Towards Understanding and Improving Knowledge Distillation for Neural Machine Translation

Songming Zhang, Yunlong Liang, Shuaibo Wang, Yufeng Chen[*]
Wenjuan Han, **Jian Liu**, and **Jinan Xu**
Beijing Key Lab of Traffic Data Analysis and Mining,
Beijing Jiaotong University, Beijing, China
{smzhang22,yunlongliang,chenyf,wjhan,jianliu,jaxu}@bjtu.edu.cn

Abstract

Knowledge distillation (KD) is a promising technique for model compression in neural machine translation. However, where the knowledge hides in KD is still not clear, which may hinder the development of KD. In this work, we first unravel this mystery from an empirical perspective and show that the knowledge comes from the top-1 predictions of teachers, which also helps us build a potential connection between word- and sequence-level KD. Further, we point out two inherent issues in vanilla word-level KD based on this finding. Firstly, the current objective of KD spreads its focus to whole distributions to learn the knowledge, yet lacks special treatment on the most crucial top-1 information. Secondly, the knowledge is largely covered by the golden information due to the fact that most top-1 predictions of teachers overlap with ground-truth tokens, which further restricts the potential of KD. To address these issues, we propose a novel method named **T**op-1 **I**nformation **E**nhanced **K**nowledge **D**istillation (TIE-KD). Specifically, we design a hierarchical ranking loss to enforce the learning of the top-1 information from the teacher. Additionally, we develop an iterative KD procedure to infuse more additional knowledge by distilling on the data without ground-truth targets. Experiments on WMT'14 English-German, WMT'14 English-French and WMT'16 English-Romanian demonstrate that our method can respectively boost Transformer$_{base}$ students by +1.04, +0.60 and +1.11 BLEU scores and significantly outperform the vanilla word-level KD baseline. Besides, our method shows higher generalizability on different teacher-student capacity gaps than existing KD techniques.

1 Introduction

In recent years, neural machine translation (NMT) has made marvelous progress in generating high-quality translations (Bahdanau et al., 2014; Gehring et al., 2017; Vaswani et al., 2017; Liang et al., 2021b, 2022), especially with some exquisite and deep model architectures (Wei et al., 2020; Li et al., 2020; Liu et al., 2020; Wang et al., 2022). Despite their amazing performance on translation tasks, high computational and deployment costs still prevent these models from being applied in real life. On this problem, knowledge distillation (KD) (Liang et al., 2008; Hinton et al., 2015; Kim and Rush, 2016; Wu et al., 2020; Chen et al., 2020; Wang et al., 2021; Liang et al., 2021a) is regarded as a promising solution for model compression, which aims to transfer the knowledge from these strong teacher models into compact student models.

Generally, there are two categories of KD techniques, *i.e.*, word-level KD (Hinton et al., 2015; Kim and Rush, 2016; Wang et al., 2021) and sequence-level KD (Kim and Rush, 2016). (1) Word-level KD is conducted on each target token, where it shrinks the Kullback-Leibler (KL) divergence (Kullback and Leibler, 1951) between the predicted distributions from the student and the soft targets from the teacher. In these soft targets, the knowledge was previously deemed to come from the probability relationship between negative candidates (*i.e.*, the correlation information) (Hinton et al., 2015; Tang et al., 2020; Jafari et al., 2021). (2) Sequence-level KD instead requires no soft target and directly encourages students to maximize the sequence probability of the final translation decoded by the teacher. Although both techniques work quite differently, they still achieve similarly superior effectiveness. Therefore, we raise two heuristic questions on KD in NMT:

- *Q1: Where does the knowledge actually come from during KD in NMT?*
- *Q2: Is there any connection between the word- and the sequence-level KD techniques?*

To answer these two questions, we conduct an

[*] Yufeng Chen is the corresponding author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 8062–8079
July 9-14, 2023 ©2023 Association for Computational Linguistics

empirical study that starts from word-level KD to find out where the knowledge hides in the teacher's soft targets and then explore whether the result can be expanded to sequence-level KD. As a result, we summarize several intriguing findings:

i. Compared to the correlation information, the information of the teacher's top-1 predictions (*i.e.*, the top-1 information) actually determines the benefit of word-level KD (§3.1).

ii. The correlation information can be successfully learned by students during KD but fails to improve their final performance (§3.2).

iii. Extending the top-1 information to top-k does not lead to further improvement (§3.3).

iv. The top-1 information is important even when the teacher is under-confident in its top-1 predictions (§3.4).

v. Similar importance of the top-1 information can also be verified on sequence-level KD (§3.5).

These findings sufficiently prove that **1) the knowledge actually comes from the top-1 information of the teacher during KD in NMT**, and **2) the two kinds of KD techniques can be connected from the perspective of the top-1 information**.

On these grounds, we further point out that there are two inherent issues in vanilla word-level KD. Firstly, as the source of teachers' knowledge, the top-1 information receives no special treatment in the training objective of vanilla word-level KD since the KL divergence directly optimizes the entire distribution. Secondly, since most top-1 predictions of strong teachers overlap with ground-truth tokens (see the first row of Tab.1), the additional knowledge from teachers beyond the golden information is poor and the potential of word-level KD is largely limited (see the second row of Tab.1). To address these issues, we propose a new KD method named **T**op-1 **I**nformation **E**nhanced **K**nowledge **D**istillation (TIE-KD) for NMT. Specifically, we first design a hierarchical ranking loss that can enforce the student model to learn the top-1 information through ranking the top-1 predictions of the teacher as its own top-1 predictions. Moreover, we develop an iterative KD procedure to expose more input data without ground-truth targets for KD to exploit more knowledge from the teacher.

Datasets	En-De	En-Fr	En-Ro
Top-1 Overlap Rate	68%	78%	94%
Δ from Word-level KD	+0.61	+0.13	+0.18

Table 1: The overlap rates between the top-1 predictions of teachers and ground-truth tokens on WMT'14 English-German (En-De), WMT'14 English-French (En-Fr) and WMT'16 English-Romanian (En-Ro) and the corresponding improvement (Δ) of BLEU scores brought by word-level KD on the test set of these tasks[1].

We evaluate our TIE-KD method on three WMT benchmarks, *i.e.*, WMT'14 English-German (En-De), WMT'14 English-French (En-Fr) and WMT'16 English-Romanian (En-Ro). Experimental results show that our method can boost Transformer$_{base}$ students by +1.04, +0.60, +1.11 BLEU scores and significantly outperforms the vanilla word-level KD approach. Besides, we test the performance of existing KD techniques in NMT and our TIE-KD under different teacher-student capacity gaps and show the stronger generalizability of our method on various gaps.

Our contributions are summarized as follows[2]:

- To the best of our knowledge, we are the first to explore where the knowledge hides in KD for NMT and unveil that it comes from the top-1 information of the teacher, which also helps us build a connection between word- and sequence-level KD.
- Further, we point two issues in vanilla word-level KD and propose a novel KD method named Top-1 Information Enhanced Knowledge Distillation (TIE-KD) to address them. Experiments on three WMT benchmarks demonstrate its effectiveness and superiority.
- We investigate the effects of current KD techniques in NMT under different teacher-student capacity gaps and show the stronger generalizability of our approach to various gaps.

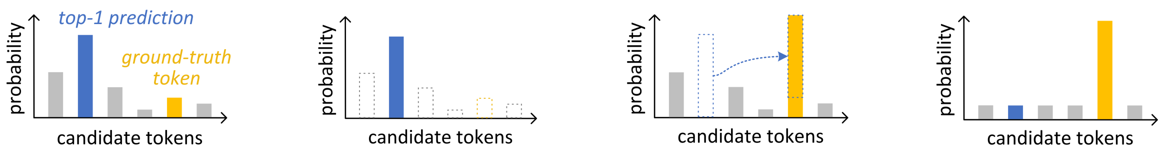
2 Background

2.1 Neural Machine Translation

Given a source sentence with M tokens $\mathbf{x} = \{x_1, x_2, \ldots, x_M\}$ and the corresponding target sentence with N tokens $\mathbf{y} = \{y_1, y_2, \ldots, y_N\}$, NMT

[1] We random sample 3000 target sentences in the training set of each task to calculate the approximate overlap rates.

[2] The code is publicly available at: https://github.com/songmzhang/NMT-KD.



(a) vanilla word-level KD (b) *w/o* correlation info (c) *w/o* top-1 info (d) *w/o* KD

Figure 1: Removing different information from the original soft targets provided by the teacher during word-level KD. Note that the soft target in "*w/o* KD" is equivalent to the soft target of label smoothing.

models are trained to maximize the probability of each target token conditioning on the source sentence by the cross-entropy (CE) loss:

$$\mathcal{L}_{ce} = -\sum_{j=1}^{N} \log p(y_j^* | \mathbf{y}_{<j}, \mathbf{x}; \theta), \tag{1}$$

where y_j^* and $\mathbf{y}_{<j}$ denote the ground-truth target and the target-side previous context at time step j, respectively. And θ is the model parameter.

2.2 Word-level Knowledge Distillation

Word-level KD (Kim and Rush, 2016) aims to minimizes the KL divergence between the output distributions of the teacher model and the student model on each target token. Formally, given the probability distribution $q(\cdot)$ from the teacher model, the KL divergence-based loss is formulated as follows:

$$\mathcal{L}_{kd} = \mathcal{L}_{\text{KL}} = \tag{2}$$

$$\sum_{j=1}^{N} D_{\text{KL}}\Big(q(y_j | \mathbf{y}_{<j}, \mathbf{x}; \theta_t) \big|\big| p(y_j | \mathbf{y}_{<j}, \mathbf{x}; \theta_s)\Big),$$

where θ_t and θ_s denote the model parameters of the teacher and the student, respectively.

Then, the overall loss function of word-level KD is the linear interpolation between the CE loss and the KL divergence loss:

$$\mathcal{L}_{word\text{-}kd} = (1-\alpha)\mathcal{L}_{ce} + \alpha\mathcal{L}_{kd}. \tag{3}$$

2.3 Sequence-level Knowledge Distillation

Sequence-level KD (Kim and Rush, 2016) encourages the student model to imitate the sequence probabilities of the translations from the teacher model. To this end, it optimizes the student model through the following approximation:

$$\begin{aligned}\mathcal{L}_{seq\text{-}kd} &= -\sum_{\mathbf{y}\in\mathcal{Y}} Q(\mathbf{y}|\mathbf{x};\theta_t) \log P(\mathbf{y}|\mathbf{x};\theta_s) \\ &\approx -\log P(\widehat{\mathbf{y}}|\mathbf{x};\theta_s),\end{aligned} \tag{4}$$

where $\mathcal{Y}$ denotes the hypothesis space of the teacher and $\widehat{\mathbf{y}}$ is the approximate result through the teacher's beam search.

3 Probing the Knowledge of KD in NMT

In this section, we start from word-level KD and offer exhaustive empirical analyses on 1) the determining information in word-level KD (§3.1); 2) whether the correlation information has been learned (§3.2); 3) whether there are more benefits when extending the top-1 to top-k information (§3.3) and 4) the importance of the top-1 information on soft targets with different confidence (§3.4). Then we expand the conclusion to sequence-level KD (§3.5) and lastly revisit KD for NMT from a novel view (§3.6).

3.1 Which Information Determines the Performance of Word-level KD?

In word-level KD, the relative probabilities between negative candidates in the soft targets from the teacher contain rich correlation information, which is previously deemed to carry knowledge from the teacher (Hinton et al., 2015; Tang et al., 2020; Jafari et al., 2021). However, in practice, strong teachers usually have high confidence in their top-1 predictions while retain little probability mass for other candidates. Hence, to study the mystery of KD, it is necessary to first investigate the real effects of the correlation information and the top-1 prediction information during KD and then figure out which one actually determines the performance of KD.

To this end, during word-level KD, we separately remove the top-1 information and the correlation information from the original soft targets of the teacher (as depicted in Fig.1) and then observe the corresponding performance. Besides the BLEU score, we also introduce a new metric, namely the **Top-1 Agreement (TA)** rate, which calculates the overlap rate of the top-1 predictions between the student and the teacher on each position under the teacher-forcing mode. As shown in Tab.2, the performance slightly increases when we remove the probabilities of all other candidates except for the

Task	Model	TA	BLEU
En-De	(a) vanilla word-level KD	88.98	26.66
	(b) *w/o* correlation info	88.69	26.76
	(c) *w/o* top-1 info	87.49	26.43
	(d) *w/o* KD	87.22	26.37
En-Fr	(a) vanilla word-level KD	89.31	34.94
	(b) *w/o* correlation info	89.19	35.09
	(c) *w/o* top-1 info	88.34	34.33
	(d) *w/o* KD	88.33	34.69
En-Ro	(a) vanilla word-level KD	83.98	34.29
	(b) *w/o* correlation info	84.27	34.30
	(c) *w/o* top-1 info	83.73	34.02
	(d) *w/o* KD	83.34	34.04

Table 2: Top-1 Agreement rates (%) and BLEU scores (%) of different soft targets during KD on the validation sets of the three tasks. Deeper colors represent better performance on the corresponding metrics.

top-1 ones in soft targets (see Fig.1(b))[3]. However, when we only remove the top-1 information and keep the remaining correlation information (see Fig.1(c))[4], the performance of KD drops close to the baseline without any KD. Moreover, we observe that the TA rates are well correlated with the final BLEU scores among these students. Therefore, we conjecture that the top-1 information is the one that actually determines the performance of word-level KD (answer to *Q1*).

3.2 Can Student Models Really Learn the Correlation Information?

To further confirm the above conjecture, we examine whether the student models have successfully learned the correlation information of the teacher during KD. To achieve this, we design two metrics to measure the ranking similarities between token rankings from the student and the teacher, named top-k edit distance and top-k ranking distance.

Top-k Edit Distance. Given the top-k predictions of the teacher at time step j as $[y_j^{t_1}, ..., y_j^{t_k}]$ and the ones of the student as $[y_j^{s_1}, ..., y_j^{s_k}]$, the top-$k$ edit distance can be expressed as:

$$\mathcal{D}_{edit} = \frac{1}{N}\sum_{j}^{N} f([y_j^{t_1}, ..., y_j^{t_k}], [y_j^{s_1}, ..., y_j^{s_k}]),$$

Task	Model	$\mathcal{D}_{edit} \downarrow$	$\mathcal{D}_{rank} \downarrow$	BLEU
En-De	(a) vanilla Word-KD	2.506	1.571	26.66
	(b) *w/o* correlation info	2.697	1.791	26.76
	(c) *w/o* top-1 info	2.601	1.656	26.43
	(d) *w/o* KD	2.739	1.820	26.37
En-Fr	(a) vanilla Word-KD	2.515	1.588	34.94
	(b) *w/o* correlation info	2.616	1.696	35.09
	(c) *w/o* top-1 info	2.495	1.563	34.33
	(d) *w/o* KD	2.587	1.657	34.69
En-Ro	(a) vanilla Word-KD	2.915	2.000	34.29
	(b) *w/o* correlation info	3.025	2.138	34.30
	(c) *w/o* top-1 info	2.893	1.998	34.02
	(d) *w/o* KD	2.967	2.083	34.04

Table 3: Ranking similarities between the students and the teachers and the corresponding BLEU scores (%)[5].

where $f(\cdot, \cdot)$ calculates the edit distance.

Top-k Ranking Distance. For each $y_j^{t_i}$ in $[y_j^{t_1}, ..., y_j^{t_k}]$, this metric measures the average ranking distance between its original rank i from the teacher, and the corresponding rank from the student, denoted as $r_s(y_j^{t_i})$:

$$\mathcal{D}_{rank} = \frac{1}{Nk}\sum_{j}^{N}\sum_{i}^{k} \min(k, |i - r_s(y_j^{t_i})|).$$

We compare the students above based on these two metrics and list the results in Tab.3. Clearly, the students perform better on both $\mathcal{D}_{edit}$ and $\mathcal{D}_{rank}$ when the soft targets contain correlation information ((a),(c) *vs.* (b),(d)), indicating that student models can successfully learn the correlation information from the teacher. However, this ranking performance fails to bring better performance of KD, as measured by BLEU scores. Thus, these results negate the previous perception that the correlation information carries the knowledge during KD, which also supports our conjecture in Sec.3.1.

3.3 Does Knowledge Increase with Top-k Information?

As the importance of the top-1 information for transferring knowledge in word-level KD has been validated, we further investigate whether more knowledge can be exploited by extending top-1 information to top-k information[6]. Similar to Fig.1(b), we keep the top-k probabilities in the original soft target and remove others to extract its top-k information. However, the results in Tab.4 give a negative answer that more information does

[3] Considering the regularization effect, we do not add a uniform distribution to complement the removed probability. Please refer to Appendix B for more detailed explanations.

[4] Note that we do not simply remove the probability of the top-1 prediction, but add this probability to the ground-truth token to maintain the correctness of the distribution, *i.e.*, the soft target is unchanged if its top-1 prediction is correct.

[5] Here we set k to 5 for both $\mathcal{D}_{edit}$ and $\mathcal{D}_{rank}$ since different k does not change the conclusion in our experiments.

[6] Equivalent to vanilla word-level KD when $k = |V|$.

k		1	3	5	30	$\|V\|$
BLEU	En-De	26.76	26.74	26.76	26.70	26.66
	En-Fr	35.09	34.91	34.79	34.79	34.94
	En-Ro	34.30	34.38	34.28	34.30	34.29

Table 4: BLEU scores (%) of word-level KD with top-k information on the validation set of the three tasks. $|V|$ is the vocabulary size.

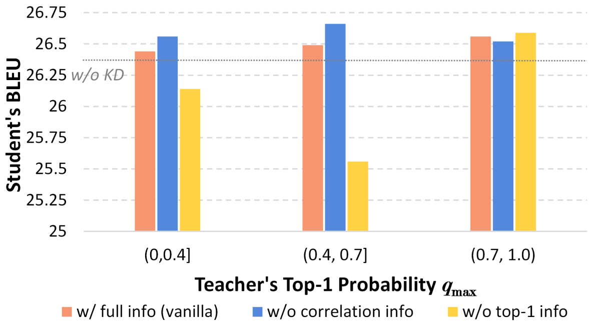


Figure 2: BLEU scores (%) of KD with different information in three intervals of soft targets on the validation set of the WMT'14 En-De task.

not bring significantly more knowledge. Thus, we can believe that almost all the knowledge of the teacher in word-level KD comes from the teacher's top-1 information, even though the whole distribution is distilled to the student.

3.4 Does Top-1 Information Work in All Soft Targets?

Although the previous results have coarsely located the knowledge in word-level KD on the top-1 information of the teacher, it is still not clear whether this holds for all types of soft targets, especially when the teacher is under-confident in its top-1 predictions. Towards this end, we divide the soft targets of the teacher into three intervals (Wang et al., 2021) based on their top-1 probabilities: $(0.0, 0.4]$, $(0.4, 0.7]$, and $(0.7, 1.0)$. Then we separately conduct the same KD processes as described in Fig.1, only using the soft targets in one of these intervals. Surprisingly, the results in Fig.2 show that even when the teacher is not so confident (*i.e.*, $q_{\max} \leq 0.7$) in its top-1 predictions, using only the top-1 information (*i.e.*, the blue bars) still achieves better performance than using the full information in corresponding soft targets. However, in these cases, removing the top-1 information in soft targets largely degrades the performance of the students. We conjecture that these under-confident top-1 predictions of the teacher can serve as hints for students to learn the difficult ground-truth labels, while the correlation information in these cases carries more noise than real knowledge for students.

ID	**top-1** (≈70%)	**non-top-1** (≈30%)	**BLEU**
1	✓	✓	26.86
2	✓	✗	26.83
3	✗	✓	2.36
4	✓(use fixed 30%)	✗	26.06
5	✓	✓+ word-level top-1 info	**26.96**

Table 5: BLEU scores (%) of sequence-level KD on the validation set of the WMT'14 En-De task when we separately use the top-1 and the non-top-1 targets of the teacher in the teacher's translations during KD.

3.5 Expanding to Sequence-level KD

Inspired by the analyses on word-level KD, we move on to sequence-level KD and decompose its loss function in Eq.(4) into a word-level form:

$$\begin{aligned}\mathcal{L}_{seq\text{-}kd} &\approx -\log P(\widehat{\mathbf{y}}|\mathbf{x};\theta_s) \\ &= -\sum_{j}^{N} \log p(\widehat{y}_j|\widehat{\mathbf{y}}_{<j},\mathbf{x};\theta_s),\end{aligned} \tag{5}$$

where $\widehat{y}_j$ is the teacher-decoded target for students at time step j. Considering the similar word-level form, it is intuitive to speculate that the top-1 information may also matter in sequence-level KD. To verify this, we divide the targets $\widehat{y}_j$ into the top-1 and the non-top-1 predictions of the teacher[7] and investigate the respective effects of these targets by separately using them during sequence-level KD. As shown in Tab.5, there is only a negligible performance change when we only use the top-1 targets for KD (row 1 *vs.* row 2). However, if we only use the non-top-1 targets, the BLEU score drastically drops (row 1 *vs.* row 3). Moreover, considering the different proportions of the two kinds of targets in the teacher's translations (*i.e.*,70% *vs.* 30%), we also use a fixed part (the same amount as the non-top-1 targets) of the top-1 targets for a fair comparison, and the performance is still steady (row 2 *vs.* row 4) and much better than using only the non-top-1 targets (row 3 *vs.* row 4). Interestingly, by adding additional word-level top-1 information to the non-top-1 part, the performance of sequence-level KD further improves (row 1 *vs.* row 5). Therefore, we can also confirm the importance of the top-1 information in sequence-level KD.

[7] There are about 70% top-1 predictions and 30% non-top-1 predictions selected by the teacher's beam search during decoding.

3.6 Rethinking KD in NMT from the Perspective of the Top-1 Information

Through the above analyses, we verify the importance of the teacher's top-1 information on both KD techniques, which actually reflects a potential connection between them. A brief theoretical analysis on this connection is provided in Appendix A. In short, the two kinds of techniques share a unified objective that imparts the teachers' top-1 predictions to student models at each time step. Thus, we believe that they are well connected on their similar working mechanisms (answer to *Q2*).

Further, we revisit word-level KD from this perspective and find two inherent issues. Firstly, the KL divergence-based objective in vanilla word-level KD directly optimizes whole distributions of students, while lacking specialized learning of the most important top-1 information. Secondly, since the top-1 predictions of the teacher mostly overlap with the ground-truth targets, the knowledge from the teacher is largely covered by the ground-truth information, which largely limits the potential of word-level KD. Therefore, we claim that the performance of the current word-level KD approach is far from perfect and the solutions to these problems are urgently needed.

4 Top-1 Information Enhanced Knowledge Distillation for NMT

To address the aforementioned issues in word-level KD, in this section, we introduce our method named **T**op-1 **I**nformation **E**nhanced **K**nowledge **D**istillation (TIE-KD), which includes a hierarchical ranking loss to boost the learning of the top-1 information from the teacher (§4.1) and an iterative knowledge distillation procedure to exploit more knowledge from the teacher (§4.2).

4.1 Hierarchical Ranking Loss

To help student models better grasp the top-1 information during distillation, we design a new loss named hierarchical ranking loss. To gently achieve this goal, we first encourage the student to rank the teacher's top-k predictions as its own top-k predictions and then rank the teacher's top-1 prediction over these top-k predictions. Formally, given the student's top-k predictions as $[y_j^{s_1}, ..., y_j^{s_k}]$ and the teacher's top-k predictions as $[y_j^{t_1}, ..., y_j^{t_k}]$, the hierarchical ranking loss $\mathcal{L}_{hr}$ can be expressed as:

$$\mathcal{L}_{hr} = \sum_{j}^{N} \Big(\sum_{u}^{k} \sum_{v}^{k} \max \{0, \mathbb{1}\{q(y_j^{t_u}) > q(y_j^{s_v})\}(p(y_j^{s_v}) - p(y_j^{t_u}))\} + \sum_{u}^{k} \max \{0, p(y_j^{t_u}) - p(y_j^{t_1})\} \Big), \quad (6)$$

where $p(\cdot)$ and $q(\cdot)$ are the probabilities from the student model and the teacher model, respectively. And $\mathbb{1}\{\cdot\}$ is an indicator function.

In this way, the student model can be enforced to rank the top-1 predictions of the teacher to its own top-1 places, and thus it can explicitly enhance the learning of the knowledge from the teacher. Then, we add this loss to the original KL divergence loss, *i.e.*, Eq.(2), forming a new loss for KD:

$$\mathcal{L}_{kd} = \mathcal{L}_{\text{KL}} + \mathcal{L}_{hr}. \quad (7)$$

Algorithm 1 Iterative Knowledge Distillation

```
Input: source and target data in current mini-batch $(\mathbf{x}, \mathbf{y})$; student model $\mathcal{S}$; teacher model $\mathcal{T}$; iteration times $N$;
Initialize $\mathbf{y}^0 = \mathbf{y}$; $\mathcal{L}_{kd} = 0$;
Compute $\mathcal{L}_{ce}$ based on Eq.(1)
for $i$ in $1, 2, ..., N$ do
    $p^i = \mathcal{S}(\mathbf{x}; \mathbf{y}^{i-1})$    ▷ probability distributions from the student model
    $q^i = \mathcal{T}(\mathbf{x}; \mathbf{y}^{i-1})$    ▷ probability distributions from the teacher model
    Compute $\mathcal{L}_{kd}^i(p^i, q^i)$ based on Eq.(7)
    $\mathcal{L}_{kd} \leftarrow \mathcal{L}_{kd} + \mathcal{L}_{kd}^i$
    $\mathbf{y}^i = \arg\max(p^i)$    ▷ student predictions as inputs in the next iteration
end for
$\mathcal{L}_{word\text{-}kd} \leftarrow (1 - \alpha)\mathcal{L}_{ce} + \frac{\alpha}{N}\mathcal{L}_{kd}$
```

4.2 Iterative Knowledge Distillation

Given that the large overlap between the top-1 predictions and ground-truth targets limits the amount of additional knowledge from the teacher during word-level KD, introducing data without ground-truth targets for KD could be helpful to mitigate this issue. Inspired by previous studies on decoder-side data manipulation (Zhang et al., 2019; Goodman et al., 2020; Liu et al., 2021a,b; Xie et al., 2021), we design an iterative knowledge distillation procedure to expose more target-side data for KD.

Methods	**WMT'14 En-De**		**WMT'14 En-Fr**		**WMT'16 En-Ro**	
	BLEU	COMET	BLEU	COMET	BLEU	COMET
Student (*Transformer*$_{base}$)	$27.42_{\pm 0.01}$	$48.11_{\pm 1.04}$	$40.97_{\pm 0.14}$	$62.19_{\pm 0.11}$	$33.59_{\pm 0.15}$	$50.96_{\pm 0.43}$
+ Word-KD (Kim and Rush, 2016)	$28.03_{\pm 0.10}$	$51.59_{\pm 0.23}$	$41.10_{\pm 0.11}$	$63.81_{\pm 0.14}$	$33.77_{\pm 0.01}$	$53.15_{\pm 0.26}$
+ Seq-KD (Kim and Rush, 2016)	$28.22_{\pm 0.02}$	$51.23_{\pm 0.15}$	$41.44_{\pm 0.02}$	$63.12_{\pm 0.14}$	$33.69_{\pm 0.02}$	$50.63_{\pm 0.11}$
+ BERT-KD (Chen et al., 2020)†	27.53	-	-	-	-	-
+ Seer Forcing (Feng et al., 2021)	$27.56_{\pm 0.10}$	$50.60_{\pm 0.12}$	$40.97_{\pm 0.01}$	$62.95_{\pm 0.39}$	$33.77_{\pm 0.09}$	$51.41_{\pm 0.60}$
+ CBBGCA (Zhou et al., 2022)†	28.36	-	41.54	-	-	-
+ Annealing KD (Jafari et al., 2021)	$27.91_{\pm 0.10}$	$51.58_{\pm 0.03}$	$41.20_{\pm 0.13}$	$63.59_{\pm 0.09}$	$33.67_{\pm 0.09}$	$52.22_{\pm 1.02}$
+ Selective-KD (Wang et al., 2021)	$28.24_{\pm 0.21}$	$52.15_{\pm 0.42}$	$41.25_{\pm 0.04}$	$64.24_{\pm 0.01}$	$33.74_{\pm 0.02}$	$53.05_{\pm 0.28}$
+ TIE-KD (ours)	$\mathbf{28.46}^{*}_{\pm 0.01}$	$\mathbf{52.63}^{*}_{\pm 0.09}$	$\mathbf{41.57}^{*}_{\pm 0.08}$	$\mathbf{65.06}^{*}_{\pm 0.44}$	$\mathbf{34.70}^{*}_{\pm 0.07}$	$\mathbf{55.76}^{*}_{\pm 0.21}$
Teacher (*Transformer*$_{big}$)	*28.81*	*53.20*	*42.98*	*69.58*	*34.70*	*57.04*

Table 6: BLEU scores (%) and COMET (Rei et al., 2020) scores (%) on three translation tasks. Results with † are taken from the original papers. Others are our re-implementation results using the released code with the same setting in Sec.5.2 for a fair comparison. We report average results over 3 runs with random initialization. Results with $*$ are statistically (Koehn, 2004) better than the vanilla Word-KD with $p < 0.01$.

Specifically, as shown in Algorithm 1, at each training step, we conduct KD for N iterations (line 3), by using the predictions of the student in the current iteration as the decoder-side inputs for KD in the next iteration (line 8). Generally, these predictions can be regarded as similar but new inputs compared to the original target inputs. Meanwhile, there is no ideal ground-truth target for these inputs since they are usually not well-formed sentences. Then during each iteration, we collect the loss of KD according to Eq.(7) (lines 4∼7) and average it across all the iterations (line 10). Since all the supervision signals are from the teacher after the first iteration, the knowledge of the teacher model will be more exploited during the following iterations and thus the potential of word-level KD can be more released.

5 Experiments

5.1 Datasets

We conduct experiments on three commonly-used WMT tasks, *i.e.*, the WMT'14 English to German (En-De), WMT'14 English to French (En-Fr) and WMT'16 English to Romanian (En-Ro). For all these tasks, we share the source and the target vocabulary and segment words into subwords using byte pair encoding (BPE) (Sennrich et al., 2016) with 32k merge operations. More statistics of the datasets can be found in Appendix C.1.

5.2 Implementation Details

All our experiments are conducted based on the open-source toolkit fairseq (Ott et al., 2019) with FP16 training (Ott et al., 2018). By default, we follow the big/base setting (Vaswani et al., 2017) to implement the teacher/student models in our experiments. More training and evaluation details can be referred to Appendix C.2. For word-level KD-based methods, we set the α in Eq.(3) to 0.5 following Kim and Rush (2016). For our method, we set top-k in Sec.4.1 to 5 and iteration time N in Sec.4.2 to 3 on all three tasks. The selection of top-k and N are shown in Appendix D.

5.3 Main Results

We compare our proposed method with existing KD techniques in NMT (the detailed description of these compared techniques can be referred to Appendix C.3) on three WMT tasks. To make the results more convincing, we report both BLEU and COMET (Rei et al., 2020) scores in Tab.6. Using Transformer$_{big}$ as the teacher, our method can boost the Transformer$_{base}$ students by +1.04/+0.60/+1.11 BLEU scores and +4.52/+2.57/+4.80 COMET scores on three tasks, respectively. Compared to the vanilla Word-KD baseline, our method can outperform it significantly on all translation tasks, which verifies the effectiveness of our proposed solutions. Additionally, as a word-level KD method, our TIE-KD can outperform Seq-KD on all three tasks and even achieves fully competitive results with the teacher on En-Ro, which demonstrates that the potential of Word-KD can be largely released by our method.

6 Analysis

6.1 Ablation Study

To separately verify the effectiveness of our solutions for the two issues in vanilla word-level KD, we conduct an ablation study on WMT'14 En-De

Methods	Validation Set		Test Set	
	BLEU	TA	BLEU	TA
vanilla Word-KD	26.66	88.98	28.03	88.46
+ $\mathcal{L}_{hr}$	26.96	89.30	28.25	88.93
+ iterative KD	27.02	89.16	28.28	88.74
+ both (TIE-KD)	**27.13**	**89.50**	**28.46**	**89.11**

Table 7: Ablation study on the WMT'14 En-De task.

Methods	BLEU	Δ
Student (Transformer$_{base}$)	27.42	ref.
Word-KD	28.03	+0.61
Seq-KD	28.22	+0.80
TIE-KD	28.46	+1.04
Word-KD + Seq-KD	28.48	+1.06
TIE-KD + Seq-KD	**28.66**	**+1.24**
Teacher (Transformer$_{big}$)	28.81	+1.39

Table 8: Combination with sequence-level KD and word-level KD methods on the WMT'14 En-DE task.

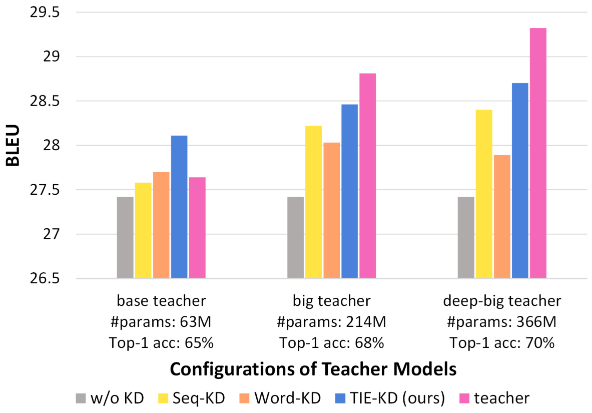


Figure 3: Performance of KD techniques with different teacher models on the test set of the WMT'14 En-De task.

task and record the results in Tab.7. When only adding hierarchical ranking loss to vanilla word-level KD, the BLEU scores and the TA rates gain by +0.3/+0.22 and +0.32/+0.47 on the validation/test set, respectively. It reflects that KL divergence only provides a loose constraint on the learning of the top-1 information from the teacher, while our hierarchical ranking loss helps to explicitly grasp this core information. When only using iterative KD, the student also improves by +0.36/+0.25 BLEU scores and +0.18/+0.28 TA rates. It indicates that our iterative KD can effectively release the potential of word-level KD by introducing data without ground-truth targets. When combined together, the two solutions finally compose our TIE-KD and can yield further improvement on both metrics. Therefore, the two issues in word-level KD are orthogonal and our proposed solutions are complementary to each other.

6.2 Combination With Sequence-Level KD

According to (Kim and Rush, 2016), word-level KD can be well combined with sequence-level KD and yields better performance. As a word-level KD approach, our TIE-KD can also theoretically be combined with sequence-level KD. We verify this on the WMT'14 En-De task and list the results in Tab.8. Like Word-KD, our TIE-KD can also achieve better performance when combined with Seq-KD and is also better than "Word-KD + Seq-KD", indicating the superiority of our method and its high compatibility with sequence-level KD.

6.3 Can a Stronger Teacher Teach a Better Student in NMT?

Among the prior literature on KD (Cho and Hariharan, 2019; Jin et al., 2019; Mirzadeh et al., 2020; Guo et al., 2020; Jafari et al., 2021; Qiu et al., 2022), a general consensus is that a large teacher-student capacity gap may harm the quality of KD. We also check this problem in NMT by using teachers of three model sizes. Besides the default configuration (*i.e.*, Transformer$_{big}$) in our experiments above, we also add Transformer$_{base}$ setting as the weaker teacher and Transformer$_{deep\text{-}big}$ setting with 18 encoder layers and 6 decoder layers as the stronger teacher[8]. We compare our method with word- and sequence-level KD under these teachers in Fig.3 and draw several conclusions:

(1) The stronger teacher can bring improvement to sequence-level KD but fails to word-level KD, where the reason may be the less additional knowledge from the stronger teacher due to its higher top-1 accuracy (68%→70%).

(2) As a word-level KD method, our TIE-KD instead brings conspicuous improvement with the stronger teacher, indicating that our method can exploit more knowledge from the teacher.

(3) Under the weaker teacher, the student from our method even significantly surpasses the teacher, while other methods are largely limited by the performance of the teacher, demonstrating the

[8] To stably train a deeper Transformer, we use Admin (Liu et al., 2020) in layer normalization.

high generalizability of our TIE-KD to different teacher-student capacity gaps.

6.4 Why is the Top-1 Information Important in KD?

The decoding process of language generation models can be regarded as a sequential decision-making process (Yu et al., 2017; Arora et al., 2022). As mentioned in Sec.3.5, during decoding, beam search tends to pick the top-1 predictions of the NMT model on each beam and finally selects the most probable beam. Thus, the top-1 information (including both the top-1 word index and its corresponding probability) of the teacher model largely represents its decision on each decoding step, which is exactly what we expect the student model to learn from the teacher through KD in NMT. Therefore, the top-1 information can be seen as the embodiment of the knowledge of the teacher model in NMT tasks and should be emphatically learned by the student models.

7 Related Work

Kim and Rush (2016) first introduce word-level KD for NMT and further propose sequence-level KD for better performance. Afterward, Wang et al. (2021) investigate the effectiveness of different types of tokens in KD and propose selective KD strategies. Moreover, Wu et al. (2020) distill the internal hidden states of the teacher models into the students and also obtain promising results. In the field of non-autoregressive machine translation (NAT), KD from autoregressive models has become a *de facto* standard to improve the performance of NAT models (Gu et al., 2017; Zhou et al., 2019; Gu et al., 2019). Also, KD has been used to enhance the performance of multilingual NMT (Tan et al., 2019; Sun et al., 2020). Besides, similar ideas can be found when introducing external information to NMT models. For example, Baziotis et al. (2020) use language models as teachers for low-resource NMT models. Chen et al. (2020) distill the knowledge from fine-tuned BERT into NMT models. Feng et al. (2021) and Zhou et al. (2022) leverage KD to introduce future information to the teacher-forcing training of NMT models.

Differently, in this work, 1) we aim to explore where the knowledge hides in KD and unveil that it comes from the top-1 information of the teacher and further improve KD from this perspective; 2) we try to build a connection between two kinds of KD techniques in NMT and reveal their common essence, providing new directions for future work.

8 Conclusion

In this paper, we explore where the knowledge hides in KD for NMT and unveil that it comes from the top-1 information of the teacher. This finding reflects the connection between word- and sequence-level KD and reveals the common essence of both KD techniques in NMT. From this perspective, we further propose a top-1 information enhanced knowledge distillation (TIE-KD) to address the two issues in vanilla word-level KD. Experiments on three WMT tasks prove the effectiveness of our method. Besides, we investigate the performance of the existing KD techniques in NMT and our method under different teacher-student capacity gaps and show the stronger generalizability of our method on various gaps.

Limitations

Although our method has achieved outstanding performance compared to current KD techniques, it is still a word-level KD method and also suffers from some limitations in vanilla word-level KD, *e.g.*, the *exposure bias* as analyzed in Appendix A. How to design a unified and more powerful KD method from the perspective of the connection between word- and sequence-level KD still remains unsolved. We will leave this for the future work. Moreover, our study focuses on the mainstream KD techniques in NMT, which transfer knowledge through teachers' predictions, while some other KD techniques, like directly distilling the hidden states (Wu et al., 2020), are not within the scope of this study and thus not included.

Acknowledgements

The research work described 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 and 61976015). Wenjuan Han is supported by the Talent Fund of Beijing Jiaotong University (2023XKRC006). The authors would like to thank the anonymous reviewers for their valuable comments and suggestions to improve this paper.

References

Kushal Arora, Layla El Asri, Hareesh Bahuleyan, and Jackie Cheung. 2022. Why exposure bias matters: An imitation learning perspective of error accumulation in language generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 700–710, Dublin, Ireland. Association for Computational Linguistics.

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

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.

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. *Advances in neural information processing systems*, 28.

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, Online. Association for Computational Linguistics.

Jang Hyun Cho and Bharath Hariharan. 2019. On the efficacy of knowledge distillation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4794–4802.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yang Feng, Shuhao Gu, Dengji Guo, Zhengxin Yang, and Chenze Shao. 2021. Guiding teacher forcing with seer forcing 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 1: Long Papers)*, pages 2862–2872, Online. 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.

Sebastian Goodman, Nan Ding, and Radu Soricut. 2020. Teaforn: Teacher-forcing with n-grams. *arXiv preprint arXiv:2010.03494*.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor OK Li, and Richard Socher. 2017. Non-autoregressive neural machine translation. *arXiv preprint arXiv:1711.02281*.

Jiatao Gu, Changhan Wang, and Junbo Zhao. 2019. Levenshtein transformer. *Advances in Neural Information Processing Systems*, 32.

Jia Guo, Minghao Chen, Yao Hu, Chen Zhu, Xiaofei He, and Deng Cai. 2020. Reducing the teacher-student gap via spherical knowledge disitllation. *arXiv preprint arXiv:2010.07485*.

Geoffrey Hinton, Oriol Vinyals, Jeff Dean, et al. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2(7).

Aref Jafari, Mehdi Rezagholizadeh, Pranav Sharma, and Ali Ghodsi. 2021. Annealing knowledge distillation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2493–2504, Online. Association for Computational Linguistics.

Xiao Jin, Baoyun Peng, Yichao Wu, Yu Liu, Jiaheng Liu, Ding Liang, Junjie Yan, and Xiaolin Hu. 2019. Knowledge distillation via route constrained optimization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1345–1354.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. *arXiv preprint arXiv:1606.07947*.

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

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.

Solomon Kullback and Richard A Leibler. 1951. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86.

Bei Li, Ziyang Wang, Hui Liu, Yufan Jiang, Quan Du, Tong Xiao, Huizhen Wang, and Jingbo Zhu. 2020. Shallow-to-deep training for neural machine translation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 995–1005, Online. Association for Computational Linguistics.

Percy Liang, Hal Daumé III, and Dan Klein. 2008. Structure compilation: trading structure for features. In *Proceedings of the 25th international conference on Machine learning*, pages 592–599.

Yunlong Liang, Fandong Meng, Yufeng Chen, Jinan Xu, and Jie Zhou. 2021a. Modeling bilingual conversational characteristics for neural chat 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 5711–5724, Online. Association for Computational Linguistics.

Yunlong Liang, Fandong Meng, Jinan Xu, Yufeng Chen, and Jie Zhou. 2022. Scheduled multi-task learning for neural chat translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4375–4388, Dublin, Ireland. Association for Computational Linguistics.

Yunlong Liang, Chulun Zhou, Fandong Meng, Jinan Xu, Yufeng Chen, Jinsong Su, and Jie Zhou. 2021b. Towards making the most of dialogue characteristics for neural chat translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 67–79, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Liyuan Liu, Xiaodong Liu, Jianfeng Gao, Weizhu Chen, and Jiawei Han. 2020. Understanding the difficulty of training transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5747–5763, Online. Association for Computational Linguistics.

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.

Sachin Mehta, Marjan Ghazvininejad, Srinivasan Iyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2020. Delight: Deep and light-weight transformer. *arXiv preprint arXiv:2008.00623*.

Seyed Iman Mirzadeh, Mehrdad Farajtabar, Ang Li, Nir Levine, Akihiro Matsukawa, and Hassan Ghasemzadeh. 2020. Improved knowledge distillation via teacher assistant. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 5191–5198.

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.

Zengyu Qiu, Xinzhu Ma, Kunlin Yang, Chunya Liu, Jun Hou, Shuai Yi, and Wanli Ouyang. 2022. Better teacher better student: Dynamic prior knowledge for knowledge distillation. *arXiv preprint arXiv:2206.06067*.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. 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, Berlin, Germany. Association for Computational Linguistics.

Haipeng Sun, Rui Wang, Kehai Chen, Masao Utiyama, Eiichiro Sumita, and Tiejun Zhao. 2020. Knowledge distillation for multilingual unsupervised neural machine translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3525–3535, Online. Association for Computational Linguistics.

Xu Tan, Yi Ren, Di He, Tao Qin, Zhou Zhao, and Tie-Yan Liu. 2019. Multilingual neural machine translation with knowledge distillation. *arXiv preprint arXiv:1902.10461*.

Jiaxi Tang, Rakesh Shivanna, Zhe Zhao, Dong Lin, Anima Singh, Ed H Chi, and Sagar Jain. 2020. Understanding and improving knowledge distillation. *arXiv preprint arXiv:2002.03532*.

A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. 2017. Attention is all you need. *arXiv*.

Fusheng Wang, Jianhao Yan, Fandong Meng, and Jie Zhou. 2021. Selective knowledge distillation 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 1: Long Papers)*, pages 6456–6466, Online. Association for Computational Linguistics.

Hongyu Wang, Shuming Ma, Li Dong, Shaohan Huang, Dongdong Zhang, and Furu Wei. 2022. Deepnet: Scaling transformers to 1,000 layers. *arXiv preprint arXiv:2203.00555*.

Xiangpeng Wei, Heng Yu, Yue Hu, Yue Zhang, Rongxiang Weng, and Weihua Luo. 2020. Multiscale collaborative deep models for neural machine translation. In *Proceedings of the 58th Annual Meeting of* *the Association for Computational Linguistics*, pages 414–426, Online. Association for Computational Linguistics.

Yimeng Wu, Peyman Passban, Mehdi Rezagholizade, and Qun Liu. 2020. Why skip if you can combine: A simple knowledge distillation technique for intermediate layers. *arXiv preprint arXiv:2010.03034*.

Shufang Xie, Ang Lv, Yingce Xia, Lijun Wu, Tao Qin, Tie-Yan Liu, and Rui Yan. 2021. Target-side input augmentation for sequence to sequence generation. In *International Conference on Learning Representations*.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient.

Wen Zhang, Yang Feng, Fandong Meng, Di You, and Qun Liu. 2019. Bridging the gap between training and inference for neural machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4334–4343, Florence, Italy. Association for Computational Linguistics.

Chulun Zhou, Fandong Meng, Jie Zhou, Min Zhang, Hongji Wang, and Jinsong Su. 2022. Confidence based bidirectional global context aware training framework for neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2878–2889, Dublin, Ireland. Association for Computational Linguistics.

Chunting Zhou, Graham Neubig, and Jiatao Gu. 2019. Understanding knowledge distillation in non-autoregressive machine translation. *arXiv preprint arXiv:1911.02727*.

A A Theoretical Analysis on the Connection Between Word- and Sequence-level KD

We can directly consider the KL divergence loss of word-level KD in Eq.(2) as its training objective and convert it into the equivalent form of the cross-entropy loss. For simplicity, we omit the θ_t in $q(\cdot)$ and θ_s in $p(\cdot)$ in following formulas:

$$\begin{aligned}\mathcal{L}_{kd}^{word} =& \sum_{j=1}^{N} D_{\mathrm{KL}}\Big(q(y_j|\mathbf{y}_{<j},\mathbf{x})\big|\big|p(y_j|\mathbf{y}_{<j},\mathbf{x})\Big)\\ \Leftrightarrow& -\sum_{j=1}^{N}\sum_{k\in\mathcal{V}} q(y_j=k|\mathbf{y}_{<j},\mathbf{x})\times\\ &\log p(y_j=k|\mathbf{y}_{<j},\mathbf{x}),\end{aligned} \tag{8}$$

where $\mathcal{V}$ denotes the whole target-side vocabulary. Then we can further separate the cross-entropy loss into the loss on the top-1 prediction y_j^{t1} and the losses on other candidates in the vocabulary:

$$\begin{aligned}\mathcal{L}_{kd}^{word} =& -\sum_{j=1}^{N}\sum_{k\in\mathcal{V}} q(y_j=k|\mathbf{y}_{<j},\mathbf{x})\times\\ &\log p(y_j=k|\mathbf{y}_{<j},\mathbf{x})\\ =& -\sum_{j=1}^{N}\Big(q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})\log p(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})\\ &+\sum_{k\in\mathcal{V}\setminus\{y_j^{t_1}\}} q(y_j=k|\mathbf{y}_{<j},\mathbf{x})\times\\ &\log p(y_j=k|\mathbf{y}_{<j},\mathbf{x})\Big)\\ =& -\sum_{j=1}^{N}\Big(q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})\log p(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})\\ &+R(y_j^{t_1})\Big),\end{aligned} \tag{9}$$

where $R(y_j^{t_1})$ represents the cross-entropy loss on the remaining candidates except for the top-1 prediction $y_j^{t_1}$ and can be regarded as a regularization term for the former one. As empirically verified in Sec.3, we can do the following approximation by omitting $R(y_j^{t_1})$ in Eq.(9):

$$\begin{aligned}\mathcal{L}_{kd}^{word} =& -\sum_{j=1}^{N}\Big(q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})\log p(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})\\ &+R(y_j^{t_1})\Big)\\ \approx& -\sum_{j=1}^{N} q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})\log p(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x}).\end{aligned} \tag{10}$$

Thus, we obtain the approximate form of the training objective of word-level KD.

Now we consider the training objective of sequence-level KD in Eq.(5). According to the results in Sec.3.5, we can also assume that optimizing using all targets is approximately equal to optimizing using top-1 targets:

$$\begin{aligned}\mathcal{L}_{kd}^{seq} =& -\sum_{j=1}^{N}\log p(\widehat{y}_j|\widehat{\mathbf{y}}_{<j},\mathbf{x})\\ \approx& -\sum_{j=1}^{N}\mathbb{1}\{\widehat{y}_j=y_j^{t_1}\}\log p(y_j^{t_1}|\widehat{\mathbf{y}}_{<j},\mathbf{x}),\end{aligned} \tag{11}$$

where $\mathbb{1}\{\cdot\}$ is an indicator function.

Lastly, if we replace the different weight functions before the $\log(\cdot)$ function in Eq.(10) and Eq.(11) with one function $f(\cdot)$:

$$f(j)=\begin{cases} q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x}), & \textit{word-level}\\ \mathbb{1}\{\widehat{y}_j=y_j^{t_1}\}, & \textit{sequence-level},\end{cases}$$

then we can derive a unified form of the objective for these two kinds of KD techniques:

$$\mathcal{L}_{kd}^{uni} = -\sum_{j=1}^{N} f(j)\log p(y_j^{t_1}|\widetilde{\mathbf{y}}_{<j},\mathbf{x}), \tag{12}$$

where $\widetilde{\mathbf{y}}_{<j}$ is the golden context $\mathbf{y}_{<j}$ in word-level KD and the model-generated context $\widehat{\mathbf{y}}_{<j}$ in sequence-level KD.

In this unified form, the only two variables are the weight function $f(\cdot)$ and the target-side previous context $\widetilde{\mathbf{y}}_{<j}$ in the condition of the probability $p(\cdot)$. From this expression, it is clear that student models are encouraged to learn the top-1 predictions of the teacher to obtain teachers' knowledge at each time step in both KD techniques. Therefore, we claim that the working mechanisms behind the

two kinds of KD techniques are the same to some extent, although they look quite distinct on the surface.

Notably, we also conjecture that the context difference may explain why sequence-level KD generally outperforms word-level KD. Autoregressive models trained with teacher-forcing suffer from *exposure bias* due to the gap between the golden context in training and the model-generated context in inference (Bengio et al., 2015; Zhang et al., 2019). According to the above analysis, the same thing also happens in word-level KD. However, sequence-level KD circumvents this problem by conditioning on model-generated contexts during distillation, thus leaving no gap between training and inference. This conjecture can also be verified by the performance of sequence-level KD on WMT'16 En-Ro, where the teacher's translations achieve considerably high similarities (BLEU score > 62) with the original target sentences, and the improvement brought by sequence-level KD is much less than the one on other datasets since the model-generated context is too close to the golden context.

B Why Not Re-normalize the Soft Target in "*w/o* correlation info"?

We would like to explain this from the perspective of the loss function. As we analyzed in Eq.(9), the loss of vanilla word-level KD is:

$$\mathcal{L}_{kd}^{word} = -\sum_{j=1}^{N} \Big(q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x}) \log p(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x}) + \sum_{k\in\mathcal{V}\setminus\{y_j^{t_1}\}} q(y_j = k|\mathbf{y}_{<j},\mathbf{x})\times \log p(y_j = k|\mathbf{y}_{<j},\mathbf{x})\Big). \quad (13)$$

Based on this, we remove all other probabilities in the soft target of the teacher except for the top-1 one to remove the "correlation information", *i.e.*, the second term of the loss in Eq.(13) is discarded:

$$\mathcal{L}_{kd}^{nocorr} = -\sum_{j=1}^{N} q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x}) \log p(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x}).$$

In this objective, the effect of KD is fully dominated by the top-1 information of the teacher. If we try to re-normalize the soft target with an additional uniform distribution, the result of KD will be affected by the regularization term of this uniform distribution:

$$\mathcal{L}_{kd}^{nocorr} = -\sum_{j=1}^{N} \Big(q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x}) \log p(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x}) \underbrace{+\, u \sum_{k\in\mathcal{V}\setminus\{y_j^{t_1}\}} \log p(y_j = k|\mathbf{y}_{<j},\mathbf{x})\Big)}_{uniform\ regularization},$$

where $u = \frac{1-q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})}{|\mathcal{V}|-1}$. Another way to re-normalize the distribution is to directly let $q(y_j^{t_1}|\mathbf{y}_{<j},\mathbf{x})$ as 1, but the original top-1 probability information from the teacher will be lost. Therefore, we keep the modified soft target in "*w/o* correlation info" unnormalized.

C Experimental Details

C.1 Statistics of the Datasets

For the En-De task, the training data contains nearly 4.5M sentence pairs. We choose *newstest2013* and *newstest2014* as the validation set and the test set, respectively. For the En-Fr task, there totally remains 35.8M sentence pairs after the cleaning procedure. Then we choose *newstest2013* and *newstest2014* as the validation set and the test set, respectively. For the En-Ro task, we directly use the pre-processed data from Mehta et al. (2020) and there are about 608K sentence pairs in the training data. Then *newsdev2016* is selected as the validation set and *newstest2016* is the test set. The overall statistics of the datasets are listed in Table 9.

Dataset	#Train	#Valid	#Test	Vocab
WMT'14 En-De	4.5M	3000	3003	37184
WMT'14 En-Fr	35.8M	3000	3003	36528
WMT'16 En-Ro	608K	1999	1999	34976

Table 9: Statistics of the datasets for three WMT tasks.

C.2 Implementation Details and Model Configurations

Training. To assure the reproducibility of our experimental results, we provide comprehensive training details and model configurations of our experiments in Tab.10. All our experiments are conducted on 4 NVIDIA RTX 3090 GPUs with gradient accumulation step 2, 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. To obtain strong

Hyperparameters	WMT'14 En-De		WMT'14 En-Fr		WMT'16 En-Ro	
	Student	Teacher	Student	Teacher	Student	Teacher
Embedding Dim	512	1024	512	1024	512	1024
FFN Dim	2048	4096	2048	4096	2048	4096
Encoder Layers	6	6	6	6	6	6
Decoder Layers	6	6	6	6	6	6
Attention Heads	8	16	8	16	8	16
Residual Dropout	0.1	0.3	0.1	0.3	0.1	0.3
Attention Dropout	0.1	0.1	0.1	0.1	0.1	0.1
Activation Dropout	0.1	0.1	0.1	0.1	0.1	0.1
Label Smoothing	0.1	0.1	0.1	0.1	0.1	0.3
Learning Rate	7e-4	5e-4	7e-4	5e-4	7e-4	5e-4
Learning Rate Decay	inverse sqrt	inverse sqrt	inverse sqrt	inverse sqrt	inverse sqrt	inverse sqrt
Warmup Steps	4000	4000	4000	4000	4000	4000
Layer Normalization	PostNorm	PostNorm	PostNorm	PostNorm	PostNorm	PostNorm
Model Parameters	63.2M	214.4M	62.8M	213.8M	62.0M	212.2M
Training Steps	200K	300K	200K	300K	20 epochs	30 epochs

Table 10: Training hyperparameters and model configurations of our experiments.

teachers and enlarge the gaps between teacher models and student models, we train teachers for 50% more steps than the corresponding students. Then we use the checkpoint with the highest BLEU of the teacher on the validation set to conduct distillation.

Evaluation. During inference, we set beam size to 4 and length penalty to 0.6 for En-De and En-Fr. For En-Ro, we set beam size to 5 and length penalty to 1.2. For a more convincing evaluation, we use *multibleu.perl* to calculate case-sensitive BLEU and *unlabel-comet*[9] to calculate COMET scores (Rei et al., 2020) for all three tasks. For student models, we average the last 5 checkpoints for evaluation following Vaswani et al. (2017). We use the paired bootstrap resampling methods (Koehn, 2004) for the statistical significance test. For the En-De task and the En-Fr task, we evaluate and save the checkpoint every 5000 training steps. For the En-Ro task, since the models tend to overfit, we only train students for 20 epochs and save the checkpoint after every epoch.

C.3 Compared Systems and Hyperparameters

Transformer. We follow the standard base/big model configurations (Vaswani et al., 2017) to implement the student/teacher models.

Word-KD. The standard method to conduct word-level KD in NMT proposed by Kim and Rush (2016).

Seq-KD. Kim and Rush (2016) also propose a sequence-level KD approach that directly substitutes the original target-side training data with the translations of the teacher from beam search. In our experiments, the hyperparameters of beam search keep the same with the inference stage.

BERT-KD. Chen et al. (2020) propose to distill the knowledge from BERT (Devlin et al., 2018) for text generation tasks.

Seer Forcing. Feng et al. (2021) design a seer forcing method for NMT to distill future information to the teacher forcing. Following the suggestion in (Feng et al., 2021), we set the α in their paper to 0.5 for both En-De and En-Fr, and 0.25 for En-Ro. Besides, we set the seer dropout to 0.1 for En-De and En-Fr and 0.2 for En-Ro.

CBBGCA. Zhou et al. (2022) also propose to distill bi-directional contextual information in CMLM for uni-directional training of NMT based on the confidence of the NMT model.

Annealing KD. Our implementation of the method in (Jafari et al., 2021) which gradually anneals the temperature of the teacher during KD. Different from the original paper, we use the KL divergence as the loss function of KD instead of Mean Squared Error (MSE) due to its better performance on NMT tasks. In our carefully chosen recipe, we set the max temperature to 1.1 and gradually reduce it to 1.0 during the first 2/3 epochs. Then we use vanilla CE loss to train the student

[9] https://github.com/Unbabel/COMET

model for the remaining 1/3 epochs.

Selective-KD. Wang et al. (2021) investigate the effectiveness of different data for distillation and propose a knowledge selection method for selecting more valuable data for word-level KD. In our experiments, we choose the global-level selection that performs better according to Wang et al. (2021).

D Hyperparameter Selection

D.1 Effect of Hierarchical Ranking Range k

In this section, we investigate the effect of k in hierarchical ranking loss on our method. We search k in [3, 5, 10, 20] and compare their performance on the validation set of the WMT'14 En-De task. As shown in Fig.4, our method performs best when k is set to 5. Thus, we keep k to 5 for all three tasks in our experiments.

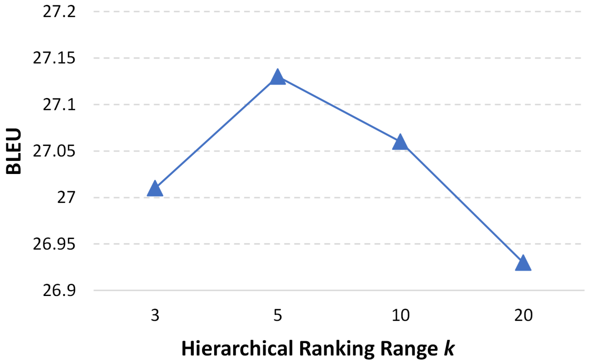


Figure 4: BLEU scores of our method with different k on the validation set of the WMT'14 En-De task.

D.2 Effect of Iteration Times N

Since our method includes several iterations of KD, we further investigate the effects of the iteration times on the performance of our method. Intuitively, with more iteration times, more knowledge will be exploited from the teacher, while the computational cost will also increase. To check this, we try each iteration time in [1, 2, 3, 4] and record the corresponding performance and training time in Fig.5. It is obvious that the performance of our method gradually improves with N increasing, while the training time per step also linearly increases. Balancing the cost and the performance, we choose 3 as the final iteration time.

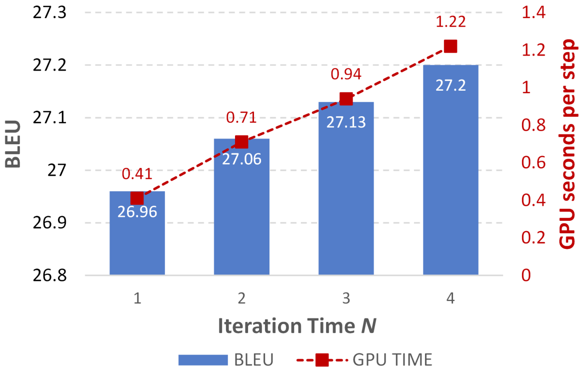


Figure 5: BLEU scores of our method with different iteration times N on the validation set of the WMT'14 En-De task and the corresponding training costs.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section 9 (Limitations)

□ A2. Did you discuss any potential risks of your work?
Not applicable. Left blank.

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section 0,1 (Abstract & Introduction)

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 5,6

☑ B1. Did you cite the creators of artifacts you used?
Section 5,6

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 5,6

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 5,6

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Section 5, Appendix B

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Appendix B

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Appendix B

C ☑ Did you run computational experiments?

Section 1,3,5, Appendix B,C,D,E

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Appendix B

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 5, Appendix C

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 5

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Appendix B

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

□ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

□ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

□ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

□ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

□ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.