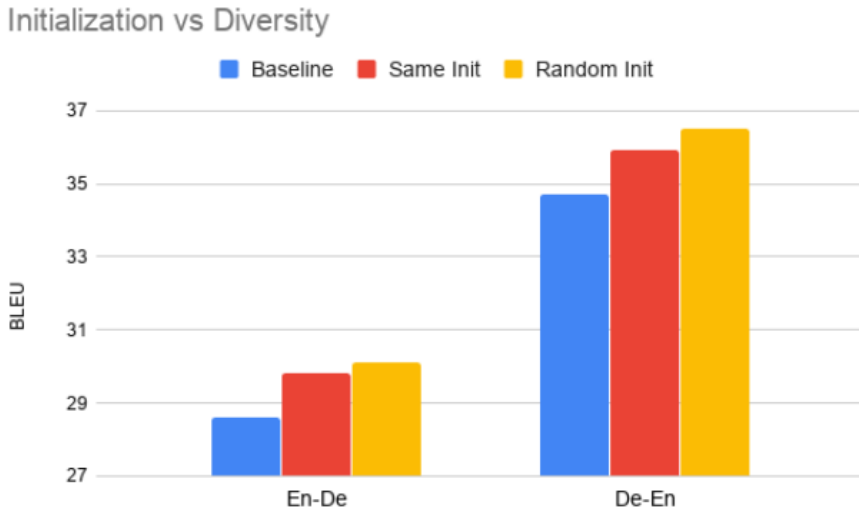


Figure 2: Effects of random and fixed parameters initialization on the performance on the IWSLT'14 English-German and German-English translation tasks.

Diversification: An Elegant Strategy For Neural Machine Translation

- Forward-Translation is Important
  Hypothesis *Forward-translation is as vital as back-translation*.

Task	Baseline	Backward	Forward	Bidirectional
En-De	28.6	29.2	29.86	30.4
De-En	34.7	35.8	35.94	36.8

Table 6: The performance of forward and backward diversification in comparison to bidirectional diversification and the baseline in the IWSLT'14 English-German and German-English tasks.

Diversification: An Elegant Strategy For Neural Machine Translation

- Data Diversification
  - the forward models are used to translate the source-side original corpus *S* to synthetic target-side corpora as:

$$T_1^1 = M_{S\to T,1}^1(S), ..., T_1^k = M_{S\to T,1}^k(S)$$

  - the backward models are used to translate the target-side original corpus *T* to synthetic source-side corpora as:

$$S_1^1 = M_{T\to S,1}^1(T), ..., S_1^k = M_{T\to S,1}^k(T)$$

  - we augment the original data with the newly generated synthetic data

$$\mathcal{D}_1 = (S,T) + \cup_{i=1}^k (S,T_1^i) + \cup_{i=1}^k (S_1^i,T)$$