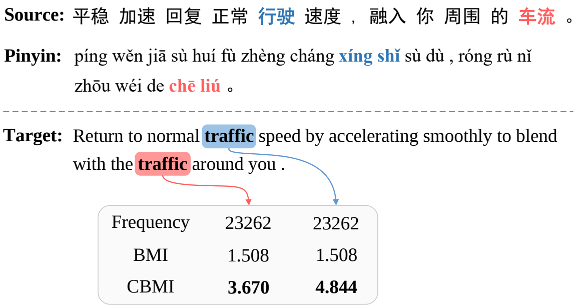
Conditional Bilingual Mutual Information Based Adaptive Training for Neural Machine Translation

Songming Zhang[1]*, **Yijin Liu**[2]*, **Fandong Meng**[2], **Yufeng Chen**[1]†,
Jinan Xu[1], **Jian Liu**[1] and **Jie Zhou**[2]
[1]Beijing Key Lab of Traffic Data Analysis and Mining,
Beijing Jiaotong University, Beijing, China
[2]Pattern Recognition Center, WeChat AI, Tencent Inc, China
{zhangsongming,chenyf,jaxu,jianliu}@bjtu.edu.cn,
{yijinliu,fandongmeng,withtomzhou}@tencent.com

Abstract

Token-level adaptive training approaches can alleviate the token imbalance problem and thus improve neural machine translation, through re-weighting the losses of different target tokens based on specific statistical metrics (*e.g.,* token frequency or mutual information). Given that standard translation models make predictions on the condition of previous target contexts, we argue that the above statistical metrics ignore target context information and may assign inappropriate weights to target tokens. While one possible solution is to directly take target contexts into these statistical metrics, the target-context-aware statistical computing is extremely expensive, and the corresponding storage overhead is unrealistic. To solve the above issues, we propose a target-context-aware metric, named conditional bilingual mutual information (CBMI), which makes it feasible to supplement target context information for statistical metrics. Particularly, our CBMI can be formalized as the log quotient of the translation model probability and language model probability by decomposing the conditional joint distribution. Thus CBMI can be efficiently calculated during model training without any pre-specific statistical calculations and large storage overhead. Furthermore, we propose an effective adaptive training approach based on both the token- and sentence-level CBMI. Experimental results on WMT14 English-German and WMT19 Chinese-English tasks show our approach can significantly outperform the Transformer baseline and other related methods.

1 Introduction



Source: 平稳 加速 回复 正常 行驶 速度 ， 融入 你 周围 的 车流 。

Pinyin: píng wěn jiā sù huí fù zhèng cháng **xíng shǐ** sù dù , róng rù nǐ zhōu wéi de **chē liú** 。

Target: Return to normal **traffic** speed by accelerating smoothly to blend with the **traffic** around you .

Frequency	23262	23262
BMI	1.508	1.508
CBMI	**3.670**	**4.844**

Figure 1: An example from the WMT19 Zh-En training set. Despite the different mappings from the source sentence, existing target-context-free metrics (*i.e.,* frequency and BMI) equally assess the two '*traffic*' tokens, while our CBMI can distinguish the different dependencies of the two tokens on the source sentence with the guidance of target contexts.

Neural machine translation (NMT) (Bahdanau et al., 2014; Gehring et al., 2017; Vaswani et al., 2017; Meng and Zhang, 2019; Liu et al., 2021a,b) has made remarkable achievements in recent years. Generally, NMT models are trained to maximize the likelihood of the next target token given ground-truth tokens as inputs (Johansen and Juselius, 1990; Goodfellow et al., 2016). Due to the token imbalance phenomenon in natural language (Zipf, 1949), for an NMT model, the learning difficulties of different target tokens may be various. However, the vanilla NMT model equally weights the training losses of different target tokens, irrespective of their difficulties.

Recently, various adaptive training approaches (Gu et al., 2020; Xu et al., 2021) have been proposed to alleviate the above problem for NMT. Generally, these approaches re-weight the losses of different target tokens based on specific statistical metrics. For example, Gu et al. (2020) take the token frequency as an indicator and encourage the NMT model to focus more on low-frequency tokens. Xu et al. (2021) further propose the bilingual mutual information (BMI) to measure the word mapping diversity between bilinguals, and down-weight the tokens with relatively lower BMI values.

Despite their achievements, there are still limita-

[*] Equal contribution. Work was done when Songming were interning at Pattern Recognition Center, WeChat AI, Tencent Inc, China.

[†] Yufeng Chen is the corresponding author.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 2377 - 2389
May 22-27, 2022 ©2022 Association for Computational Linguistics

tions in these adaptive training approaches. Given that the standard translation model autoregressively makes predictions on the condition of previous target contexts, we argue that the statistical metrics used in the above approaches ignore target context information and may assign inaccurate weights for target tokens. Specifically, although existing statistical metrics can reflect complex characteristics of target tokens (*e.g.,* mapping diversity), they fail to model how these properties vary across different target contexts. Secondly, for the identical target tokens in different positions of a target sentence (*e.g.,* two '*traffic*' tokens in the Figure 1), they may be mapped from different source-side tokens, but such target-context-free metrics cannot distinguish the above different mappings. In summary, it is necessary to incorporate target context information into the above statistical metrics. One possible solution is to directly take target context information into account and conduct target-context-aware statistical calculations. But in this way, the calculation cost and storage overhead will become huge and unrealistic[1]. Therefore, it is non-trivial to design a suitable target-context-aware statistical metric for adaptive training in the field of NMT.

In this paper, we aim to address the above issues in adaptive training methods. Firstly, we propose a novel target-context-aware metric, named **C**onditional **B**ilingual **M**utual **I**nformation (CBMI), to measure the importance of different target tokens by their dependence on the source sentence. Specifically, we calculate CBMI by the mutual information between a target token and its source sentence on the condition of its target contexts. With the aid of target-context-aware calculations, CBMI can easily model the various characteristics of target tokens under different target contexts, and of course can distinguish identical target tokens with different source mappings. Regarding the computational efficiency, through decomposing the conditional joint distribution in the aforementioned mutual information, our CBMI can be formalized as the log quotient of the translation model probability and language model probability[2]. Therefore, CBMI can be efficiently calculated during model training without any pre-specific statistical calculations and huge storage overhead, which makes it feasible to supplement target context information for statistical metrics. Subsequently, we design an adaptive training approach based on both the token- and sentence-level CBMI, which dynamically re-weights the training losses of the corresponding target tokens.

We evaluate our approach on the WMT14 English-German and WMT19 Chinese-English translation tasks. Experimental results on both datasets demonstrate that our approach can significantly outperform the Transformer baseline and other adaptive training methods. Further analyses reveal that CBMI can also reflect the adequacy of translation, and our CBMI-based adaptive training can improve translation adequacy meanwhile maintain fluency. The main contributions of this paper can be summarized as follows:

- We propose a novel target-context-aware metric, named CBMI, which can reflect the importance of target tokens for NMT models. Theoretical analysis and experimental results show that CBMI is computationally efficient, which makes it feasible to complement target context information in statistical metrics.
- We further propose an adaptive training approach based on both the token- and sentence-level CMBI, which dynamically re-weights the training losses of target tokens.
- Further analyses show that CBMI can also reflect the adequacy of translation, and CBMI-based adaptive training can improve translation adequacy meanwhile maintain fluency[3].

2 Background

2.1 Neural Machine Translation

An NMT model is designed to translate a source sentence with M tokens $\mathbf{x} = \{x_1, x_2, \ldots, x_M\}$ into a target sentence with N tokens $\mathbf{y} = \{y_1, y_2, \ldots, y_N\}$ by predicting the probability of each target token:

$$P(\mathbf{y}|\mathbf{x};\theta) = \prod_{j=1}^{N} p(y_j|\mathbf{y}_{<j}, \mathbf{x};\theta) \qquad (1)$$

where j is the index of each time step, $\mathbf{y}_{<j}$ is the target-side previous context for y_j, and θ is the model parameter.

[1] Take the vanilla BMI (Xu et al., 2021) as an example, to process the raw WMT14 En-De training data (about 1.5GB), it takes about 12 CPU hours and 2GB disk storage to save the BMI values. To make matters worse, the cost will increase dozens of times in target-context-aware statistical calculations.

[2] The detailed derivation process is shown in Equation (7). Please note that the language model is only used during training and thus does not affect the inference speed.

[3] The code is publicly available at: https://github.com/songmzhang/CBMI.

During training, NMT models are generally optimized with the cross-entropy (CE) loss:

$$\mathcal{L}_{\text{CE}}(\theta) = -\sum_{j=1}^{N} \log p(y_j|\mathbf{y}_{<j}, \mathbf{x}; \theta) \quad (2)$$

During inference, NMT models predict the probabilities of target tokens in an auto-regressive mode and generate hypotheses using heuristic search algorithms like beam search (Reddy, 1977).

2.2 Token-level Adaptive Training for NMT

Token-level adaptive training aims to alleviate the token imbalance problem for NMT models by re-weighting the training losses of target tokens. How to design a suitable weight adjustment strategy matters, which is we aim to improve in this paper. Formally, for the j-th target token and its adaptive weight w_j, the standard cross-entropy loss in Equation (2) is expanded to the following formula:

$$\mathcal{L}_{\text{ada}}(\theta) = -\sum_{j=1}^{N} w_j \log p(y_j|\mathbf{y}_{<j}, \mathbf{x}; \theta) \quad (3)$$

2.3 Mutual Information for NMT

Mutual information (MI) is a general metric in information theory (Shannon, 1948), which measures the mutual dependence between two random variables a and b as follows[4]:

$$\text{MI}(a; b) = \log \left(\frac{p(a, b)}{p(a) \cdot p(b)} \right) \quad (4)$$

Xu et al. (2021) propose token-level bilingual mutual information (BMI) to measure the word mapping diversity between bilinguals and further conduct BMI-based adaptive training for NMT. The BMI is formulated as:

$$\text{BMI}(\mathbf{x}; y_j) = \sum_{i=1}^{|\mathbf{x}|} \log \left(\frac{f(x_i, y_j)}{f(x_i) \cdot f(y_j)} \right) \quad (5)$$

where $f(\cdot)$ is an word frequency counter. Although BMI can reflect the bilingual mapping properties to some extent, it cannot correspondingly vary with the target context. However, simply introducing target-context-aware calculations into BMI would make the above statistical calculations unrealistic.

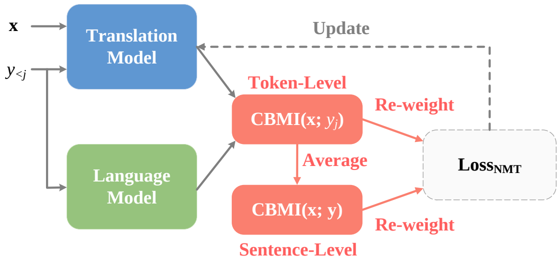


Figure 2: Overview of the training process of our method. For the target token y_j, we calculate its token-level CBMI by the translation model and the language model, and average all the token-level CBMI values in a sentence into the sentence-level CBMI. Then the two CBMI values with different granularities are combined to form the final training loss weight of the token y_j.

3 Approaches

In this section, we first introduce the definition of CBMI (Section 3.1). Then, we illustrate how to adjust the weights for the training losses of target tokens based on the token- and the sentence-level CBMI (Section 3.2). Figure 2 shows the overall training process of our approach.

3.1 Definition of CBMI

As mentioned above, it is necessary to incorporate target context information into the statistical metrics (*e.g.,* BMI) for adaptive training. However, it is impractical to directly conduct target-context-aware statistical computations due to the expensive computational costs and storage overhead. In this paper, we propose a new target-context-aware metric, named conditional bilingual mutual information (CBMI), to solve the above issues. Specifically, CBMI is calculated by the mutual information between each target token and its source sentence under the condition of previous target context. Formally, the CBMI of a target token y_j and its source sentence $\mathbf{x}$ is calculated as follow:

$$\begin{aligned} \text{CBMI}(\mathbf{x}; y_j) &= \text{MI}\left(\mathbf{x}; y_j|\mathbf{y}_{<j}\right) \\ &= \log \left(\frac{p(y_j, \mathbf{x}|\mathbf{y}_{<j})}{p(y_j|\mathbf{y}_{<j}) \cdot p(\mathbf{x}|\mathbf{y}_{<j})} \right) \end{aligned} \quad (6)$$

The original CBMI definition presented in the above equation still struggles in computation, thus we further simplify it by decomposing the condi-

[4] We use the point-wise MI here instead of the original expectation form, since we aim to calculate the mutual information between individual samples in this paper.

tional joint distribution:

$$\begin{aligned}\mathrm{CBMI}(\mathbf{x};y_j) &= \log\left(\frac{p(y_j,\mathbf{x}|\mathbf{y}_{<j})}{p(y_j|\mathbf{y}_{<j})\cdot p(\mathbf{x}|\mathbf{y}_{<j})}\right)\\ &= \log\left(\frac{p(y_j|\mathbf{x},\mathbf{y}_{<j})\cdot p(\mathbf{x}|\mathbf{y}_{<j})}{p(y_j|\mathbf{y}_{<j})\cdot p(\mathbf{x}|\mathbf{y}_{<j})}\right)\\ &= \log\left(\frac{p(y_j|\mathbf{x},\mathbf{y}_{<j})}{p(y_j|\mathbf{y}_{<j})}\right)\\ &= \log\left(\frac{p_{\mathrm{NMT}}(y_j)}{p_{\mathrm{LM}}(y_j)}\right)\end{aligned} \tag{7}$$

where $p_{\mathrm{NMT}}(y_j)$ is the probability output by the NMT model, and $p_{\mathrm{LM}}(y_j)$ is the probability output by an additional target-side language model (LM). In this way, we formalize the complex target-context-aware calculation in Equation (6) as the log quotient of the NMT probability and LM probability. Based on the simplified Equation (7), CBMI can be computed in real time during the model training, thus enabling both target-context-aware and efficient computations. Considering the massive computation required by existing methods to perform the target-context-aware calculation, the LM in our CBMI only brings a modest computational cost in training and finally leads to better performance. We will give a detailed comparison of the calculation cost and storage overhead between our CBMI and existing approaches in Section 5.2.

3.2 CBMI-based Weight Adjustment

According to the definition, CBMI measures the mutual dependence between a target token and its corresponding source sentence on the condition of its context. Namely, target tokens with larger CBMI value rely more on the source-side information and less on the target historical translations, which is exactly in line with the goal of the adequacy translation model. Given that current NMT models tend to generate fluent but inadequate translations (Weng et al., 2020; Miao et al., 2021), we speculate that making the NMT models pay more attention to target tokens with larger CBMI values can improve translation adequacy and thus improve translation performance. Furthermore, we observe a phenomenon that if target sentences contain many words with small CBMI values, they generally do not match well with the corresponding source sentences. To alleviate the negative effect of these poorly matched sentence pairs, we average all the token-level CBMI values in a target sentence into a sentence-level CBMI and incorporate it into our approach. Consequently, we propose to dynamically adjust the training weight of each target token based on both the token- and sentence-level CBMI. For clarity, we use t to mark the 'token-level' intermediate variables and s to mark the 'sentence-level' ones in the following formulas.

Token-Level CBMI. The token-level CBMI can reflect the importance of target tokens for improving translation adequacy (*i.e.*, dependency of the source side information). Thus we amplify the weights of target tokens with larger token-level CBMI to make the NMT model pay more attention to them. Particularly, to reduce the variances and stabilize the distribution of the token-level CBMI in each target sentence, we firstly conduct intra-sentence normalization for the token-level CBMI $\mathrm{CBMI}^t(\mathbf{x};y_j)$:

$$\mathrm{CBMI}^t_{norm}(\mathbf{x};y_j) = (\mathrm{CBMI}^t(\mathbf{x};y_j) - \mu^t)/\sigma^t \tag{8}$$

where μ^t, σ^t represent the mean values and the standard deviations of $\mathrm{CBMI}^t(\mathbf{x};y_j)$ in each target sentence.

Then we scale the normalized CBMI value $\mathrm{CBMI}^t_{norm}(\mathbf{x};y_j)$ to obtain the token-level training weight for y_j:

$$w^t_j = \max\{0, scale^t\cdot\mathrm{CBMI}^t_{norm}(\mathbf{x};y_j)+1\} \tag{9}$$

where $scale^t$ is a hyperparameter that controls the effect of $\mathrm{CBMI}^t_{norm}(\mathbf{x};y_j)$.

Sentence-level CBMI. We average all the token-level CBMI values in a target sentence to form the sentence-level CBMI, which can further reflect the matching degree between the bilingual sentences in a sentence pair. To alleviate the negative effect of poorly matched sentence pairs and encourage the NMT model focus on well-matched sentences pairs, we up-weight the sentence pairs with larger sentence-level CBMI values and down-weight those sentence pairs with smaller sentence-level CBMI values. Specifically, the sentence-level CBMI between the source sentence $\mathbf{x}$ and the target sentence $\mathbf{y}$ can be derived from Equation (4) and represented as the arithmetic average of token-level CBMI values[5]:

[5] We divide the original sentence CBMI with its corresponding sentence length to reduce its variance.

$$
\begin{aligned}
\mathrm{CBMI}^s(\mathbf{x};\mathbf{y}) &= \frac{1}{|\mathbf{y}|}\log\left(\frac{p(\mathbf{x},\mathbf{y})}{p(\mathbf{x})\cdot p(\mathbf{y})}\right) \\
&= \frac{1}{|\mathbf{y}|}\log\left(\frac{p(\mathbf{y}|\mathbf{x})}{p(\mathbf{y})}\right) \\
&= \frac{1}{|\mathbf{y}|}\log\left(\frac{\prod_j p(y_j|\mathbf{x},\mathbf{y}_{<j})}{\prod_j p(y_j|\mathbf{y}_{<j})}\right) \\
&= \frac{1}{|\mathbf{y}|}\sum_j \log\left(\frac{p(y_j|\mathbf{x},\mathbf{y}_{<j})}{p(y_j|\mathbf{y}_{<j})}\right) \\
&= \frac{1}{|\mathbf{y}|}\sum_j \mathrm{CBMI}^t(\mathbf{x};y_j)
\end{aligned} \tag{10}
$$

Similarly, we conduct inter-sentence normalization for $\mathrm{CBMI}^s(\mathbf{x};\mathbf{y})$:

$$\mathrm{CBMI}^s_{norm}(\mathbf{x};\mathbf{y}) = (\mathrm{CBMI}^s(\mathbf{x};\mathbf{y}) - \mu^s)/\sigma^s \tag{11}$$

where μ^s, σ^s represent the mean values and the standard deviations of $\mathrm{CBMI}^s(\mathbf{x};\mathbf{y})$ in each mini-batch during training.

Subsequently, we also scale $\mathrm{CBMI}^s_{norm}(\mathbf{x};\mathbf{y})$ in Equation (11) with another hyperparameter $scale^s$ to obtain the sentence-level training weight:

$$w^s = \max\{0, scale^s \cdot \mathrm{CBMI}^s_{norm}(\mathbf{x};\mathbf{y}) + 1\} \tag{12}$$

Final Loss Weight. In our adaptive training approach, for the target token y_j, its final loss weight w_j in Equation (3) is the multiplication of the above two weights in Equation (9) and (12):

$$w_j = w^t_j \cdot w^s \tag{13}$$

4 Experiments

4.1 Datasets

We conduct experiments on two large-scale WMT tasks, *i.e.,* the WMT14 English to German (En-De) and WMT19 Chinese to English (Zh-En). For the En-De task, the training set contains 4.5M sentence pairs. The validation set and test set are newstest2013 and newstest2014, respectively. For the Zh-En task, the training set totally contains 20M sentence pairs and the validation set and test set are newstest2018 and newstest2019, respectively. Following previous work, we share the vocabulary for the En-De task and segment words into subwords using byte pair encoding (BPE) (Sennrich et al., 2016) with 32k merge operations for both datasets.

4.2 Implementation Details

Training. We implement baselines and our approach under Transformer$_{base}$ and Transformer$_{big}$ settings based on the open-source toolkit fairseq (Ott et al., 2019) with mixed precision (Ott et al., 2018). We train all the translation models with the cross-entropy loss for 100k steps, and further finetune them with different adaptive training objectives for another 200k steps on both tasks. The target-side language model is a Transformer decoder without the cross-attention modules, which is trained synchronously with the translation model. The training data for the language model is the target-side monolingual data from the NMT training set. All the experiments are conducted on 8 NVIDIA Tesla V100 GPUs, and each batch on each GPU contains approximately 4096 tokens. We use Adam optimizer (Kingma and Ba, 2014) with 4000 warmup steps to optimize models. More training details are listed in Appendix B.

In our experiments, we have not been able to bring further improvement to our approach through simply enhancing the language model. Our conjecture is that stronger language models will generate sharper distribution, and will increase the variances of CBMI values when used as the denominator, resulting in detriment for NMT model training. We will leave this for the future work.

Evaluation. During inference, we set beam size to 4 and length penalty to 0.6 for both tasks. We use *multibleu.perl* to calculate case-sensitive BLEU for WMT14 En-De and *SacreBLEU*[6] to calculate case-sensitive BLEU for WMT19 Zh-En. We use the paired bootstrap resampling methods (Koehn, 2004) for the statistical significance test.

4.3 Hyperparameter Experiments.

In this section, we introduce the hyperparameter settings of our approach according to the performance on the validation set of the WMT14 En-De dataset, and we share the same hyperparameter settings with the WMT19 Zh-En dataset.

Scale Setting. The two hyperparameter $scale^t$ and $scale^s$ in Equation (9) and Equation (12) determine the effects of token-level and sentence-level CBMI. To investigate the effects of the two CBMI in different granularities, we firstly fix $scale^t$ to a moderate value, *i.e.*, 0.1, and tune $scale^s$ from 0.0 to 0.3 with the step of 0.05. The detailed results are shown in Figure 3. We observe that models perform better with larger $scale^s$, which conforms with our conjecture in Section 3.2 that well-matched sentence pairs contribute more to NMT models. Then

[6] SacreBLEU hash: BLEU+case.mixed+numrefs.1 +smooth.exp+tok.13a+version.1.5.1.

Model	WMT14 En→De	WMT19 Zh→En
Transformer$_{base}$ (Vaswani et al., 2017) †	27.30	–
Transformer$_{base}$ (Vaswani et al., 2017)	28.10	25.36
+ Freq-Exponential (Gu et al., 2020)	28.43 (+0.33)	24.99 (-0.37)
+ Freq-Chi-Square (Gu et al., 2020)	28.47 (+0.37)	25.43 (+0.07)
+ BMI-adaptive (Xu et al., 2021)	28.56 (+0.45)	25.77 (+0.41)
+ Focal Loss (Lin et al., 2017)	28.43 (+0.33)	25.37 (+0.01)
+ Anti-Focal Loss (Raunak et al., 2020)	28.65 (+0.55)	25.50 (+0.14)
+ Self-Paced Learning (Wan et al., 2020)	28.69 (+0.59)	25.75 (+0.39)
+ Simple Fusion (Stahlberg et al., 2018)	27.82 (-0.28)	23.91 (-1.45)
+ LM Prior (Baziotis et al., 2020)	28.27 (+0.17)	25.71 (+0.35)
+ CBMI-adaptive (ours)	**29.01 (+0.91)****	**26.21 (+0.85)****
Transformer$_{big}$ (Vaswani et al., 2017) †	28.40	–
Transformer$_{big}$ (Vaswani et al., 2017)	29.31	25.48
+ Freq-Exponential (Gu et al., 2020)	29.66 (+0.35)	25.57 (+0.09)
+ Freq-Chi-Square (Gu et al., 2020)	29.64 (+0.33)	25.64 (+0.14)
+ BMI-adaptive (Xu et al., 2021)	29.69 (+0.38)	25.81 (+0.33)
+ Focal Loss (Lin et al., 2017)	29.65 (+0.34)	25.54 (+0.06)
+ Anti-Focal Loss (Raunak et al., 2020)	29.72 (+0.41)	25.64 (+0.16)
+ Self-Paced Learning (Wan et al., 2020)	29.85 (+0.54)	25.88 (+0.40)
+ CBMI-adaptive (ours)	**30.12 (+0.81)***	**26.30 (+0.82)***

Table 1: BLEU scores (%) on two translation tasks. Each experiment runs over 3 times and we list the mean values and improvements in this table (full results including standard deviations are shown in Appendix A). '†' represents the results taken from the corresponding papers. Results with mark $*/**$ are statistically (Koehn, 2004) better than the most related method 'BMI-Adaptive' with $p < 0.05$ and $p < 0.01$.

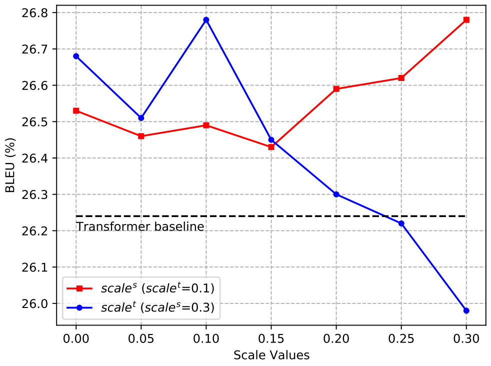


Figure 3: BLEU scores (%) on the validation set of WMT14 En-De with different $scale^t$ and $scale^s$ that defined in the Equation (9) and (12).

we fix $scale^s$ to 0.3 and tune $scale^t$ in a similar way. We find it better to keep $scale^t$ in a small range and too large value is harmful for models. We conjecture that over-focus on the high-CBMI tokens brings another imbalance for training and may hurt the models. Thus we set $scale^t$ to 0.1 in our following experiments.

4.4 Baseline Systems

We implement our approach based on the Transformer (Vaswani et al., 2017) and compare it with some mainstream adaptive training methods (detailed hyperparameter settings are provided in Appendix C).

Transformer. We follow the standard base/big model configurations (Vaswani et al., 2017) to implement our baseline systems.

Freq-Exponential. Gu et al. (2020) use monolingual token frequency to design an exponential weight function for token-level adaptive training:

$$w_j = A \cdot e^{-T \cdot Count(y_j)} + 1$$

where A and T are two hyperparameters to adjusting the distribution of weights.

Freq-Chi-Square. Gu et al. (2020) use the chi-square distribution to filter out extremely low frequency target tokens:

$$w_j = A \cdot Count(y_j)^2 e^{-T \cdot Count(y_j)} + 1$$

where A and T play the same roles as above.

BMI-adaptive. Xu et al. (2021) calculate BMI (in Equation (5)) during the data pre-processing stage and scale it for adaptive loss weights.

$$w_j = S \cdot \mathrm{BMI}(\mathbf{x}, y_j) + B \tag{14}$$

where S and B are hyperparameters to scale BMI to an appropriate range.

Focal Loss. Lin et al. (2017) propose the focal loss for objective detection tasks to solve the class imbalance problem. Here we introduce it into NMT.

$$\mathcal{L}_{fl} = -(1 - \alpha p)^{\gamma} \log p \tag{15}$$

where α and γ are hyperparameters to adjust the loss weight and p is the NMT predicted probability.

Anti-Focal Loss. Raunak et al. (2020) design an anti-focal loss function to solve the long-tailed problem in NMT by incorporating the inductive bias of inference into training.

$$\mathcal{L}_{afl} = -(1 + \alpha p)^{\gamma} \log p \tag{16}$$

where α and γ are similarly as the above focal loss.

Self-Paced Learning. Wan et al. (2020) calculate model confidence via Monte Carlo dropout sampling (Gal and Ghahramani, 2016) to measure the token difficulty and use it to re-weight the training losses of tokens.

Simple Fusion. Stahlberg et al. (2018) propose two simple strategies (i.e., PRENORM and POSTNORM) to fuse the NMT probabilities with the LM probablities and directly optimize the fusion during the NMT training process[7].

LM Prior. Baziotis et al. (2020) propose to distill the prior knowledge from LMs trained on rich-resource monolingual data to low-resource NMT models[8]:

$$\mathcal{L}_{lmp} = \mathcal{L}_{\mathrm{NMT}} + \lambda \cdot \mathcal{L}_{\mathrm{KL}}(p_{\mathrm{LM}} || p_{\mathrm{NMT}}; \tau) \tag{17}$$

where λ weights the distillation term and τ is the softmax temperature (Hinton et al., 2015).

4.5 Results

The overall results on two WMT tasks based on the Transformer$_{base}$ and Transformer$_{big}$ configurations are shown in Table 1. Under the Transformer$_{base}$ setting, CBMI-based adaptive training can respectively improve +0.91 and +0.85 BLEU scores on En-De and Zh-En tasks compared to the Transformer baseline. Compared to the most related yet target-context-free strategy 'BMI-adaptive', our CBMI-based adaptive training strategy can respectively yield significant improvements up to +0.46 and +0.44 BLEU scores on En-De and Zh-En, which demonstrate the significance of the target context for token assessment in token-level adaptive training. Compared with the best performing baseline 'Self-Paced Learning', our approach still outperforms it by +0.32 and +0.46 BLEU scores on the two tasks. Our conjecture is that CBMI not only reflects the model competence used in 'Self-Paced Learning' but also further incorporates the linguistic statistical information from the target-side LM, thus reflects more explicit translation property (*i.e.,* adequacy). However, other LM enhanced methods (*e.g.,* 'Simple Fusion' and 'LM Prior') bring limited improvement or even degradation to the NMT models when there is no extra data for the LMs, which further proves the utilization of the LM in our approach is more effective. Under the Transformer$_{big}$ setting, where the performances of existing methods are limited, our method can still bring the improvement of +0.81 and +0.82 BLEU scores on the En-De and Zh-En, which demonstrates the superiority of CBMI under stronger baselines.

5 Analysis

In this section, we provide in-depth analyses on the effectiveness of our CBMI and conduct experiments on the validation set of WMT14 En-De with the Transformer$_{base}$ model.

5.1 Effects of Different Levels of CBMI

We take the Transformer$_{base}$ as baseline, and then apply adaptive training based on the token-level CBMI, the sentence-level CBMI, and both of them, respectively. Results are listed in Table 2. We observe certain improvements (+0.29 and +0.44 BLEU scores) when separately applying the token- and sentence-level CBMI based approaches. It suggests that our CBMI can measure the token importance from different granularities, and up-weight

[7] The results in Table 1 are the higher ones between the two strategies.

[8] We did not use extra monolingual data for the LMs in 'Simple Fusion' and 'LM Prior' in our implementation for fair comparison.

Model	BLEU
Transformer$_{base}$	26.24
+ token-level CBMI	26.53 (+0.29)
+ sentence-level CBMI	26.68 (+0.44)
+ token- & sentence-level CBMI	**26.78 (+0.54)**

Table 2: BLEU scores (%) of CBMI at different granularities on the validation set of WMT14 En-De.

Method	Pre-process	#Params	Train	Disk
	(hour)	(M)	(hour)	(GB)
Transformer$_{base}$	0	65	10	0
+ BMI	12	65	11	2.0
+ target context	≈ 12×N	65	–	≈ 2.0×N
+ CBMI	0	101	12	0

Table 3: The costs of calculation and storage of the BMI- and CBMI-based approaches on the WMT14 En-De (100k training steps). 'N' refers to the average length of target sentences.

Model	Adequacy	Fluency	Avg.
Transformer$_{base}$	4.25	4.69	4.47
+ CBMI-adaptive	**4.53***	**4.75**	**4.64**

Table 4: Human evaluation on adequacy and fluency. $*$ means the average Cohen's Kappa (Cohen, 1960) is higher than 0.6, which indicates substantial agreement between three annotators (Landis and Koch, 1977).

the important tokens or sentence pairs can improve translation quality. Furthermore, the combination of both the token- and sentence-level CBMI brings further improvement (+0.55 BLEU scores), which illustrates that the CBMI in different granularities are complementary and have cumulative gains.

5.2 Costs of Computing and Storage

In this section, we compare our CBMI-based approach with the BMI-based adaptive training in terms of the number of trainable parameters, the CPU computational costs of pre-processing, the GPU computational costs of training, and disk cost for storing intermediate variables. As shown in Table 3, the vanilla BMI-based approach requires additional 12 CPU hours to obtain the BMI values during the pre-processing stage, and about 2.0 GB of disk space to store these BMI values. To make matters worse, the costs of CPU calculation and disk storage will increase dozens of times (approximately equal to the average length of target sentences) when conducting the target-context-aware calculations for BMI. In contrast, our CBMI-based approach gets rid of the CPU computational costs, and thus has no additional storage overhead. Although we introduce an additional LM to calculate the CBMI values, it only brings a slight increase of model parameters and GPU calculation cost during model training. Particularly, our proposed method simply modifies the training loss of NMT, and thus has no effect on the inference speed. In short, our CBMI can be efficiently calculated during model training without any pre-specific statistical calculations and storage overhead, which makes it feasible to supplement target context information for statistical metrics.

5.3 Human Evaluation

To verify whether our CBMI measurement is indeed highly related to the translation adequacy of NMT models, as we conjectured in Section 3.2, we conduct the human evaluation in terms of adequacy and fluency. We randomly sample 100 sentences from the test set of WMT19 Zh-En and invite three annotators to evaluate the translation adequacy and fluency. Scores for both indexes are limited in [1,5]. For adequacy, '1' represents irrelevant to the source sentence and '5' represents semantically equal. For fluency, '1' means unintelligible and '5' means fluent and native. We finally average the scores from three annotators and list the results in Table 4. We observe that our approach significantly promotes the translation adequacy of the Transformer$_{base}$ baseline, and meanwhile slightly promotes the translation fluency. It indicates that the CBMI measurement is highly related to the adequacy of NMT models, and focusing more on the tokens with high CBMI can improve translation adequacy, and thus improve translation performance.

5.4 Prior Selection based on CBMI

Given that CBMI reflects the dependency between a target token and its source sentence on the condition of its target context, in this section, we explore whether CBMI can serve as an indicator for selecting an appropriate prior distribution to improve the NMT model. Prior distributions have been proved for their ability to provide additional knowledge for models (Baziotis et al., 2020; Li et al., 2020). Thus we try three generated distributions as prior distributions for NMT models, *i.e.*, the translation model distribution (TM prior), the language model distribution (LM prior), and the softmax normalized CBMI distribution (CBMI prior).

To verify the correctness of these prior distributions, we firstly calculate the top-1 accuracies of

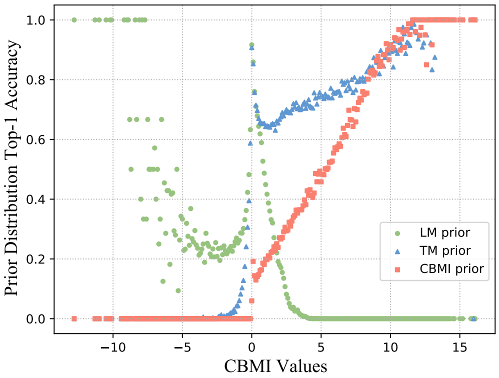


Figure 4: The most accurate prior distribution for tokens with different CBMI values. The LM prior (green circles) performs most accurate for tokens with lower CBMI values, the TM prior (blue triangles) performs best for tokens with moderate CBMI values, and the CBMI prior (red squares) performs best for tokens with higher CBMI values.

Prior Distribution	**BLEU**
Transformer$_{base}$	26.24
+ LM Prior	26.73 (+0.49)
+ TM Prior	26.61 (+0.37)
+ CBMI Prior	26.57 (+0.33)
+ Prior Selection	**26.75 (+0.51)**

Table 5: BLEU scores (%) on WMT14 En-De validation set for different prior distributions on all tokens.

these distributions according to different tokens and surprisingly observe that the accuracies are highly related to the CBMI values of tokens. As shown in Figure 4, the most accurate prior for target tokens with different CBMI values is not always consistent. Based on this observation, we further design a CBMI-based prior selection strategy to choose the best prior distribution for each token. The details of the selection strategy are seen in Appendix D.

As shown in Table 5, all these prior distributions can provide helpful guidance and enhance the baseline model. More importantly, the CBMI-based prior selection strategy can achieve a better performance compared with the single prior, demonstrating that CBMI also serves as an appropriate indicator for the translation prior selection. We will explore the more sophisticated CBMI-based prior selection strategy in the future work.

6 Related Work

Language Model Enhanced NMT. Exploiting the information in language models is a common solution to improve NMT models. In low-resource scenarios, LMs trained on extra monolingual data are usually more informative and thus used to fuse with NMT prediction (Gulcehre et al., 2015, 2017; Sriram et al., 2017; Stahlberg et al., 2018), provide prior knowledge for NMT models (Baziotis et al., 2020) and enhance representations of NMT (Clinchant et al., 2019; Zhu et al., 2020). In data augmentation methods, LMs are also widely used to generate contextual substitutions of words in sentences (Kobayashi, 2018; Wu et al., 2018; Gao et al., 2019). Differently, all the aforementioned methods rely on the LMs that are trained on extra data, while the LM in our method does not require extra data and also has no influence on the inference speed.

7 Conclusion

In this paper, we propose a target-context-aware metric for target tokens, named conditional bilingual mutual information (CBMI). Compared with previous statistical metrics, our CBMI only increases limited computational costs to incorporate the target context and provides a more suitable assessment for tokens. Furthermore, based on the token- and sentence-level CBMI, we design a CBMI-based adaptive training strategy to amply the contributions of the important tokens. Experimental results on two WMT tasks demonstrate the effectiveness of our proposed approach. Further analyses show that CBMI can improve translation adequacy and serve as an appropriate indicator for the translation prior selection.

Acknowledgements

The research work descried in this paper has been supported by the National Key R&D Program of China (2020AAA0108001) and the National Nature Science Foundation of China (No. 61976016, 61976015, and 61876198). The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this paper.

References

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

Christos Baziotis, Barry Haddow, and Alexandra Birch. 2020. Language model prior for low-resource neural machine translation. In *Proceedings of the 2020*

Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 7622–7634, Online. Association for Computational Linguistics.

Stephane Clinchant, Kweon Woo Jung, and Vassilina Nikoulina. 2019. On the use of BERT for neural machine translation. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 108–117, Hong Kong. Association for Computational Linguistics.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA. PMLR.

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, Florence, Italy. Association for Computational Linguistics.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N. Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1243–1252. JMLR.org.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press. `http://www.deeplearningbook.org`.

Shuhao Gu, Jinchao Zhang, Fandong Meng, Yang Feng, Wanying Xie, Jie Zhou, and Dong Yu. 2020. Token-level adaptive training for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1035–1046, Online. Association for Computational Linguistics.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, and Yoshua Bengio. 2017. On integrating a language model into neural machine translation. *Computer Speech & Language*, 45:137–148.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Soren Johansen and Katarina Juselius. 1990. Maximum likelihood estimation and inference on cointegration—with appucations to the demand for money. *Oxford Bulletin of Economics and statistics*, 52(2):169–210.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *International Conference on Learning Representations*.

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, New Orleans, Louisiana. Association for Computational Linguistics.

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, Barcelona, Spain. Association for Computational Linguistics.

J Richard Landis and Gary G Koch. 1977. The measurement of observer agreement for categorical data. *biometrics*, pages 159–174.

Zuchao Li, Rui Wang, Kehai Chen, Masso Utiyama, Eiichiro Sumita, Zhuosheng Zhang, and Hai Zhao. 2020. Data-dependent gaussian prior objective for language generation. In *International Conference on Learning Representations*.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2999–3007.

Yijin Liu, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. 2021a. Confidence-aware scheduled sampling for neural machine translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2327–2337, Online. Association for Computational Linguistics.

Yijin Liu, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. 2021b. Scheduled sampling based on decoding steps for neural machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3285–3296, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Fandong Meng and Jinchao Zhang. 2019. DTMT: A novel deep transition architecture for neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 224–231.

Mengqi Miao, Fandong Meng, Yijin Liu, Xiao-Hua Zhou, and Jie Zhou. 2021. Prevent the language model from being overconfident in neural machine translation. 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 3456–3468, Online. Association for Computational Linguistics.

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

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. 2018. Scaling neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 1–9, Brussels, Belgium. Association for Computational Linguistics.

Vikas Raunak, Siddharth Dalmia, Vivek Gupta, and Florian Metze. 2020. On long-tailed phenomena in neural machine translation. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3088–3095, Online. Association for Computational Linguistics.

Raj Reddy. 1977. Speech understanding systems: summary of results of the five-year research effort at carnegie-mellon university. *Pittsburgh, Pa*.

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, Berlin, Germany. Association for Computational Linguistics.

C. E. Shannon. 1948. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423.

Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates. 2017. Cold fusion: Training seq2seq models together with language models.

Felix Stahlberg, James Cross, and Veselin Stoyanov. 2018. Simple fusion: Return of the language model. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 204–211, Brussels, Belgium. Association for Computational Linguistics.

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 *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Yu Wan, Baosong Yang, Derek F. Wong, Yikai Zhou, Lidia S. Chao, Haibo Zhang, and Boxing Chen. 2020. Self-paced learning for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1074–1080, Online. Association for Computational Linguistics.

Rongxiang Weng, Heng Yu, Xiangpeng Wei, and Weihua Luo. 2020. Towards enhancing faithfulness for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2675–2684, Online. Association for Computational Linguistics.

Xing Wu, Shangwen Lv, Liangjun Zang, Jizhong Han, and Songlin Hu. 2018. Conditional bert contextual augmentation.

Yangyifan Xu, Yijin Liu, Fandong Meng, Jiajun Zhang, Jinan Xu, and Jie Zhou. 2021. Bilingual mutual information based adaptive training for neural machine translation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 511–516, Online. Association for Computational Linguistics.

Jinhua Zhu, Yingce Xia, Lijun Wu, Di He, Tao Qin, Wengang Zhou, Houqiang Li, and Tieyan Liu. 2020. Incorporating bert into neural machine translation. In *International Conference on Learning Representations*.

George Kingsley Zipf. 1949. Human behavior and the principle of least effort.

A Complete Results

To prove the generality of the experimental results, we provide the complete results on two translation tasks which contain mean values and standard deviations in Table 6.

B Training Hyperparameters and Model Configurations

To assure the reproducibility of our experimental results, we provide the training details of our method and the model configurations in Table 7. The NMT models and LMs in our method use the same corpus and BPE vocabulary, so that they can generate two corresponding probability distributions for the same token and calculate its CBMI during training. Our LMs have the same model architecture and configuration with the NMT models' decoder except for the cross-attention module, yet we do not share their embedding layers for higher performance.

Model	WMT14 En→De	WMT19 Zh→En
Transformer$_{base}$ (Vaswani et al., 2017)	28.10 ± 0.20	25.36 ± 0.19
+ Freq-Exponential (Gu et al., 2020)	28.43 ± 0.03	24.99 ± 0.01
+ Freq-Chi-Square (Gu et al., 2020)	28.47 ± 0.24	25.43 ± 0.72
+ BMI-adaptive (Xu et al., 2021)	28.56 ± 0.09	25.77 ± 0.04
+ Focal Loss (Lin et al., 2017)	28.43 ± 0.10	25.37 ± 0.25
+ Anti-Focal Loss (Raunak et al., 2020)	28.65 ± 0.13	25.50 ± 0.33
+ Self-Paced Learning (Wan et al., 2020)	28.69 ± 0.22	25.75 ± 0.25
+ Simple Fusion (Stahlberg et al., 2018)	27.82 ± 0.17	23.91 ± 0.22
+ LM Prior (Baziotis et al., 2020)	28.27 ± 0.10	25.71 ± 0.42
+ CBMI-weight (ours)	**29.01 ± 0.08****	**26.21 ± 0.30****
Transformer$_{big}$ (Vaswani et al., 2017)	29.31 ± 0.29	25.48 ± 0.31
+ Freq-Exponential (Gu et al., 2020)	29.66 ± 0.04	25.57 ± 0.15
+ Freq-Chi-Square (Gu et al., 2020)	29.64 ± 0.45	25.64 ± 0.23
+ BMI-adaptive (Xu et al., 2021)	29.69 ± 0.15	25.81 ± 0.13
+ Focal Loss (Lin et al., 2017)	29.65 ± 0.11	25.54 ± 0.09
+ Anti-Focal Loss (Raunak et al., 2020)	29.72 ± 0.16	25.64 ± 0.18
+ Self-Paced Learning (Wan et al., 2020)	29.85 ± 0.18	25.88 ± 0.23
+ CBMI-weight (ours)	**30.12 ± 0.13***	**26.30 ± 0.26***

Table 6: The complete results of Table 1 containing mean values and standard deviations of BLEU scores.

Hyperparameters	Base	Big
Embedding Size	512	1024
Encoder Layers	6	6
Decoder Layers	6	6
Attention Heads	8	16
LM Layers	6	6
LM Attention Heads	8	16
Residual Dropout	0.1	0.3
Attention Dropout	0.1	0.1
Activation Dropout	0.1	0.1
Learning Rate	7e-4	5e-4
Learning Rate Decay	inverse sqrt	inverse sqrt
Warmup Steps	4000	4000
Layer Normalization	PostNorm	PostNorm

Table 7: Training hyperparameters and model configurations of our method.

C Implementation Details for Baseline Systems

To make our experimental comparison more convincing, we present the details of hyperparameters involved in the baseline systems described in Section 4.4.

Freq-Exponential. Following the best hyperparameter setting in (Gu et al., 2020), we set A to 1.0 and T to 1.75 for the En-De task, and A to 1.0 and T to 0.35 for the Zh-En task.

Freq-Chi-Square. Similarly, we set A to 1.0 and T to 2.50 for the En-De task, and A to 1.0 and T to 1.75 for the Zh-En task according to (Gu et al., 2020).

BMI-adaptive. According to the settings in (Xu et al., 2021), we set S to 0.15 and B to 0.8 for the En-De task and S to 0.1 and B to 1.0 for the Zh-En task.

Focal Loss. As suggested in (Lin et al., 2017), we fix γ to 1.0 and search a α among [0.1, 0.5] which performs best on the validation sets of two tasks. Finally, we set α to 0.1 for both tasks.

Anti-Focal Loss. Similar with the settings in focal loss, we also fix γ to 1.0 and tune α for two tasks. Lastly, we also set α to 0.1 for both tasks.

LM Prior. We set the softmax temperature τ to 2.0 following the settings in (Baziotis et al., 2020) while λ to 0.1 according to the performances on the validation sets.

D Details for the Prior Selection Strategy

In our prior selection strategy, we firstly divide the target tokens in each mini-batch into three intervals according to their original CBMI values. Corresponding to the observation in Figure 4, we respectively apply the LM prior, the TM prior and

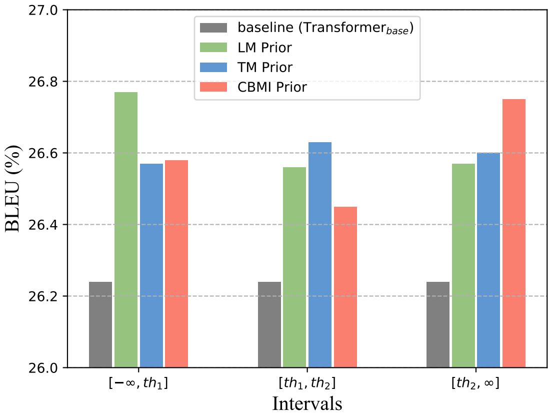


Figure 5: BLEU scores on the validation set of WMT14 En-De for different prior distributions on different CBMI intervals.

the CBMI prior on the tokens in the three intervals. Formally, the prior distribution $q(y_j)$ for target token y_j can be represented as follows:

$$q(y_j) = \begin{cases} q_{\text{LM}}, & \text{CBMI}(\mathbf{x}; y_j) \in [-\infty, th_1] \\ q_{\text{TM}}, & \text{CBMI}(\mathbf{x}; y_j) \in [th_1, th_2] \\ q_{\text{CBMI}}, & \text{CBMI}(\mathbf{x}; y_j) \in [th_2, \infty] \end{cases} \quad (18)$$

where th_1 and th_2 are two hyperparameters and empirically set to 0 and 8 according to the observations in Figure 4. q_{LM}, q_{TM}, q_{CBMI} represent the aforementioned three prior distributions.

Subsequently, we calculate the cross-entropy loss between the selected prior distribution and the model predicted distribution as an additional term and incorporate it with the original cross-entropy loss in Equation (2) to make up the new training objective:

$$\begin{aligned}\mathcal{L}(\theta) =& \mathcal{L}_{\text{CE}}(\theta) \\ & + \lambda \cdot \sum_{y=1}^{|\mathbf{y}|} -q(y_j) \log p(y_j|\mathbf{y}_{<j}, \mathbf{x}; \theta)\end{aligned} \quad (19)$$

where λ is a hyperparameter that controls the effect of prior distribution. In our experiments, we set λ to 0.1 according to the performances on the validation set.

To verify the reasonablility of the prior selection strategy, we compare the effects of the three priors on each single CBMI intervals in Figure 5. As we expected, the BLEU results also conform with the accuracy results in Figure 4, indicating that the most helpful prior distribution can be highly related to the CBMI values of tokens.