Semantically Consistent Data Augmentation for Neural Machine Translation via Conditional Masked Language Model

Qiao Cheng, Jin Huang, Yitao Duan
NetEase Youdao
Beijing, China
{chengqiao, huangjin, duan}@rd.netease.com

Abstract

This paper introduces a new data augmentation method for neural machine translation that can enforce stronger semantic consistency both within and across languages. Our method is based on Conditional Masked Language Model (CMLM) which is bi-directional and can be conditional on both left and right context, as well as the label. We demonstrate that CMLM is a good technique for generating context-dependent word distributions. In particular, we show that CMLM is capable of enforcing semantic consistency by conditioning on *both* source and target during substitution. In addition, to enhance diversity, we incorporate the idea of *soft* word substitution for data augmentation which replaces a word with a probabilistic distribution over the vocabulary. Experiments on four translation datasets of different scales show that the overall solution results in more realistic data augmentation and better translation quality. Our approach consistently achieves the best performance in comparison with strong and recent works and yields improvements of up to 1.90 BLEU points over the baseline. [1]

1 Introduction

Neural network models have achieved remarkable results in many fields such as computer vision, natural language processing, and speech. In order to obtain adequate expressivity, the models usually come with a large number of parameters. However, such models are prone to overfitting if trained with an insufficient amount of training data. Data Augmentation (DA) is an effective technique that has been used in many areas to augment existing labeled data and boost the performance of machine learning models. For example, in computer vision, training data is often augmented by ways such as horizontal flipping, random cropping, tilting, and color shifting (Krizhevsky et al., 2012; Cubuk et al., 2018). While DA has become a standard technique to train deep networks for image processing, it is relatively under-explored in Natural Language Processing (NLP).

The exact mechanisms and theoretical foundations of data augmentation are still under investigation. Most studies show empirically that data augmentation is effective and provide some intuitive explanations. A recent work in the field of vision (Gontijo-Lopes et al., 2020) demonstrates that affinity (the distributional shift caused by DA) and diversity (the complexity of the augmentation) can predict the performance of data augmentation methods. However, neither metric can be measured without completing the entire DA process. Therefore, it is still challenging to evaluate the goodness of a DA technique without full-fledged experimentation and it is not clear how the result can be used to guide the design of data augmentation schemes.

Generally speaking, DA can be classified into two categories. The first tries to produce realistic samples that resemble the inherent semantics of naturally generated data. In areas such as computer vision, this is often achieved via heuristics that mimic the intrinsic processes that could have actually happened in the physical world, such as photometric noise, flipping, and scaling, etc. The second perturbs the data in a stochastic fashion, resulting in unrealistic samples. Some (e.g., Bishop (1995)) interpret this as a type of regularization that boosts model performance by reducing overfitting. Both are being exploited in NLP.

This paper focuses on lexical replacement methods that augment the training data by altering existing sentences in the parallel corpus of a neural machine translation (NMT) system. We have observed frequently in practice, as well as in literature (Gao et al., 2019; Fadaee et al., 2017; Kobayashi, 2018; Wu et al., 2019; Dong et al., 2021; Liu et al., 2021), that augmented data samples that preserve

[1] Our code is available at https://github.com/netease-youdao/cmlm_da.

Proceedings of the 29th International Conference on Computational Linguistics, pages 5148–5157
October 12–17, 2022.

the semantics of the real labeled data increase the effective training size and are beneficial for model performance. We call this property *semantic consistency*. In the case of NMT, the training data comes in the form of a collection of <source, target> sentence pairs where source is a sentence in the source language and target its translation in the target language. Semantic consistency requires that (1) both source and target are fluent and grammatically correct in their respective languages; and (2) target is a high quality translation of source.

German	Es ist ja ganz **angenehm**, in eine kleine Klasse zu kommen.
English	You know, it's very **pleasant** to walk into a small class.
Case 1	You know, it's very **please** to walk into a small class.
Case 2	You know, it's very **uncomfortable** to walk into a small class.
Case 3	You know, it's very **enjoyable/comfortable** to walk into a small class.

Table 1: Data augmentation examples with varying degrees of semantic consistency.

Existing methods augment the training data using word swapping, removal or substitution (Artetxe et al., 2017; Lample et al., 2017) on either source or target, or both. Due to the discrete nature of language, these transformations are not always semantic-preserving. Quite often they either weaken the fluency of source or/and target, or break their relationships. To illustrate, consider the example given in Table 1 that shows a sentence pair from an English-German parallel corpus. Case 1 to 3 are three synthetic English sentences generated by some DA processes. Both Case 1 and 2 are undesirable because the former, although substituting the word **pleasant** with a word close in meaning, is grammatically incorrect, whereas the latter is not a good translation of the German sentence. Case 3, on the other hand, is a good augmentation that satisfies the two requirements of semantic consistency.

1.1 Our Contributions

To achieve better augmentation, the generation process must make better use of context and label. In this paper, we introduce Conditional Masked Language Model (CMLM) (Wu et al., 2019; Ghazvininejad et al., 2019; Chen et al., 2020) to data augmentation for NMT. A Masked Language Model can make use of both left and right context, and a CMLM is an enhanced version that can be conditional on more information. CMLM has been used successfully in tasks such as text classification (Wu et al., 2019). However, to the best of our knowledge, its application to text generation, especially using deep bidirectional models such as BERT (Devlin et al., 2019), has not been explored. We demonstrate in this paper that CMLM is a good technique for generating context-dependent word distributions. In particular, we show that CMLM is capable of enforcing semantic consistency by conditioning on *both* source and target during substitution. In addition, to enhance diversity, we combine the *soft* word substitution approach for DA, which replaces a word with a probabilistic distribution over the vocabulary (Gao et al., 2019). Experiments on four translation datasets of different scales show that the overall solution results in more realistic data augmentation and better translation quality. Our approach consistently achieves the best performance in comparison with strong and recent works and yields improvements of up to 1.90 BLEU points over the baseline.

In addition, we introduce an unsupervised method to measure semantic consistency without full-fledged training of NMT models, which may take many days even on GPU clusters. This could be used to provide an efficient early assessment of a data augmentation scheme.

2 Related Work

From a technical perspective, previous work on data augmentation for NLP can be classified as either context-independent or context-dependent. Context-independent approaches often apply predetermined, easy-to-compute transformations that depend solely on the word or sentence to be altered. Not surprisingly, most of them are not semantically consistent. Wei and Zou (2019) improves performance on many text classification tasks through a set of word level random perturbation operations, including random insertion, deletion, and swapping. Similar ideas have been applied to NMT, but the methods differ in how and what to alter. *Swap* (Artetxe et al., 2017; Lample et al., 2017) randomly swaps words in nearby positions within a window size k and *Drop* (Iyyer et al., 2015;

Lample et al., 2017) randomly drops word tokens. *Blank* (Xie et al., 2017) replaces the candidate word with a placeholder token and *Smooth* (Xie et al., 2017) replaces it with a word sampled from the frequency distribution of the vocabulary, showing that data noising is an effective regularizer for NMT. *SwitchOut*, introduced in (Wang et al., 2018), formulates the design of a DA algorithm as an optimization problem that maximizes an objective that encourages two desired properties: smoothness and diversity. *SwitchOut* independently replaces words in both `source` and `target` by other words uniformly sampled from their respective vocabularies. Others try to preserve a certain level of semantic consistency by replacing words with their synonyms selected from a handcrafted ontology such as WordNet (Zhang et al., 2015) or words based on similarity calculation (Wang and Yang, 2015).

These works do not make use of important context and label information and, in practice, usually cause a very small or even negative impact on performance. Context-dependent approaches, on the other hand, modify words, phrases, or the whole sentence based on their contextual information that is usually modeled using neural networks. We summarize a few representative ones below.

Fadaee et al. (2017) propose a simple but effective approach to augment the training data of NMT for low-resource language pairs. Their work uses shallow LSTM language models (LM) trained on large amounts of monolingual data to first substitute a word in `source`, and then put the corresponding translation in `target`, using automatic word alignments and the traditional statistical MT practice. LM_{sample} (Kobayashi, 2018) proposes contextual augmentation for text classification by offering a wide range of substitute words, which are predicted by a label-conditional bidirectional language model. Wu et al. (2019) retrofit BERT to conditional BERT that allows it to augment sentences without breaking the label-compatibility. The BERT-based solution brings two benefits. First, BERT's Transformer core provides a more structured memory for handling long-term dependencies in text. Second, BERT, as a deep bidirectional model, is strictly more powerful than the shallow concatenation of left-to-right and right-to-left models.

A recent work (Liu et al., 2021) treats a translation language model as a causal model and performs data augmentation by counterfactual-based causal inference. Their DA replaces source phrases according to a masked language model and the aligned target phrase by a cross-lingual language model (XLM) (Conneau and Lample, 2019) conditional on the changed source phrase.

Different from their work, we use two separate CMLMs to augment source and target respectively, which means that, instead of model prediction, the condition is always true information for both CMLMs. We show its superiority in section 5.1.

The way we incorporate augmented data into the NMT training is drawn from the idea of "soft" word introduced by *SCA* (Gao et al., 2019). Basically, the embedding of a chosen word in a sentence is replaced by its probabilistic distribution predicted by a language model. This brings in more diversity to the DA process. However, Gao et al. (2019) as a DA solution is based on a uni-directional language model and is *not* label-conditional. As we show in section 4.3 that this is less optimal.

3 Approach

In this section, we present our method in detail. We first introduce conditional MLM, then we show how to apply CMLM to data augmentation in neural machine translation tasks.

Let (X, Y) be a pair of source and target sentences where $X = (x_1, x_2, \ldots, x_M)$ and $Y = (y_1, y_2, \ldots, y_N)$ are two sequences of tokens in source and target languages, with lengths M and N, respectively. A neural machine translation system learns the conditional probability $p(y_1, y_2, \ldots, y_N | x_1, x_2, \ldots, x_M)$.

3.1 Conditional MLM

Recall that our goal is to augment NMT's parallel corpus with synthesized data that preserves the semantics within source and target sentences, as well as their cross-lingual relations. To this end, we resort to Conditional MLM for generating context-dependent word distributions, with which we then find the best substitutes for a given word. CMLM is a variation of MLM, which allows further fine-tuning of the pre-trained model. It makes the strong assumption that the masked tokens are conditionally independent of each other given the context and predicts the probabilities individually (Ghazvininejad et al., 2019).

In our case, we apply the following two practices when instantiating our CMLMs:

- We condition the CMLM on *both* X and Y.

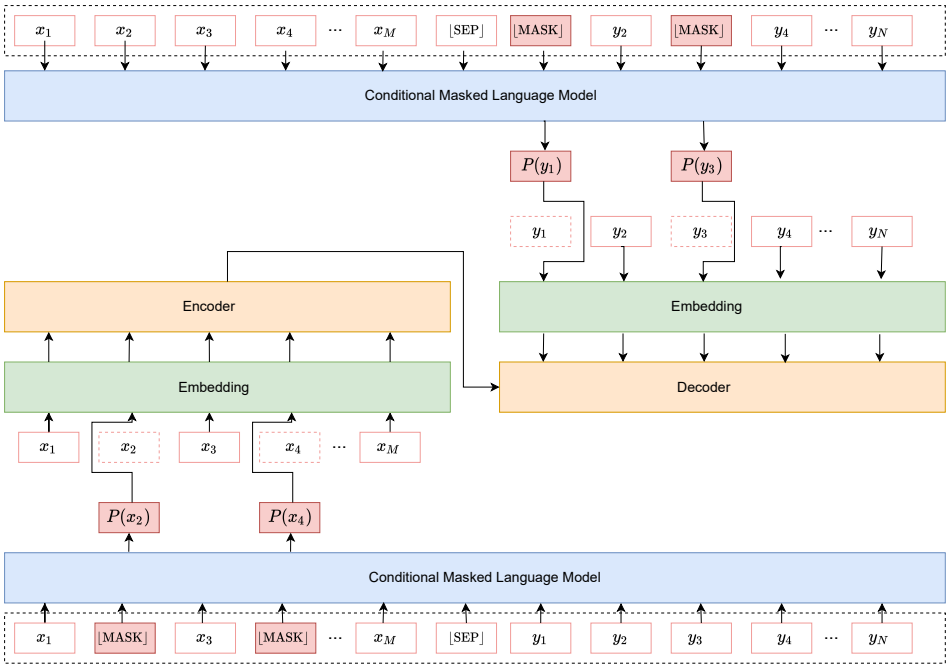

Figure 1: The architecture of our CMLM-based soft contextual data augmentation approach.

- During the training of a CMLM, we only mask out tokens in either X, or Y, but not both.

We call this approach "**Conditioning on Both but Predicting One**", referring to how it treats the source and target sides in the NMT training. Specifically, for each sentence pair (X, Y), we first concatenate X and Y, then randomly mask 15% of the words in X, and then train a CMLM to predict the masked words:

$$P\left(x_1^m, \ldots, x_i^m \mid X^u, Y\right) \quad (1)$$

where x_i^m denotes a masked token and X^u the unmasked ones within X. For the tokens in the target sentence, we train a *separate* CMLM to get their distribution similarly:

$$P\left(y_1^m, \ldots, y_i^m \mid X, Y^u\right) \quad (2)$$

During the training of an NMT model, both X and Y are available. Conditioning on the reference sentence Y allows the model to enforce stronger consistency between input and label, resulting in meaningful translations when applied to DA in NMT. We show in section 5.1, using metrics developed for translation quality estimation, that this choice significantly improves the translation quality of the generated sentence pairs.

Changing X or Y but not both for DA is a deliberate choice. Typical modern languages have diverse vocabularies, with synonyms and semantically equivalent or close expressions. This already provides abundant opportunities for semantic-preserving transformations. Therefore, it is not necessary to alter X and Y simultaneously. In section 5.1, we compare our choice with an XLM (cross-lingual language model) (Conneau and Lample, 2019) approach which changes X and Y simultaneously. The empirical study shows that our approach can avoid introducing incorrect `<source, target>` pairs and improve NMT performance.

3.2 Soft Conditional Contextual DA

Once a CMLM is trained, one could use it to expand training data for NMT. This is typically done by replacing words with others predicted by the language model at the corresponding positions (e.g., Kobayashi (2018); Wu et al. (2019)). In our case, since the probability distribution of the masked words $P\left(x_1^m, \ldots, x_i^m \mid X^u, Y\right)$, or $P\left(y_1^m, \ldots, y_i^m \mid X, Y^u\right)$ if we mask out words in Y, contains information from both backward and forward contexts, as well as target sentence, sampling from such distribution could potentially generate better substitutions for the word on the masked position. However, such a method could be expensive: to generate enough samples with adequate variation, exponentially many candidates have to be processed.

Instead, inspired by Gao et al. (2019), we take

a *soft* approach. In essence, this method works directly with the word embeddings and uses the expectation of a word's embedding over the CMLM's output distribution to replace its original embedding. Let w be a candidate word and $P(w)$ its distribution defined by the CMLM. Note that $P(w)$ is conditional on the context that we described earlier and is over the entire vocabulary. Suppose E is the embedding matrix of all the $|V|$ words. We use E_W to denote the embedding vector of a word W. The embedding of the soft word w is:

$$e_w = \mathbb{E}_{W \sim P(w)}[E_W] = \sum_{j=0}^{|V|} p_j(w) E_j \quad (3)$$

3.3 NMT Training with DA

In this section, we elaborate on the training process of the NMT model with our DA method. Figure 1 shows the architecture of the scheme. There are two independently trained CMLMs, one for augmenting the encoder, and the other the decoder. The two CMLMs can be turned on/off independently and we study the effects in section 5.2.

We use BERT (Devlin et al., 2019) as our CMLM, for its deep bidirectional natural, and superior capability for handling long-term dependencies. We start by taking a pre-trained multilingual BERT, and fine-tune it using the method described in 3.1. The NMT training proceeds as usual, except that, at each sentence pair (X, Y), for each word in X (or Y), with probability γ we replace its embedding by its soft version defined by Equation 3. Notice that, our method does *not* generate any data explicitly. Rather, we use embedding substitution to incorporate augmentation directly into the training process. We study the effect of different values of γ in section 5.3.

4 Experiments

In this section, we demonstrate the effectiveness of our method on four datasets with diverse language variation. They include three relatively small-scale datasets, {German, Spanish, Hebrew} to English ({De, Es, He}-> En) from the well-known IWSLT 2014, and one large-scale English to German (En->De) dataset from WMT14 .

4.1 Data

For IWSLT14 De->En task we follow the same pre-processing steps and the same train/dev/test split as in Gao et al. (2019). The training dataset and validation dataset contains about 160K and 7K sentence pairs, respectively. Consistent with previous work, tst2010, tst2011, tst2012, dev2010, and dev2012 are concatenated as our test data. For Es->En and He->En tasks, there are 181K and 151K parallel sentence pairs in each training set. We validate on tst2013 and test on tst2014 for these two tasks. For all IWSLT translation tasks, we use a joint source and target vocabulary with 10K byte-pair-encoding (BPE) (Sennrich et al., 2016) types. For the WMT2014 En-De translation task, again, we follow Gao et al. (2019) to filter out 4.5M sentence pairs for training. We concatenate newstest2012 and newstest2013 as the validation set and use newstest2014 as the test set. We use a joint source and target vocabulary built upon the BPE with 40k sub-word types. For fair comparison to previous work, we report tokenized BLEU (Papineni et al., 2002) scores computed with the *multi-bleu.perl* script from Moses.[2] To further boost comparability, we also report detokenized BLEU scores computed using sacreBLEU (Post, 2018). (Post, 2018). For all experiments, we performed significance tests based on bootstrap resampling introduced by Koehn (2004).

4.2 Model

Our model uses the Transformer architecture, which is solely based on attention mechanisms and dominates most of the sequence-to-sequence tasks. For all IWSLT tasks, we adopt the transformer_iwslt_de_en configuration for the NMT model. Specifically, both the encoder and decoder consist of 6 blocks, and the source and target word embedding are shared for the language pair. The dimensions of embedding and feed-forward sub-layer are set to 512 and 1024, respectively. The number of attention heads is set to 4. The default dropout rate is 0.3. For WMT14 En-De, we use the default transformer_big configuration for the NMT model. Specifically, the dimensions of embedding and feed-forward sub-layer are 1024 and 4096, respectively. The NMT models are trained by Adam (Kingma and Ba, 2015) optimizer with default learning rate schedule as Vaswani et al. (2017).

For all tasks, we adopt the BERT-base configuration for the CMLM model, except that the number of hidden layers is set to 4 to speed up the train-

[2] https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

	IWSLT			WMT
	De->En	**Es->En**	**He->En**	**En->De**
Other Reported Results				
Base*	34.79	41.58	33.64	28.40
*Swap**	34.70	41.60	34.25	28.13
*Drop**	35.13	41.62	34.29	28.29
*Blank**	35.37	42.28	34.37	28.89
*Smooth**	35.45	41.69	34.61	28.97
LM_{sample}*	35.40	42.09	34.31	28.73
*SCA**	**35.78**	**42.61**	**34.91**	**29.70**
mixSeq†	**35.78**	41.39	-	29.61
Our Implementations				
Base	34.37	41.67	33.76	28.25
CMLM_{hard}	35.76	42.25	34.66	30.01
CMLM_{soft}	**35.93(+1.56)**	**42.92(+1.25)**	**35.21(+1.45)**	**30.15(+1.9)**

Table 2: BLEU scores over the test sets. (∗) from Gao et al. (2019). (†) from Wu et al. (2021)

	IWSLT			WMT
	De->En	**Es->En**	**He->En**	**En->De**
Base	33.62	40.87	33.15	27.49
CMLM_{hard}	35.07	41.45	34.01	29.08
CMLM_{soft}	**35.31(+1.69)**	**42.01(+1.14)**	**34.51(+1.36)**	**29.37(+1.88)**

Table 3: SacreBLEU scores over the test sets.

ing process. We use the bottom 4 layers of the pre-trained BERT-base-multilingual-cased model as the starting point of CMLM fine-tuning. We also experiment with an entirely randomly-initialized CMLM model and find that the pre-trained weights result in faster CMLM training. We follow Devlin et al. (2019) for the CMLM fine-tuning and use a triangular learning rate schedule with maximum learning rate η. The CMLM parameters are also updated with the Adam optimizer.

4.3 Main Results

We compare our method against several other strong data augmentation methods, including several context-independent approaches such as *Swap* (Artetxe et al., 2017; Lample et al., 2017), *Drop* (Iyyer et al., 2015; Lample et al., 2017), *Blank* (Xie et al., 2017) and *Smooth* (Xie et al., 2017), and two context-dependent ones, LM_{sample} (Kobayashi, 2018) and *SCA* (Gao et al., 2019). We also compare it against a sentence-level augmentation method, mixSeq (Wu et al., 2021), which randomly selects two sentence pairs, concatenates the source sentences and the target sentences, respectively, with a special label <sep> separating two samples, and trains the model on such augmented dataset.

Our baseline is the vanilla transformer described earlier without DA. For comparison, we performed two sets of data augmentation experiments using CMLM: (1) CMLM_{soft} uses the soft approach described in section 3.2 and follows the training framework in section 3.3. (2) CMLM_{hard} uses the conventional hard substitution approach, with the substitution words generated by sampling from the CMLMs. Both CMLM_{soft} and CMLM_{hard} augment both the encoder and the decoder, and use the same mask probability $\gamma = 0.25$, which we find to be the optimal configuration. See sections 5.2 and 5.3.

The BLEU and SacreBLEU scores on four translation tasks are presented in Table 2 and 3, respectively. Both CMLM_{soft} and CMLM_{hard} are superior to the base system, with CMLM_{soft} consistently achieves the best performance on all tasks and across all comparisons. The CMLM (soft) approach significantly outperformed the baseline in Table 2 and Table 3 for all four tasks, with p-values lower than 0.02. Most remarkably, our DA improves the baseline by as much as 1.90 BLEU points on the WMT14 En->De dataset.

In addition to experiments on publicly available corpora, we also evaluate the scheme on Youdao's

production NMT engine, [3] a major multilingual neural machine translation service that is trained with data at least three orders of magnitudes larger than the public corpora. The method achieves similar consistent improvements. Our DA mechanism has been built into the production NMT engine, serving billions of requests each day.

5 Analysis

Our method consists of multiple modules, and we design several groups of comparative experiments to analyze their effects.

5.1 Semantic Consistency

Recall that the "soft" substitution approach that we use works directly with embeddings and does not generate synthetic data explicitly. The quality of the DA process depends on the distributions defined by the two CMLMs (equations 1 and 2). There is no straightforward metric to measure the distributions in terms of semantic consistency. Here we propose a simple sampling-based approach. The intuition is: if the distribution is used for text generation, the quality of resulting sentence pairs is a good indicator of the effectiveness of its role in the DA process.

Specifically, given a sentence pair (X, Y), we randomly replace some tokens from X (resp. Y) with those sampled from the source (resp. target) CMLM, resulting in (X', Y) (resp. (X, Y')). We manually inspect a small sample and find that our method indeed produces sentence pairs that are generally both fluent in their respective languages and correct in terms of translation quality. However, our goal is to have an automatic method that can be used to assess semantic consistency at large scale. To this end, we draw on the research in Quality Estimation (QE) for Machine Translation. Self-Supervised QE aims to evaluate the quality of machine-translated sentences without human labeling, which aligns perfectly with our goal.

Zheng et al. (2021) show that the conditional probability computed by the CMLM in Equation 2 is a good indicator of translation quality (which also implies fluency). Specifically, let y^m be a word in the target, the translation quality score of this word is defined as $P\left(y^m \mid X, Y^u\right)$ as computed by the CMLM. The sentence-level quality score is simply averaging the quality scores over all target words.

[3] https://fanyi.youdao.com/

Our case is slightly different. Since we have both X and Y, we can use the idea of Zheng et al. (2021) but with a more direct approach: we can compare the words in X' (resp. Y') against the original ones in X (resp. Y) and compute the accuracy. This is equivalent to taking expectations over the test sentences.

	Source Acc	Target Acc	BLEU
MLM	53.5%	44.0%	35.56
XLM	74.8%	70.4%	35.65
CMLM	80.1%	75.5%	35.93

Table 4: The prediction accuracy of source and target, and BLEU for IWSLT14 German-English translation.

We compare our CMLM-based approach against the DA results from (1) an XLM-based scheme in Liu et al. (2021), which alters both X and Y by treating a translation language model as a causal model and performing data augmentation by counterfactual-based causal inference; and (2) a simple MLM which does not condition on any portion of Y. All implementations use models with the same configuration as the CMLM described in section 4.2, fine-tuned with the same training data but their individual conditions and objectives. Table 4 shows the prediction accuracy of masked words on the 7K IWSLT14 German-English validation data set. Consistent with the mask probability during CMLM training, we let the model predict 15% of the words in X or Y. For ease of comparing the final effects on the machine translation task, Table 4 also shows the BLEU scores measured on IWSLT14 German-English dataset after applying the DA method to the NMT engine.

Our CMLM-based solution achieves strong prediction accuracy rates of 80.1% and 75.5% on source and target sides, respectively, significantly outperforming the MLM approach by near 30 percentage points. This shows that our method is capable of generating synthetic sentence pairs with much better translation quality. The improvement over XLM is milder but still significant, with 5+ percentage points. BLEU scores follow a similar trend. Recall that we use independent CMLMs to alter either X or Y but not both, while XLM uses a single cross-lingual language model to change both. The results confirm our conjecture that altering both X and Y simultaneously while preserving semantic consistency may be too difficult for the language models. Doing so may introduce too

much noise and hurt translation quality.

This method also provides an efficient way to assess a data augmentation scheme for NMT. It can save days or even months of GPU time (for training NMT models) since computing the word prediction accuracy rates on a few thousands of sentence pairs is very fast.

5.2 Encoder vs. Decoder

Our CMLM-based data augmentation method can be applied to either encoder or decoder, or both. In this section, we conduct experiments to study the effects of these choices. We train two CMLMs independently. The first is used to augment the encoder, the latter the decoder. Note that, per our discussion in section 3.1, the CMLMs, when activated, only augment one side of the sentence pair. The encoder (resp. decoder) CMLM mask out words **only** in X (resp. Y), thus replacing their embeddings by their soft versions.

Table 5 shows the BLEU scores for different augmentation configurations. It is clear that both encoder and decoder augmentations are beneficial, with encoder augmentation obtaining slightly more gain. The maximum improvement can be achieved when the method is applied to both.

	IWSLT			**WMT**
	De-En	Es-En	He-En	En-De
Base	34.37	41.67	33.76	28.25
+Encoder	35.23	42.31	34.66	29.57
+Decoder	34.93	42.13	34.41	29.34
+Both	35.93	42.92	35.21	30.15

Table 5: BLEU scores over the test sets.

5.3 Mask Probability

As mentioned in section 3.2, for each word in X or Y, we replace its embedding by its soft version with probability γ. This parameter controls the extent to which the DA method will exert its effect. Intuitively, a small value of γ will preserve the original semantics better while a large value of γ can bring in more diversity. A balance must be struck. We experiment with different values, and Figure 2 shows their influence on BLEU on the IWSLT14 De-En dataset. The strongest performance is reached with a mask probability of 0.25.

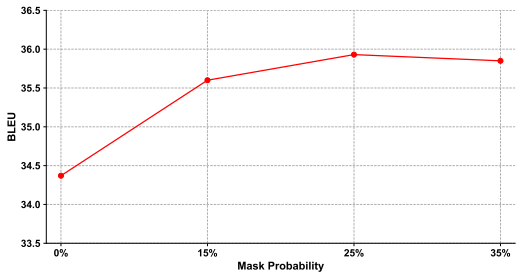

Figure 2: BLEU score for IWSLT14 German-English with difference mask probability.

5.4 Computation Overhead

Our DA method introduces two additional steps into the NMT training process: fine-tuning the CMLMs and augmenting the NMT model. The actual overhead depends on the scale of the data sets. In our experiments, IWSLT De-En and WMT En-De corpora consist of 160K and 4.5M sentence pairs, respectively. Fine-tuning the CMLMs on the two corpora takes about 3 and 20 hours, respectively, on a *single* A40 GPU.

Our training process has the same complexity as that of *SCA* (Gao et al., 2019) so they should have similar computation performance. From our experiments, the training time on IWSLT dataset increases about 84%, up from 2.5 hours to 4.6 hours, again on a *single* A40. The overhead is less significant for large corpora. The WMT tasks take 25% more time to train, up from one day to roughly 32 hours on 4 A40 cards. We see only a 10% increase in training time when we apply the DA method to our production NMT engine.

6 Conclusion

In this paper, we advocate performing semantically consistent data augmentation for neural machine translation and propose a scheme based on Conditional Masked Language Model and soft word substitution. We show that a deep, bi-directional CMLM is capable of enforcing semantic consistency by conditioning on *both* source and target during data augmentation. Experiments demonstrate that the overall solution results in more realistic data augmentation and better translation quality. Our approach consistently achieves the best performance in comparison with strong and recent works and yields improvements of up to 1.90 BLEU points over baseline.

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. 2017. Unsupervised neural machine translation. *arXiv preprint arXiv:1710.11041*.

Chris M. Bishop. 1995. Training with noise is equivalent to tikhonov regularization. *Neural Computation*, 7(1):108–116.

Yen-Chun Chen, Zhe Gan, Yu Cheng, Jingzhou Liu, and Jingjing Liu. 2020. Distilling knowledge learned in bert for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7893–7905.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. *Advances in Neural Information Processing Systems*, 32:7059–7069.

Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. 2018. Autoaugment: Learning augmentation policies from data. *arXiv preprint arXiv:1805.09501*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. 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 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Xin Luna Dong, Yaxin Zhu, Zuohui Fu, Dongkuan Xu, and Gerard de Melo. 2021. Data augmentation with adversarial training for cross-lingual nli. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5158–5167.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. Data augmentation for low-resource neural machine translation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 567–573.

Fei Gao, Jinhua Zhu, Lijun Wu, Yingce Xia, Tao Qin, Xueqi Cheng, Wengang Zhou, and Tie-Yan Liu. 2019. Soft contextual data augmentation for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5539–5544.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. 2019. Mask-predict: Parallel decoding of conditional masked language models. 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 6112–6121, Hong Kong, China. Association for Computational Linguistics.

Raphael Gontijo-Lopes, Sylvia J Smullin, Ekin D Cubuk, and Ethan Dyer. 2020. Affinity and diversity: Quantifying mechanisms of data augmentation. *arXiv preprint arXiv:2002.08973*.

Mohit Iyyer, Varun Manjunatha, Jordan Boyd-Graber, and Hal Daumé III. 2015. Deep unordered composition rivals syntactic methods for text classification. In *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing (volume 1: Long papers)*, pages 1681–1691.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

Sosuke Kobayashi. 2018. Contextual augmentation: Data augmentation by words with paradigmatic relations. 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 452–457.

Philipp Koehn. 2004. Statistical significance tests for machine translation evaluation. In *Proceedings of the 2004 conference on empirical methods in natural language processing*, pages 388–395.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2017. Unsupervised machine translation using monolingual corpora only. *arXiv preprint arXiv:1711.00043*.

Qi Liu, Matt Kusner, and Phil Blunsom. 2021. Counterfactual data augmentation for neural machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 187–197.

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 of the Association for Computational Linguistics*, pages 311–318.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. 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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

William Yang Wang and Diyi Yang. 2015. That's so annoying!!!: A lexical and frame-semantic embedding based data augmentation approach to automatic categorization of annoying behaviors using# petpeeve tweets. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2557–2563.

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.

Jason Wei and Kai Zou. 2019. Eda: Easy data augmentation techniques for boosting performance on text classification tasks. 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 6382–6388.

Xing Wu, Shangwen Lv, Liangjun Zang, Jizhong Han, and Songlin Hu. 2019. Conditional bert contextual augmentation. In *International Conference on Computational Science*, pages 84–95. Springer.

Xueqing Wu, Yingce Xia, Jinhua Zhu, Lijun Wu, Shufang Xie, Yang Fan, and Tao Qin. 2021. mixSeq: A simple data augmentation methodfor neural machine translation. In *Proceedings of the 18th International Conference on Spoken Language Translation (IWSLT 2021)*, pages 192–197, Bangkok, Thailand (online). Association for Computational Linguistics.

Ziang Xie, Sida I Wang, Jiwei Li, Daniel Lévy, Aiming Nie, Dan Jurafsky, and Andrew Y Ng. 2017. Data noising as smoothing in neural network language models. *arXiv preprint arXiv:1703.02573*.

Xiang Zhang, Junbo Zhao, and Yann Lecun. 2015. Character-level convolutional networks for text classification. *Advances in Neural Information Processing Systems*, 2015:649–657.

Yuanhang Zheng, Zhixing Tan, Meng Zhang, Mieradilijiang Maimaiti, Huanbo Luan, Maosong Sun, Qun Liu, and Yang Liu. 2021. Self-supervised quality estimation for machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics.