Proceedings of Machine Learning Research 129:321–336, 2020 ACML 2020

Dual Learning: Theoretical Study and an Algorithmic Extension

Zhibing Zhao* ZHIBING.ZHAO@MICROSOFT.COM
Microsoft, Bellevue, WA, US

Yingce Xia YINGCE.XIA@MICROSOFT.COM
Microsoft Research Asia, Beijing, China

Tao Qin TAOQIN@MICROSOFT.COM
Microsoft Research Asia, Beijing, China

Lirong Xia XIAL@CS.RPI.EDU
Rensselaer Polytechnic Institute, Troy, NY, US

Tie-Yan Liu TYLIU@MICROSOFT.COM
Microsoft Research Asia, Beijing, China

Editors: Sinno Jialin Pan and Masashi Sugiyama

Abstract

Dual learning has been successfully applied in many machine learning applications including machine translation, image-to-image transformation, etc. The high-level idea of dual learning is very intuitive: if we map an x from one domain to another and then map it back, we should recover the original x. Although its effectiveness has been empirically verified, theoretical understanding of dual learning is still very limited. In this paper, we aim at understanding why and when dual learning works. Based on our theoretical analysis, we further extend dual learning by introducing more related mappings and propose multi-step dual learning, in which we leverage feedback signals from additional domains to improve the qualities of the mappings. We prove that multi-step dual learning can boost the performance of standard dual learning under mild conditions. Experiments on WMT 14 English↔German and MultiUN English↔French translations verify our theoretical findings on dual learning, and the results on the translations among English, French, and Spanish of MultiUN demonstrate the effectiveness of multi-step dual learning.

Keywords: Neural machine translation, dual learning, semi-supervised learning

1. Introduction

Most machine learning tasks can be formulated as learning a mapping from one domain to another one, including image classification (from image to label), neural machine translation (from the source language to the target language), speech recognition (from voice to text), etc. Among them, many tasks are of dual forms, like image classification v.s. image generation (from label to image), the neural machine translation between two languages (e.g., English→French v.s. French→English), speech recognition v.s. speech synthesis (from text to voice), etc. Such duality can be utilized to improve the model qualities.

One prominent framework is dual learning, first proposed by He et al. (2016) for machine translation and then applied to many other applications like image translation (Kim et al., 2017;

* This work was conducted at Microsoft Research Asia.

Zhu et al., 2017), question answering and generation (Tang et al., 2017), etc. In dual learning, two mapping functions between two domains are trained simultaneously so that one function is close to the inverse of the other. The intuition is that, if we translate a sentence from English to French and then translate the obtained French sentence back to English, we should get the same sentence or a very similar one. Dual learning is of great interest because it can accommodate any unidirectional architecture, e.g. a transformer (Vaswani et al., 2017), and provide a performance boost. Moreover, dual learning can be used in semi-supervised learning, which is highly desirable since deep neural networks are generally thirst for labeled data.

Despite the empirical success of dual learning, theoretical understanding is very limited. In this paper, we conduct both theoretical analyses and empirical studies to answer the following questions:

- *Why and when does dual learning improve a mapping function?*
- *Can we further improve the performance of a mapping function?*

1.1. Our Contributions

Our contributions are twofold: a theoretical study of dual learning and the framework of multi-step dual learning, which subsumes dual learning as a special case. Without loss of generality, we take machine translation as an example for the study and algorithm presentation.

Dual learning theory. We take a novel statistical approach to model the problem. Suppose there are two vanilla translators between two language spaces, one forward and the other backward. Based on our Theorem 1, dual learning outperforms both vanilla translators under natural assumptions. Empirical studies show that an improvement is observed even if the reconstruction is far from perfect.

Multi-step dual learning. We propose the multi-step dual learning framework by extending dual learning. This framework uses dual learning as the basic building block and leverages a third, a fourth, or more languages to help boost the translator qualities between the original two languages. We prove that under mild conditions, this framework outperforms dual learning (Theorem 2). Our experiments on MultiUN dataset show a significant improvement (1.45 BLEU points, see Table 6) from dual learning.

1.2. Related Work

Dual learning was first proposed by He et al. (2016) in the context of machine translation, where the two dual translators are updated in a reinforcement learning manner with the reconstructed distortion as the feedback signal. A similar approach proposed by Cheng et al. (2016) has the same high-level idea but their implementation is very different. Since then dual learning architectures have been proposed for other applications including image processing (Kim et al., 2017; Zhu et al., 2017), sentiment analysis (Xia et al., 2017a), image segmentation (Luo et al., 2017), etc.

Built upon the dual learning framework, Xia et al. (2017b) and Wang et al. (2018) considered the joint distribution constraint, which says the joint distribution of samples over two domains is invariant when computing from either domain. We relax this constraint for simplicity of analysis. Xia et al. (2018) proposed model-level dual learning, which shares components between the primary direction and the dual direction. Dual learning was also leveraged for unsupervised learning (Lample et al., 2018; Artetxe et al., 2018).

Despite the vast number of works related to dual learning, theoretical analysis is very limited. (Xia et al., 2017a,b)[15,16] conducted simple analysis of generalization ability in the supervised setting, which are different from our semi-supervised setting. Galanti et al. (2018) claim that dual learning does not circumvent the alignment problem, where a sentence is translated wrong by the forward translator but translated back to it by the backward translator. We show that the alignment problem occurs with a small probability under dual learning, and this probability can be further reduced by our multi-step dual learning. Furthermore, their hypothesis that the translator should not be too complex is not verified in the context of machine translation.

Another line of research is back-translation (Sennrich et al., 2016a; Poncelas et al., 2018; Edunov et al., 2018), which leverages a backward translator to generate parallel data. There are two major differences between dual learning and back-translation: (1) Dual learning aims at improving the performances of all candidate models, while back-translation focuses on using a reversed model (fixed) to boost the primal model; (2) Back-translation generate synthesis offline, which are fed into the primal model; dual learning generates data iteratively, by which the quality of synthesis data is better due to the optimization of each model. Furthermore, our multi-step dual learning utilizes three or more language domains to enhance translators.

2. Preliminaries

Let $S_1, \ldots, S_k$ be k language spaces, composed of sentences in each language. For any S_i, we denote the distribution of sentences in S_i by $\mu^{(i)}$ and let $X^{(i)}$ be the associated random variable, i.e. $\Pr(X^{(i)} = x) = \mu^{(i)}(x)$. As there are multiple sentences for the same meaning in each language, we assume there are a finite number of clusters in each language space, where the sentences within each cluster have the same meaning.

Table 1: Notations

k	number of language spaces
S_i	i-th language space
$\mu^{(i)}$	distribution of sentences in S_i
$X^{(i)}$	the random variable that follows $\mu^{(i)}$
$x^{(i)}$	one sample (sentence) in S_i
T^*_{ij}	the oracle translator from S_i to S_j
T_{ij}	vanilla translator that translates from S_i to S_j
p_{ij}	accuracy of the T_{ij}
T^d_{ij}	translator that translates from S_i to S_j trained using dual learning
p^d_{ij}	accuracy of $T^d ij$
T^m_{ij}	translator that translates from S_i to S_j trained using multi-step dual learning
p^m_{ij}	accuracy of T^m_{ij}

Let T^*_{ij} denote the oracle translator that maps a cluster or a sentence in the cluster from S_i to the correct cluster of S_j. See Figure 1 left for an example. There are clusters C_1, C_2, etc. in S_1, and an oracle translator T^*_{12} maps any cluster (e.g., C_1) or an element in the cluster (e.g., $x \in C_1$) to the correct cluster ($T^*_{12}(C_1)$), which is a set of sentences. Let $C(x^{(i)})$ denote the cluster to which

$x^{(i)}$ belongs. Let T_{ij} denote a vanilla translator that translates from a sentence in S_i to one in S_j. The desired mapping is $T_{ij}(x^{(i)}) \in T^*_{ij}(x^{(i)})$, where T^*_{ij} is the oracle translator. When a sentence $x^{(i)}$ is randomly sampled from S_i according to $\mu^{(i)}$, it is possible that this sentence is translated incorrectly. We use p_{ij}, the accuracy of the translator, to describe the probability of translating a sentence correctly when this sentence is randomly sampled from S_i according to $\mu^{(i)}$. Formally,

$$p_{ij} = \Pr_{X^{(i)} \sim \mu^{(i)}}(T_{ij}(X^{(i)}) \in T^*_{ij}(X^{(i)})) = \sum_{x^{(i)} \in S_i, T_{ij}(x^{(i)}) \in T^*_{ij}(x^{(i)})} \mu^{(i)}(x^{(i)}).$$

We sometimes omit the subscript $X^{(i)} \sim \mu^{(i)}$ for simplicity. It is also easy to see that $\Pr(T_{ij}(x^{(i)}) \notin T^*_{ij}(x^{(i)})) = 1 - p_{ij}$.

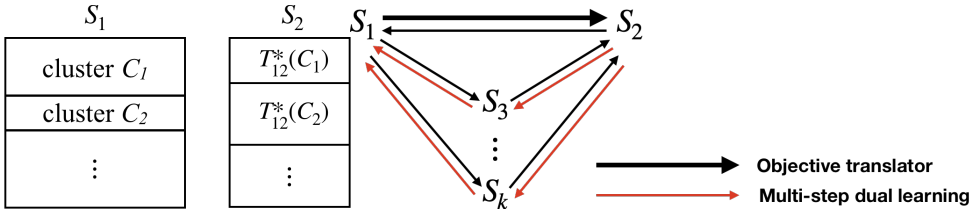

Figure 1: Left: illustration of two language spaces and an oracle translator. Right: the proposed multi-step dual learning framework.

In order to characterize reconstruction accuracy, we let $\tilde{X}^{(j)}$ denote the random variable that follows the distribution $T_{ij}(X^{(i)})$ where $X^{(i)} \sim \mu^{(i)}$. We define $\tilde{\mu}^j(x) = \Pr(\tilde{X}^{(j)} = x)$, and further define

$$\tilde{p}_{ji} = \Pr_{\tilde{X}^{(j)} \sim \tilde{\mu}^{(j)}}(T_{ji}(\tilde{X}^{(j)}) \in T^*_{ji}(\tilde{X}^{(j)})) = \sum_{\tilde{X}^{(j)} \in S_j, T_{ji}(\tilde{X}^{(j)}) \in T^*_{ji}(\tilde{X}^{(j)})} \tilde{\mu}^{(j)}(\tilde{X}^{(j)}).$$

The difference between $\tilde{p}_{ji}$ and p_{ji} lies in the distributions of samples in space S_j. See Table 1 for the list of notations.

3. Theoretical Study of Dual Learning

Algorithm 1 Semi-supervised Dual Learning Framework

Require: Parallel data D_{12} for languages S_1 and S_2, additional monolingual data D_1 for S_1 and D_2 for S_2.

1: Train vanilla translators for directions $S_1 \to S_2$ and $S_2 \to S_1$ respectively using parallel data. The obtained vanilla translators are T_{12} and T_{21}.

2: Continue training both translators so that the translation loss on the parallel data D_{12} and reconstruction loss on monolingual datasets D_1 and D_2 are minimized. The obtained translators are T^d_{12} and T^d_{21}.

We consider a semi-supervised learning task where some parallel sentences are available to train the vanilla translators in both directions, and a large amount of monolingual sentences in addition to

the parallel data are available for dual learning. The structures of the translators, the way to define losses, and the optimization algorithms are decided by the designer (see Algorithm 1). W.l.o.g. we focus on two language spaces S_1 and S_2. Recall that T_{ij} denotes the vanilla translator from S_i to S_j. For each sentence $x^{(1)} \in S_1$, we will focus on the 4-tuple $(x^{(1)}, \mu^{(1)}(x^{(1)}), T_{12}(x^{(1)}), T_{21}(T_{12}(x^{(1)})))$. Define random variables Y_{12} and Y_{21}, which indicate whether T_{12} and T_{21} produce correct translations in each 4-tuple. Formally, we have

$$Y_{12} = \begin{cases} 1, \text{ if } T_{12}(x^{(1)}) \in T_{12}^*(x^{(1)}) \\ 0, \text{ otherwise.} \end{cases}$$

and

$$Y_{21} = \begin{cases} 1, \text{ if } T_{21}(T_{12}(x^{(1)})) \in T_{21}^*(T_{12}(x^{(1)})) \\ 0, \text{ otherwise.} \end{cases}.$$

Then by definition, we have

$$p_{12} = \Pr(Y_{12} = 1) \tag{1}$$
$$\tilde{p}_{21} = \Pr(Y_{21} = 1) \tag{2}$$

In order to analyze dual learning, we consider the joint distribution of Y_{12} and Y_{21}. We use λ to model the dependence of Y_{12} and Y_{21}. Formally,

$$\Pr(Y_{12} = 1, Y_{21} = 1) = p_{12}\tilde{p}_{21} + \lambda \tag{3}$$

It's easy to see that

$$\begin{aligned} \Pr(Y_{12} = 1, Y_{21} = 0) &= p_{12}(1 - \tilde{p}_{21}) - \lambda \\ \Pr(Y_{12} = 0, Y_{21} = 1) &= (1 - p_{12})\tilde{p}_{21} - \lambda \\ \Pr(Y_{12} = 0, Y_{21} = 0) &= (1 - p_{12})(1 - \tilde{p}_{21}) + \lambda \end{aligned}$$

using (1) and (2).

Because all these probabilities are nonnegative, we have

$$-\min\{p_{12}\tilde{p}_{21}, (1 - p_{12})(1 - \tilde{p}_{21})\} \le \lambda \le \min\{p_{12}, \tilde{p}_{21}\} \tag{4}$$

The probability of the alignment issue, which means for some $x^{(1)} \in S_1$, $T_{21}(T_{12}(x^{(1)})) \in C(x^{(1)})$ and $Y_{12} = Y_{21} = 0$, is part of $\Pr(Y_{12} = 0, Y_{21} = 0)$. We use δ to model how likely this issue occurs. Formally,

$$p_{\text{align}} = \delta((1 - p_{12})(1 - \tilde{p}_{21}) + \lambda), \tag{5}$$

where $0 \le \delta \le 1$. For translators T_{12}^d and T_{21}^d obtained from dual learning, we construct 4-tuples in the same way, i.e., $(x^{(1)}, \mu^{(1)}(x^{(1)}), T_{12}^d(x^{(1)}), T_{21}^d(T_{12}^d(x^{(1)})))$ and define random variables Y_{12}^d and Y_{21}^d similarly. Let

$$Y_{12}^d = \begin{cases} 1, \text{ if } T_{12}^d(x^{(1)}) \in T_{12}^*(x^{(1)}) \\ 0, \text{ otherwise.} \end{cases}$$

and

$$Y_{21}^d = \begin{cases} 1, \text{ if } T_{21}^d(T_{12}^d(x^{(1)})) \in T_{21}^*(T_{12}^d(x^{(1)})) \\ 0, \text{ otherwise.} \end{cases}.$$

We are interested in the accuracy of T_{12}^d, $p_{12}^d = \Pr(Y_{12}^d = 1)$. To bridge the vanilla translators and dual translators, we make an assumption, which says if a sample in S_1 is successfully reconstructed by vanilla translators, it is also successfully reconstructed by dual translators, formally stated as follows.

Assumption 1 *For any $x \in S_1$, if $T_{21}(T_{12}(x)) \in C(x)$, then $T_{21}^d(T_{12}^d(x)) \in C(x)$ holds.*

For simplicity, we denote this case as Case 1 and the remaining cases as Case 2. Formally, for any $x \in S_1$,
Case 1: $T_{21}(T_{12}(x)) \in C(x)$;
Case 2: $T_{21}(T_{12}(x)) \notin C(x)$.

For any given $x \in S_1$ which falls in Case 2, we define

$$\begin{aligned}
\alpha &= \Pr(T_{12}^d(x) \in T_{12}^*(x), T_{21}^d(T_{12}^d(x)) \in C(x) | \text{Case 2}) \\
\beta &= \Pr(T_{12}^d(x) \notin T_{12}^*(x), T_{21}^d(T_{12}^d(x)) \in C(x) | \text{Case 2}) \\
\gamma &= \Pr(T_{21}^d(T_{12}^d(x)) \notin C(x) | \text{Case 2}),
\end{aligned} \tag{6}$$

where "Case 2" denotes the condition $T_{21}(T_{12}(x)) \notin C(x)$. Here α can be viewed as the probability of correcting the wrong translations by dual learning, β the probability of the occurrence of the alignment problem under Case 2, and γ the probability of nonzero reconstruction error. γ models the imperfectness of dual learning, which should be zero in the ideal case. It is easy to see $\alpha + \beta + \gamma = 1$. The following theorem give a theoretical study of why dual learning outperforms the baseline translator by the following theorem.

Theorem 1 *Under Assumption 1, for any language spaces S_1 and S_2, the accuracy of dual learning outcome T_{12}^d is $p_{12}^d = (1-\alpha)(p_{12}\tilde{p}_{21} + \lambda) + \alpha\delta(p_{12} + \tilde{p}_{21} - p_{12}\tilde{p}_{21} - \lambda) + \alpha(1-\delta)$, where λ, δ, α are defined in (3),(5) and (6).*

Proof Consider a random sample x and the translation from $x \in S_1$ to S_2. Before dual learning, the accuracy is p_{12}. We analyze the two cases defined earlier in Section 3.
Case 1. $T_{21}(T_{12}(x)) \in C(x)$. Case 1 consists of two subcases:
Case 1.1: $T_{12}(x) \in T_{12}^*(x)$;
Case 1.2: $T_{12}(x) \notin T_{12}^*(x)$.

Although Case 1.2 is not desired, dual learning does not detect it. From (3) and (5), the probabilities of the Case 1.1 and Case 1.2 are

$$\Pr(\text{Case 1.1}) = \Pr(Y_{12} = Y_{21} = 1) = p_{12}\tilde{p}_{21} + \lambda,$$

$$\Pr(\text{Case 1.2}) = p_{\text{align}} = \delta((1-p_{12})(1-\tilde{p}_{21}) + \lambda).$$

Case 2. $T_{21}(T_{12}(x)) \notin C(x)$. Dual learning will train the translators so that this case is minimized. The probability of this case is simply the complement of Case 1:

$$\begin{aligned}
\Pr(\text{Case 2}) &= 1 - (p_{12}\tilde{p}_{21} + \lambda) - \delta((1-p_{12})(1-\tilde{p}_{21}) + \lambda) \\
&= 1 - \delta - (1+\delta)(p_{12}\tilde{p}_{21} + \lambda) + \delta(p_{12} + \tilde{p}_{21}).
\end{aligned}$$

After dual learning, Case 2 is redistributed to Case 1.1 and Case 1.2, with probabilities α and β respectively. So we have

$$\begin{aligned}&\Pr(T_{12}^d(x) \in T_{12}^*(x), T_{21}^d(T_{12}^d(x)) \in C(x))\\ =&p_{12}\tilde{p}_{21} + \lambda + \alpha \Pr(\text{Case 2})\\ =&(1-\alpha)(p_{12}\tilde{p}_{21}+\lambda) + \alpha\delta(p_{12}+\tilde{p}_{21}-p_{12}\tilde{p}_{21}-\lambda) + \alpha(1-\delta),\end{aligned}$$

which is the accuracy of dual learning. ■

Relation to the vanilla translators. Observing that $1-\alpha \geq 0$, $p_{12}+\tilde{p}_{21}-p_{12}\tilde{p}_{21}-\lambda \geq 0$ (due to (4)) and $1-\delta \geq 0$, the accuracy is dual learning improvement is positively correlated to the vanilla translators of both directions. The larger the p_{12} or $\tilde{p}_{21}$ is, the higher accuracy of T_{12}^d dual learning can achieve.

The role of α and δ. We have $p_{12}^d = \alpha(1-\delta-(1+\delta)(p_{12}\tilde{p}_{21}+\lambda)+\delta(p_{12}+\tilde{p}_{21}))+p_{12}\tilde{p}_{21}+\lambda$ by reorganization. So a larger α is desirable, which is intuitively true. Also, p_{12}^d can be reorganized as $-\alpha\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)+\alpha+(1-\alpha)(p_{12}\tilde{p}_{21}+\lambda)$, which means a small δ is desirable.

A hypothesis on α and β. We consider the case where the probabilities of redistribution to α case and β case are proportional to $\Pr(\text{Case 1.1})$ and $\Pr(\text{Case 1.2})$. Formally,

$$\frac{\alpha}{\beta} = \frac{\Pr(T_{12}(x) \in T_{12}^*(x), T_{21}(T_{12}(x)) \in C(x))}{\Pr(T_{12}(x) \notin T_{12}^*(x), T_{21}(T_{12}(x)) \in C(x))} = \frac{p_{12}\tilde{p}_{21}+\lambda}{\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)}.$$

Then we have

$$\begin{aligned}p_{12}^d &= \frac{(p_{12}\tilde{p}_{21}+\lambda)(1-\gamma(1-p_{12}\tilde{p}_{21}-\lambda-\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)))}{p_{12}\tilde{p}_{21}+\lambda+\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)}\\ &= \frac{(p_{12}\tilde{p}_{21}+\lambda)(1-\Gamma)}{p_{12}\tilde{p}_{21}+\lambda+\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)},\end{aligned} \tag{7}$$

where $\Gamma = \gamma(1-p_{12}\tilde{p}_{21}-\lambda-\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda))$. To compare p_{12}^d with the accuracy of the original translator, we compute the difference

$$\begin{aligned}p_{12}^d - p_{12} =&p_{12}(\frac{(\tilde{p}_{21}+\lambda/p_{12})(1-\Gamma)}{p_{12}\tilde{p}_{21}+\lambda++\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)}-1)\\ =&p_{12}(\frac{\tilde{p}_{21}+\lambda/p_{12}}{p_{12}\tilde{p}_{21}+\lambda+\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)}-1-\Gamma\Delta)\\ =&p_{12}(\frac{((1+\delta)\tilde{p}_{21}-\delta)(1-p_{12})+\lambda(1/p_{12}-1+\delta)}{p_{12}\tilde{p}_{21}+\lambda+\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)}-\Gamma\Delta)\end{aligned}$$

where $\Delta = \frac{\tilde{p}_{21}+\lambda/p_{12}}{p_{12}\tilde{p}_{21}+\lambda+\delta((1-p_{12})(1-\tilde{p}_{21})+\lambda)}$. Ideally, we have $\gamma = 0$, which means $\Gamma = 0$. If $\tilde{p}_{21} > \frac{\delta}{1+\delta}$, the outcome of dual learning is better than the vanilla translator. This condition is very mild because δ is small in general. The expression with the Γ factor is negative, which is consistent with the intuition that γ should be minimized.

4. Extension: Multi-Step Dual Learning

In Theorem 1, we found that both p_{ij} and $\tilde{p}_{ji}$ play positive roles in improving p_{ij}^d under mild assumptions. A natural question is whether this probability could be further enhanced by exploiting multiple language domains. Therefore, we propose the frameworks of multi-step dual learning, leveraging multiple languages and significantly extend the standard dual learning.

The proposed frameworks are illustrated in Figure 1 right. Let S_1 and S_2 denote the source language space and the target language space respectively. To use these frameworks, we first train the following translators: $S_1 \leftrightarrow S_2$, $S_1 \leftrightarrow S_k$ and $S_2 \leftrightarrow S_k$ where $k \geq 3$. Then, we require a sentence from S_2 to be very similar to $S_2 \rightarrow S_k \rightarrow S_1 \rightarrow S_2$ (or equivalently, a sentence from S_1 to be very similar to $S_1 \rightarrow S_2 \rightarrow S_k \rightarrow S_1$); In this way, we build another constraint, where the translation $S_1 \rightarrow S_2$ could leverage the information pivoted by the domain S_k. In practice, to use multi-step dual learning to enhance the $S_1 \rightarrow S_2$ model, we need to minimize $\sum_{x^{(2)} \in S_2} D(T_{12}(T_{k1}(T_{2k}(x^{(2)}))), x^{(2)})$, where $D(\cdot, \cdot)$ measures the differences of two inputs. Similar update could also be applied to $S_2 \rightarrow S_1$ translation and leverage more language domains. If no auxiliary domain is provided, multi-step dual learning will degenerate to the standard dual learning. We design sampling based algorithms for this framework. Let θ_{ij} denote the parameters of translator T_{ij}. The algorithm is formally shown as Algorithm 2.

Algorithm 2 Multi-Step Dual Learning Framework

Require: Samples from spaces $S_1 \dots S_k$, initial translators T_{12}, T_{21} and T_{1i}, T_{i1} $\forall i = 3, \dots, K$; learning rates η;

1: Train each of T_{12}, T_{21} and T_{1i}, T_{i1} $\forall i = 3, \dots, k$ by dual learning;
2: Randomly sample a k from $\{3, 4, \cdots, K\}$; randomly sample one $x^{(1)} \in S_1$ and one $x^{(2)} \in S_2$;
3: Generate $\tilde{x}^{(2)}$ by $T_{k2}(T_{1k}(x^{(1)}))$ and generate $\tilde{x}^{(1)}$ by $T_{k1}(T_{2k}(x^{(2)}))$;
4: Update the parameters of T_{12} and T_{21}, denoted as θ_{12} and θ_{21}, as follows:

$$\theta_{12} \leftarrow \theta_{12} + \eta \nabla_{\theta_{12}} \ln \Pr(x^{(2)} | \tilde{x}^{(1)}; \theta_{12});$$
$$\theta_{21} \leftarrow \theta_{21} + \eta \nabla_{\theta_{21}} \ln \Pr(x^{(1)} | \tilde{x}^{(2)}; \theta_{21}); \tag{8}$$

5: Repeat Step 3 to Step 5 until convergence;

4.1. Theoretical Analysis

We provide a theoretical analysis of this framework. For simplicity, we focus on the triangle structure that contains only S_1, S_2 and S_3. For each sentence $x \in S_1$, we will focus on the 5-tuple $(x, \mu^{(1)}(x), T_{12}^d(x), T_{23}^d(T_{12}^d(x)), T_{31}^d(T_{23}^d(T_{12}^d(x))))$. Define random variables Z_{12}, Z_{23} and Z_{31}, which indicate whether T_{12}^d, T_{23}^d and T_{31}^d produce correct translations in each 5-tuple. Formally, we have

$$Z_{12} = \begin{cases} 1, \text{ if } T_{12}^d(x) \in T_{12}^*(x) \\ 0, \text{ otherwise.} \end{cases} \tag{9}$$

$$Z_{23} = \begin{cases} 1, \text{ if } T_{23}^d(T_{12}^d(x)) \in T_{23}^*(T_{12}^d(x)) \\ 0, \text{ otherwise.} \end{cases} \tag{10}$$

and

$$Z_{31} = \begin{cases} 1, \text{ if } T_{31}^d(T_{23}^d(T_{12}^d(x))) \in T_{31}^*(T_{23}^d(T_{12}^d(x))) \\ 0, \text{ otherwise.} \end{cases} \tag{11}$$

We define $q_{12} = \Pr(Z_{12} = 1)$, $q_{23} = \Pr(Z_{23} = 1)$, and $q_{31} = \Pr(Z_{31} = 1)$. For simplicity, we assume the same dependence on any two of Z_{12}, Z_{23}, Z_{31}. Formally,

$$\begin{aligned} \Pr(Z_{12} = Z_{23} = 1) &= q_{12}q_{23} + \lambda_1 \\ \Pr(Z_{23} = Z_{31} = 1) &= q_{23}q_{31} + \lambda_1 \\ \Pr(Z_{12} = Z_{31} = 1) &= q_{12}q_{31} + \lambda_1 \end{aligned} \tag{12}$$

We let

$$\Pr(Z_{12} = Z_{23} = Z_{31} = 1) = q_{12}q_{23}q_{31} + \lambda_2, \tag{13}$$

where λ_2 captures the dependence of all three variables. Then the joint distribution of Z_{12}, Z_{23}, and Z_{31} can be written as expressions of λ_1 and λ_2. Similar to the analysis for dual learning, we use δ to describe how likely $T_{31}^d(T_{23}^d(T_{12}^d(x))) \in C(x)$ occurs when one or more of the three translators give incorrect translations. Formally, for any $x \in S_1$ and Z_{12}, Z_{23}, Z_{31} s.t. $Z_{12} + Z_{23} + Z_{31} \leq 2$, $\Pr(T_{31}^d(T_{23}^d(T_{12}^d(x))) \in C(x) | Z_{12}, Z_{23}, Z_{31}) = \delta \Pr(Z_{12}, Z_{23}, Z_{31})$. Now we are interested in the accuracy of T_{12}^m obtained from multi-step dual learning, which is $q_{12}^m = \Pr(Z_{12} = 1)$. To bridge q_{ij} and q_{ij}^m, we make the following assumption, which says if a sentence is successfully reconstructed in a cycle by translators obtained from dual learning, it will also be successfully reconstructed in a cycle by translators obtained from multi-step dual learning.

Assumption 2 *For any $x \in S_1$, if $T_{31}^d(T_{23}^d(T_{12}^d(x))) \in C(x)$, then $T_{31}^m(T_{23}^m(T_{12}^m(x))) \in C(x)$.*

For simplicity, we denote the invariant case as Case 1 and the remaining cases as Case 2. Formally, for any $x \in S_1$,

We focus on the following two cases:
Case 1: $T_{31}^d(T_{23}^d(T_{12}^d(x))) \in C(x)$;
Case 2: $T_{31}^d(T_{23}^d(T_{12}^d(x))) \notin C(x)$.

Multi-step dual learning will train the translators so that Case 2 is minimized. To quantify this effect, we define the following probabilities:

$$\begin{aligned} \alpha' &= \Pr(T_{12}^m(x) \in T_{12}^*(x), T_{31}^m(T_{23}^m(T_{12}^m(x))) \in C(x) | \text{Case 2}) \\ \beta' &= \Pr(T_{12}^m(x) \notin T_{12}^*(x), T_{31}^m(T_{23}^m(T_{12}^m(x))) \in C(x) | \text{Case 2}) \\ \gamma' &= \Pr(T_{31}^m(T_{23}^m(T_{12}^m(x))) \notin C(x) | \text{Case 2}), \end{aligned} \tag{14}$$

where Case 2 denotes the condition $T_{31}^d(T_{23}^d(T_{12}^d(x))) \notin C(x)$. α', β', γ' can be viewed as the probability of correcting the wrong translations by multi-step dual learning, the probability of the occurrence of the alignment problem under Case 2, and the probability of nonzero reconstruction error. γ' models the imperfectness of dual learning. And we have $\alpha' + \beta' + \gamma' = 1$.

We have the following theorem about the triangle structure. The more general case can be viewed as adding one path a time so Theorem 2 can be applied.

Theorem 2 *Given languages spaces S_1, S_2, and S_3, where the objective is to train a translator that maps from S_1 to S_2 and under Assumption 2, the accuracy of multi-step dual learning outcome q_{12}^m is*

$$\begin{aligned}&(1-\alpha')(q_{12}q_{23}q_{31}+\delta q_{12}(1-q_{23})(1-q_{31})+(1+\delta)\lambda_2)\\&+\alpha'(1-\delta(1-q_{12})(1-q_{23}q_{31}-\lambda_1+\lambda_2).\end{aligned}\tag{15}$$

The proof technique of Theorem 2 is similar to that of Theorem 1, though more complicated. The full proof is deferred to Section 8.

Roles of λ_1 and λ_2. It is easy to see q_{12}^m improves when λ_1 increases. For the impact of λ_2, we reorganize the term with λ_2 and have $(1+\delta)\lambda_2-\alpha'(1+\delta)\lambda_2-\alpha'\delta(1-q_{12})\lambda_2$. α' is generally not close to 1, so a larger λ_2 helps in most cases. In the rest of our analysis we assume $\lambda_1=\lambda_2=0$ for simplification.

A similar hypothesis. Similar to the analysis of dual learning, we consider the condition where $\frac{\alpha'}{\beta'}=\frac{\Pr(T_{12}^d(x)\in T_{12}^*(x),T_{31}^d(T_{23}^d(T_{12}^d(x)))\in C(x))}{\Pr(T_{12}^d(x)\notin T_{12}^*(x),T_{31}^d(T_{23}^d(T_{12}^d(x)))\in C(x))}$ and define $\Gamma'=\gamma'\Pr(T_{31}^d(T_{23}^d(T_{12}^d(x)))\notin C(x))$. Then the accuracy simplifies to

$$q_{12}^m=\frac{\alpha'(1-\Gamma')}{\alpha'+\beta'}=\frac{1-\Gamma'}{1+\frac{\delta(1-q_{12})(1-q_{23}q_{31}}{q_{12}q_{23}q_{31}+\delta q_{12}(1-q_{23})(1-q_{31})}}=\frac{1-\Gamma'}{1+M\frac{1-q_{12}}{q_{12}}},$$

where $M=\frac{\delta(1-q_{23}q_{31})}{q_{23}q_{31}+\delta(1-q_{23})(1-q_{31})}$. We observe that When $\gamma'=0$ (and therefore $\Gamma'=0$) and $M=1$, it simplifies to q_{12}, which is the accuracy of dual learning and that q_{12}^m increases as M decreases. To characterize the condition when $q_{12}^m>q_{12}$, we let $M<1$. We have $\frac{\delta(1-q_{23}q_{31})}{q_{23}q_{31}+\delta(1-q_{23})(1-q_{31})}<1$, which leads to $q_{23}(\frac{2\delta+1}{2\delta}q_{31}-1)+q_{31}(\frac{2\delta+1}{2\delta}q_{23}-1)>0$. When $q_{23},q_{31}>\frac{\delta}{\delta+0.5}$, which is also mild, multi-step dual learning outperforms dual learning.

5. Experiments of Dual Learning

Since previous works (He et al., 2016; Xia et al., 2017a,b; Wang et al., 2018; Xia et al., 2018) have demonstrated the strength of dual learning, we aim at providing some theoretical insights. We choose WMT14 English↔Germen translation[1] and MultiUN (Eisele and Chen, 2010) English↔French translation[2] to verify our theoretical analysis for dual learning. For ease of reference, denote English, French, and German as En, Fr, and De respectively.

Datasets. Following the common practice in NMT, for the En↔De tasks, we preprocess the data in the same way as that used in Ott et al. (2018), including tokenizing the words and applying BPE (Sennrich et al., 2016b) with $32k$ merge operations. Eventually, we obtain $4.5M$ training sentence pairs. We concatenate newstest2012 and newstest2013 as the validation set ($6K$ sentence pairs) and choose newstest2014 as the test set ($3K$ sentence pairs). For the MultiUN En↔Fr translation, following Ren et al. (2018), we sample $2M$/$6K$/$3K$ sentence pairs as the training/validation/test sets. All sentences from MultiUN datasets are split into wordpiece following (Johnson et al., 2016). To leverage dual learning, for WMT'14 En↔De translation, we choose $40M$ monolingual English sentences and $40M$

1. Data available at http://www.statmt.org/wmt14/translation-task.html.
2. http://opus.nlpl.eu/MultiUN.php

monolingual German sentences from newscrawl[3]. For MultiUN En↔Fr translation, we randomly sample $1M$ English and $1M$ French sentences as the monolingual data to construct the duality loss.

Architecture. We use the Transformer model (Vaswani et al., 2017) for each translation task. For WMT En↔De translation, we choose the *transformer_big* configuration, in which the word embedding dimension, hidden dimension and number of heads in multi-head attention are 1024, 4096 and 16 respectively. For MultiUN En↔Fr translation, we choose the *transformer_base* configuration, in which the aforementioned three numbers are 512, 2048 and 8 respectively. Both *transformer_big* and *transformer_base* represent networks with six layers.

Optimization. We choose Adam (Kingma and Ba, 2015) with `inverse_sqrt` learning rate scheduler (Vaswani et al., 2017) to optimize the models. All experiments are conducted on eight GPUs. For WMT En↔De tasks, following (Ott et al., 2018), we set the learning rate as 5×10^{-4} and the batch size as 4096 tokens per GPU. The gradient is accumulated 16 times before update. For MultiUN tasks, the learning rate is 2×10^{-4} and the batch size is 7168 tokens per GPU. All the models are trained until convergence.

Evaluation. We use beam search with beam width 4 to generate candidates. The evaluation metric is BLEU score (Papineni et al., 2002), which is a geometric mean of n-gram precisions ($n = 1, 2, 3, 4$). We choose the script `multi-bleu.perl`[4] to calculate BLEU scores. A large BLEU score indicates a better translation quality.

Translation qualities. The BLEU scores of all translation tasks are summarized in Table 2, in which the second row and third row represent the results of the standard Transformer and dual learning. We can see that after applying dual learning, the performances of all tasks are boosted. Specifically, on En→De and De→En translation tasks, we can boost the baseline from 29.79 to 32.18 (2.39 points improvement), and from 34.15 to 38.06 (3.91 points improvement). On the other task, dual learning can achieve 0.65 and 0.86 point improvement, which demonstrates its effectiveness. We found that on MultiUN, we do not achieve as much improvement as WMT. The reason is that the MultiUN dataset is a collection of translated documents from the United Nations, which are usually of formal and simple patterns that are easy to learn. As a result, introducing more data might not increase the BLEU so much.

Table 2: BLEU scores of WMT2014 En↔De and MultiUN En↔Fr translations tasks.

	En→De	De→En	En→Fr	Fr→En
Vanilla	29.79	34.15	50.26	50.56
Dual	32.18	38.06	50.91	51.42

We are aware that back translation (Sennrich et al., 2016a) is another baseline of leveraging monolingual data. We apply this technique to WMT En→De and De→En. We obtain 30.43 and 37.17 BLEU scores respectively, which are not as good as dual learning. We leave the study of back translation as future work.

Translator accuracy. We interpret the results in terms of accuracy, i.e., the p_{ij} and p^d_{ij} in Table 1. A sentence is regarded to be correctly translated if the corresponding BLEU score is larger than a given threshold BLEU score. We choose threshold BLEU score to be 10 and 20.

3. http://data.statmt.org/news-crawl/
4. https://github.com/moses-smt/mosesdecoder/blob/master/scripts/generic/multi-bleu.perl

Table 3: Accuracy of Translators Using Different Threshold BLEU.

Threshold BLEU	En↔De				En↔Fr			
	p_{12}	p_{21}	p_{12}^d	p_{21}^d	p_{12}	p_{21}	p_{12}^d	p_{21}^d
10	0.65	0.73	0.70	0.77	0.82	0.80	0.82	0.81
20	0.54	0.65	0.60	0.70	0.77	0.74	0.78	0.75

The accuracy of each translator is shown in Table 3. Let S_1 and S_2 denote English and German respectively for the En↔De task (English and French respectively for the En↔Fr task). Values are percentages of translations that are above the threshold. Qualitatively, we observe that dual learning outcomes are better than standard transformers. More interesting observations lie in the following quantitative analysis on En↔De task.

Evaluation of Assumption 1 and empirical α, β, γ. For a given test dataset, we define the empirical estimate of α, β, and γ as follows.

$$\hat{\alpha} = \frac{\#\text{ of } x | T_{12}^d(x) \in T_{12}^*(x), T_{21}^d(T_{12}^d(x)) \in C(x)}{\#\text{ of } x | T_{21}(T_{12}(x)) \notin C(x)}$$

$$\hat{\beta} = \frac{\#\text{ of } x | T_{12}^d(x) \notin T_{12}^*(x), T_{21}^d(T_{12}^d(x)) \in C(x)}{\#\text{ of } x | T_{21}(T_{12}(x)) \notin C(x)}$$

$$\hat{\gamma} = \frac{\#\text{ of } x | T_{21}^d(T_{12}^d(x)) \notin C(x)}{\#\text{ of } x | T_{21}(T_{12}(x)) \notin C(x)}$$

To evaluate Assumption 1, we define $\eta = \frac{\#\text{ of } x | T_{21}^d(T_{12}^d(x)) \in C(x)}{\#\text{ of } x | T_{21}(T_{12}(x)) \in C(x)}$.

Ideally $\eta = 1$. The empirical estimates using the En↔De test data under threshold BLEU scores 10 and 20 are shown in Table 4. In each cell, the values on the left are for En→De direction and the values on the right are for De→En direction. We observe that η values are close to 1, which means Assumption 1 is reasonable. The high $\hat{\gamma}$ values indicate that the reconstruction loss is still high after dual learning. Therefore, we believe there exist approaches to improve dual learning and how to further reduce the reconstruction loss is a promising direction.

Table 4: Estimated Parameters Using Test Data.

Threshold BLEU	10	20
η	96.8%|94.3%	96.4%|93.3%
α	0.30|0.31	0.27|0.32
β	0.28|0.23	0.32|0.24
γ	0.42|0.45	0.41|0.44

6. Experiments of Multi-Step Dual Learning

To verify the effectiveness of multi-step dual learning, we work on the translation between English (En), French (Fr) and Spanish (Es). Again, we choose to use the MultiUN dataset to train the

translation models since any two of the aforementioned three languages have bilingual sentence pairs. We study two different settings, where for each language pair, we are provided with $2M$ or $0.2M$ bilingual sentence pairs. For both settings, we choose $1M$ monolingual sentences for each language. We use *transformer_base* for all experiments in this section, where the model is a six-block network, with word embedding size, hidden dimension size and number of heads 512, 2048 and 6. The training process is the same as that in Section 5.

Table 5: Experimental Results on MultiUN (2M bilingual data)

	En→Fr	Fr→En	En→Es	Es→En	Es→Fr	Fr→Es
Vanilla	50.26	50.56	55.15	55.23	47.75	48.13
Dual	50.91	51.42	55.51	55.77	48.23	48.52
Multi-step	51.28	51.89	55.97	56.17	48.62	48.87

The experimental results of using $2M$ bilingual data and $1M$ monolingual data are shown in Table 5. We can see that on average, dual learning can boost the six baselines (i.e., standard transformer) by 0.55 point. Although dual learning can achieve very high scores on MultiUN translation tasks, our proposed multi-step dual learning can still improve it by 0.41 point on average.

Table 6: Experimental Results on MultiUN (0.2M bilingual data)

	En→Fr	Fr→En	En→Es	Es→En	Es→Fr	Fr→Es
Vanilla	43.12	43.26	49.28	47.80	41.47	41.21
Dual	45.54	45.44	51.07	50.31	42.81	42.57
Multi-step	47.23	46.72	52.56	51.65	43.52	44.97

The results of using 0.2M bilingual data plus 1M monolingual data is shown in Table 6. We have the following observations:

(1) Since there are fewer bilingual sentences, the baselines of the six translation tasks are not as good as those in Table 5.

(2) For this setting, dual learning can improve the BLEU scores by 1.93 points on average, which is consistent with the discovery in He et al. (2016) that dual learning can obtain more improvements when the number of bilingual sentences is small.

(3) When multi-step dual learning is added to the conventional dual learning, we can achieve extra 1.45 improvements on average, which demonstrates the effectiveness of multi-step dual learning. We also observe that multi-step dual learning can bring more improvement when the number of labeled data is limited.

7. Conclusions

We provide the first theoretical study of dual learning and characterize conditions when dual learning outperforms vanilla translators. We also propose an algorithmic extension of dual learning, the multi-step dual learning framework, which is provably better than dual learning under mild conditions. Our dual learning experiments demonstrate the efficacy of dual learning w.r.t. accuracy and provide

insights into the potential power of dual learning. Our experiments on multi-step dual learning framework show further improvement from dual learning.

8. Proof of Theorem 2

We focus on the mapping from $x \in S_1$ to S_2 and consider the following two cases:
Case 1: $T^d_{31}(T^d_{23}(T^d_{12}(x))) \in C(x)$;
Case 2: $T^d_{31}(T^d_{23}(T^d_{12}(x))) \notin C(x)$.
Case 1: Recall the definitions of Z_{12}, Z_{23} and Z_{31} in (9)-(11). There are two subcases in Case 1:
Case 1.1: $T^d_{12}(x) \in T^*_{12}(x)$ $(Z_{12} = 1)$,
Case 1.2: $T^d_{12}(x) \notin T^*_{12}(x)$ $(Z_{12} = 0)$.
Case 1.1. We have $\Pr(\text{Case 1.1}) = \Pr(Z_{12} = Z_{23} = Z_{31} = 1) + \delta \Pr(Z_{12} = 1, Z_{23} = Z_{31} = 0)$, where $Z_{12} = Z_{23} = Z_{31} = 1$ means all translators give correct translations and $Z_{12} = 1, Z_{23} = Z_{31} = 0$ means T^d_{12} translates correctly but T^d_{23} and T^d_{31} both give incorrect translations. Only a small fraction happen to give correct translations at S_1, captured by δ. By (13), $\Pr(Z_{12} = Z_{23} = Z_{31} = 1) = q_{12}q_{23}q_{31} + \lambda_2$. Now we compute $\Pr(Z_{12} = 1, Z_{23} = Z_{31} = 0)$.

$$
\begin{aligned}
&\Pr(Z_{12} = 1, Z_{23} = Z_{31} = 0) \\
=&\Pr(Z_{12} = 1, Z_{23} = 0) - \Pr(Z_{12} = 1, Z_{23} = 0, Z_{31} = 1) \\
=&\Pr(Z_{12} = 1, Z_{23} = 0) - \Pr(Z_{12} = 1, Z_{31} = 1) + \Pr(Z_{12} = Z_{23} = Z_{31} = 1) \\
=&q_{12} - (q_{12}q_{23} + \lambda_1) - (q_{12}q_{31} + \lambda_1) + q_{12}q_{23}q_{31} + \lambda_2 \\
=&q_{12}(1 - q_{23})(1 - q_{31}) + \lambda_2
\end{aligned}
$$

where the third equality is obtained by (12) and (13). So we have

$$\Pr(\text{Case 1.1}) = q_{12}q_{23}q_{31} + \lambda_2 + \delta(q_{12}(1 - q_{23})(1 - q_{31}) + \lambda_2) \tag{16}$$

Case 1.2. This case is possible only if $Z_{12} = 0$, $Z_{23} + Z_{31} \leq 1$, which means T^d_{12} gives incorrect translations; T^d_{23} and T^d_{31} do not give correct translations simultaneously. We write the probability of this case as $\Pr(\text{Case 1.2}) = \delta(\Pr(Z_{12} = Z_{23} = Z_{31} = 0) + \Pr(Z_{12} = Z_{23} = 0, Z_{31} = 1) + \Pr(Z_{12} = 0, Z_{23} = 1, Z_{31} = 0))$.

To compute it, we have

$$
\begin{aligned}
&\Pr(Z_{12} = 0, Z_{23} = 1, Z_{31} = 0) \\
=&\Pr(Z_{12} = 0, Z_{23} = 1) - \Pr(Z_{12} = 0, Z_{23} = 1, Z_{31} = 1) \\
=&\Pr(Z_{23} = 1) - \Pr(Z_{12} = Z_{23} = 1) - \Pr(Z_{23} = Z_{31} = 1) + \Pr(Z_{12} = 1, Z_{23} = 1, Z_{31} = 1) \\
=&q_{23} - (q_{12}q_{23} + \lambda_1) - (q_{23}q_{31} + \lambda_1) + q_{12}q_{23}q_{31} + \lambda_2 \\
=&(1 - q_{12})q_{23}(1 - q_{31}) - 2\lambda_1 + \lambda_2
\end{aligned}
$$

Similarly we can compute $\Pr(Z_{12} = Z_{23} = 0, Z_{31} = 1) = (1 - q_{12})(1 - q_{23})q_{31} - 2\lambda_1 + \lambda_2$ and $\Pr(Z_{12} = Z_{23} = Z_{31} = 0) = (1 - q_{12})(1 - q_{23})(1 - q_{31}) + 3\lambda_1 - \lambda_2$. Then the probability of Case 1.2 is

$$\Pr(\text{Case 1.2}) = \delta(1 - q_{12})(1 - q_{23}q_{31} - \lambda_1 + \lambda_2) \tag{17}$$

Case 2. The probability of Case 2 is simply the complement of Case 1.

$$\Pr(\text{Case 2}) = 1 - \Pr(\text{Case 1.1}) - \Pr(\text{Case 1.2})$$

Then the accuracy of this triple learning is

$$q_{12}^m = \Pr(\text{Case 1.1}) + \alpha' \Pr(\text{Case 2}) = (1-\alpha')\Pr(\text{Case 1.1}) + \alpha'(1 - \Pr(\text{Case 1.2})) \quad (18)$$

where α' is defined in (14). (15) is obtained by substitute (16) and (17) into (18).

References

Mikel Artetxe, Gorka Labaka, Eneko Agirre, and Kyunghyun Cho. Unsupervised neural machine translation. In *International Conference on Learning Representations (ICLR 2018)*, 2018.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Semi-supervised learning for neural machine translation. In *ACL*, volume 1, pages 1965–1974, 2016.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500, 2018.

Andreas Eisele and Yu Chen. Multiun: A multilingual corpus from united nation documents. In *Proceedings of the Seventh conference on International Language Resources and Evaluation*, pages 2868–2872, 5 2010.

Tomer Galanti, Lior Wolf, and Sagie Benaim. The role of minimal complexity functions in unsupervised learning of semantic mappings. In *6th International Conference on Learning Representations*, 2018.

Di He, Yingce Xia, Tao Qin, Liwei Wang, Nenghai Yu, Tieyan Liu, and Wei-Ying Ma. Dual learning for machine translation. In *Advances in Neural Information Processing Systems*, pages 820–828, 2016.

Melvin Johnson, Mike Schuster, Quoc V Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, et al. Google's multilingual neural machine translation system: enabling zero-shot translation. *arXiv preprint arXiv:1611.04558*, 2016.

Taeksoo Kim, Moonsu Cha, Hyunsoo Kim, Jung Kwon Lee, and Jiwon Kim. Learning to discover cross-domain relations with generative adversarial networks. In *International Conference on Machine Learning*, pages 1857–1865, 2017.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *ICLR*, 2015.

Guillaume Lample, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. Unsupervised machine translation using monolingual corpora only. In *International Conference on Learning Representations (ICLR 2018)*, 2018.

Ping Luo, Guangrun Wang, Liang Lin, and Xiaogang Wang. Deep dual learning for semantic image segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA*, pages 21–26, 2017.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. Scaling neural machine translation. *arXiv preprint arXiv:1806.00187*, 2018.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics, 2002.

Alberto Poncelas, Dimitar Shterionov, Andy Way, Gideon Maillette de Buy Wenniger, and Peyman Passban. Investigating backtranslation in neural machine translation. In *Proceedings of the 21st Annual Conference of the European Association for Machine Translation: 28-30 May 2018, Universitat d'Alacant, Alacant, Spain*, pages 249–258. European Association for Machine Translation, 2018.

Shuo Ren, Wenhu Chen, Shujie Liu, Mu Li, Ming Zhou, and Shuai Ma. Triangular architecture for rare language translation. *ACL*, 2018.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany, August 2016a. Association for Computational Linguistics. doi: 10.18653/v1/P16-1009. URL https://www.aclweb.org/anthology/P16-1009.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *ACL*, 2016b.

Duyu Tang, Nan Duan, Tao Qin, Zhao Yan, and Ming Zhou. Question answering and question generation as dual tasks. *arXiv preprint arXiv:1706.02027*, 2017.

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

Yijun Wang, Yingce Xia, Li Zhao, Jiang Bian, Tao Qin, Guiquan Liu, and T Liu. Dual transfer learning for neural machine translation with marginal distribution regularization. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

Yingce Xia, Jiang Bian, Tao Qin, Nenghai Yu, and Tie-Yan Liu. Dual inference for machine learning. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 3112–3118, 2017a.

Yingce Xia, Tao Qin, Wei Chen, Jiang Bian, Nenghai Yu, and Tie-Yan Liu. Dual supervised learning. In *International Conference on Machine Learning*, pages 3789–3798, 2017b.

Yingce Xia, Xu Tan, Fei Tian, Tao Qin, Nenghai Yu, and Tie-Yan Liu. Model-level dual learning. In *International Conference on Machine Learning*, pages 3789–3798, 2018.

Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *IEEE International Conference on Computer Vision*, 2017.