



# Exploiting Inactive Examples for Natural Language Generation With Data Rejuvenation

Wenxiang Jiao , Xing Wang, Shilin He, Zhaopeng Tu, Irwin King , *Fellow, IEEE*,  
and Michael R. Lyu, *Fellow, IEEE*

**Abstract**—Recent years have witnessed the success of natural language generation (NLG) accomplished by deep neural networks, which require a large amount of training data for optimization. With the constant increase of data scale, the complex patterns and potential noises make training NLG models difficult. In order to fully utilize large-scale training data, we explore inactive examples in the training data and propose to rejuvenate the inactive examples for improving the performance of NLG models. Specifically, we define inactive examples as those sentence pairs that contribute less to the performance of NLG models, and show that their existence is independent of model variants but mainly determined by the data distribution. We further introduce *data rejuvenation* to improve the training of NLG models by re-labeling the inactive examples. The rejuvenated examples and active examples are combined to train a final NLG model. We evaluate our approach by experiments on machine translation (MT) and text summarization (TS) tasks, and achieve significant improvements of performance. Extensive analyses reveal that inactive examples are more difficult to learn than active ones and rejuvenation can reduce the learning difficulty, which stabilizes and accelerates the training process of NLG models and results in models with better generalization capability.

**Index Terms**—Natural language generation, inactive example, data rejuvenation, machine translation, text summarization.

## I. INTRODUCTION

NATURAL language generation (NLG) [1]–[11] is a data-hungry approach, which requires a large amount of data to train a well-performing NLG model [12]. However, the complex patterns and potential noises in the large-scale data make training NLG models difficult. To relieve this problem, several approaches have been proposed to better exploit the training data. For example, noisy data [13], [14] is an unavoidable issue in large-scale datasets and can be cleaned with trusted data to

improve the training of NLG models. Curriculum learning [15]–[17] argues that the learning process of NLG models can mimic the way human learns such that human learns easy data first and gradually grasps the difficult data. Data diversification [18] attempts to fully exploit the training data by generating diversified synthetic data solely from the training data.

In this paper, we explore an interesting alternative which is to reactivate the *inactive examples* in the training data for NLG models. By definition, inactive examples are the training examples that only marginally contribute to or even inversely harm the performance of NLG models. Concretely, we use sentence-level output probability [19] assigned by a trained NLG model to measure the activeness level of training examples, and regard the examples with the least probabilities as inactive examples. We further propose *data rejuvenation* to rejuvenate the inactive examples to improve the performance of NLG models. Specifically, we train an NLG model on the active examples as the rejuvenation model to re-label the inactive examples, resulting in the rejuvenated examples (Section III-B). The final NLG model is trained on the combination of the active examples and rejuvenated examples. We demonstrate our findings and approach on two representative NLG tasks, i.e., machine translation (MT) and text summarization (TS). For simplicity, we focus on high-resource MT tasks for ablation studies and then extend to medium- and low-resource MT tasks and the TS tasks.

For high-resource MT tasks, which include WMT14 English-German and English-French tasks, we empirically show that removing 10% most inactive examples can marginally improve translation performance (Section IV-B). In addition, we observe a high overlapping ratio (e.g., around 80%) of the most inactive and active examples across random seeds, model capacity, and model architectures. These results provide empirical support for our hypothesis of the existence of inactive examples in large-scale datasets, which is invariant to specific NLG models and mainly depends on the data distribution itself. Besides, experimental results show that the proposed data rejuvenation approach consistently and significantly improves performance on state-of-the-art NLG models (e.g., LSTM [20], TRANSFORMER [3], and DYNAMICCONV [21]) on the two MT tasks (Section IV-D), and is also complementary to existing data manipulation methods (e.g., curriculum learning [15], data diversification [18] and data denoising [13]). Further, we conduct extensive analyses to better understand the inactive examples (Section IV-E1) and the proposed data rejuvenation approach

Manuscript received October 10, 2021; revised January 22, 2022; accepted February 1, 2022. Date of publication February 24, 2022; date of current version March 4, 2022. This work was supported in part by the National Key Research and Development Program of China under Grant 2018AAA0100204 and in part by the Research Grants Council of the Hong Kong Special Administrative Region, China under Grant CUHK 14210717 of the General Research Fund. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Sakriani Sakti. (*Corresponding author: Wenxiang Jiao.*)

Wenxiang Jiao, Irwin King, and Michael R. Lyu are with the Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China (e-mail: joelwxjiao@tencent.com; king@cse.cuhk.edu.hk; lyu@cse.cuhk.edu.hk).

Xing Wang and Zhaopeng Tu are with Tencent AI Lab, Shenzhen, Guangdong 518057, China (e-mail: brightxwang@tencent.com; zptu@tencent.com).

Shilin He is with Microsoft Research Lab-Asia, Beijing 100080, China (e-mail: shilin.he@microsoft.com).

Digital Object Identifier 10.1109/TASLP.2022.3153269

(Section IV-E2). Quantitative analyses reveal that the inactive examples are more difficult to learn than active ones, and data rejuvenation can reduce the learning difficulty. Human translations from target to source sentences also tend to be inactive examples. The rejuvenated examples stabilize and accelerate the training process of NLG models, resulting in final models with better generalization capability.

Further, we evaluate our data rejuvenation approach on the medium/low-resource MT tasks (Section V) including WMT16 Romanian-English and IWSLT14 German-English, and the TS task (Section VI) including the Annotated English Gigaword and XSum [22]. Experimental results suggest that our approach can also achieve non-trivial improvements on these tasks, further demonstrating the effectiveness and universality of our approach. Case studies on the TS task indicate that the reference summarizations of inactive examples either deviate from or miss key points of the input text, which can be fixed by data rejuvenation to a considerable extent.

This work is an extension of our previous results [23], in which (1) we present a more comprehensive description of the adopted models (Section IV-A) and analysis methods (Section IV-E); (2) we conduct more experiments for the original high-resource MT scenario including, the comparison with curriculum learning (Section IV-D2), the evaluation on a larger dataset in a new testing setup (Section IV-D5) and additional analysis results (Section IV-E1); (3) we extend our approach to medium- and low-resource MT tasks to demonstrate its effectiveness under different data scales (Section V); (4) we also extend our approach to another representative NLG task, i.e., the TS tasks, to show the generality of the approach (Section VI). In summary, our contributions of this work are as below:

- We demonstrate the existence of inactive examples, which mainly depends on the data distribution, in large-scale datasets for NLG tasks including MT and TS.
- We propose a general framework<sup>1</sup> to rejuvenate the inactive examples to improve the training of NLG models, and achieve significant improvements on state-of-the-art models (e.g., TRANSFORMER and DYNAMICCONV) on benchmark datasets without modifying the model architecture and training strategies.
- We conduct extensive analyses to understand the properties of inactive examples and the proposed data rejuvenation approach.
- We successfully demonstrate the data rejuvenation approach on various NLG scenarios, including the MT tasks with high/medium/low resources and the TS tasks.

The rest of this paper is arranged as follows: In Section II, we review previous studies that are related to this work. In Section III, we introduce our data rejuvenation approach. In Section IV, we validate our findings and approach on the high-resource MT task, including ablation studies, main experiments, and analysis results. We further demonstrate the inactive examples and rejuvenation approach on the medium- and low-resource MT tasks in Section V, and the TS tasks in Section VI.

<sup>1</sup>Source code: [Online]. Available: <https://github.com/wxjjiao/Data-Rejuvenation>

TABLE I  
EFFECT OF DIFFERENT REJUVENATION STRATEGIES ON WMT14 EN  $\Rightarrow$  DE TRANSLATION TASK

| Rejuvenation        | BLEU        | $\Delta$ |
|---------------------|-------------|----------|
| n/a                 | 27.5        | –        |
| Forward Translation | <b>28.3</b> | +0.8     |
| Back-Translation    | 27.5        | +0.0     |
| Both                | 27.8        | +0.3     |

TABLE II  
COMPARISON BETWEEN DATA REJUVENATION ON IDENTIFIED INACTIVE EXAMPLES AND FORWARD TRANSLATION ON RANDOMLY SAMPLING EXAMPLES ON WMT14 EN  $\Rightarrow$  DE TRANSLATION TASK

| Training Data          | BLEU | $\Delta$ |
|------------------------|------|----------|
| Raw Data               | 27.5 | –        |
| 10% Inactive Examples  | 27.8 | +0.3     |
| + Rejuvenated Examples | 28.3 | +0.8     |
| 10% Random Examples    | 27.4 | -0.1     |
| + Rejuvenated Examples | 27.3 | -0.2     |

At last, we draw the conclusion in Section VII and present some possible directions for the future work.

## II. RELATED WORK

### A. Manipulating Training Examples

Our work belongs to the category of data manipulation for NLG tasks, which aims at fully utilizing the original training data without incorporating extra data. Representative studies include the data denoising approach [13] and the data diversification approach [18]. Specifically, data denoising relies on trusted data to filter the noisy data in the training data to let the NLG models learn from clean data. Data diversification relies on pseudo labeling by the models trained with varied random seeds and different directions (i.e., forward translation and back-translation) to diversify the original training data. Our data rejuvenation approach is complementary to these approaches such that combining them together leads to additional improvements of performance on the high-resource MT task (see Table IV).

### B. Distinguishing Training Examples

Our work is also closely related to existing studies on distinguishing training examples in machine learning. On one hand, we can emphasize a certain kind of training examples by re-weighting during training. For example, easy examples are preferred in self-paced learning [24], hard examples are exploited in hard example mining [25], and examples with high variance are emphasized in active learning [26]. On the other hand, we can distinguish the order of feeding training examples. For example, curriculum learning schedule training examples by their difficulty and has been successfully applied to the training of NLG models [15], [16], [27]–[29]. Our data rejuvenation approach tries to re-activate inactive examples, where there is no need to change the model architectures and the training strategies.

TABLE III

EVALUATION OF TRANSLATION PERFORMANCE ACROSS MODEL ARCHITECTURES AND HIGH-RESOURCE LANGUAGE PAIRS. “↑ / ⤴”: INDICATE STATISTICALLY SIGNIFICANT IMPROVEMENT OVER THE CORRESPONDING BASELINE  $p < 0.05/0.01$  RESPECTIVELY

| System                      | Architecture        | WMT14 En⇒De       |      | WMT14 En⇒Fr       |      |
|-----------------------------|---------------------|-------------------|------|-------------------|------|
|                             |                     | BLEU              | △    | BLEU              | △    |
| <i>Existing NLG Systems</i> |                     |                   |      |                   |      |
| Vaswani et al. [3]          | TRANSFORMER-BASE    | 27.3              | –    | 38.1              | –    |
|                             | TRANSFORMER-BIG     | 28.4              | –    | 41.0              | –    |
| Ott et al. [42]             | SCALE TRANSFORMER   | 29.3              | –    | 43.2              | –    |
| Wu et al. [21]              | DYNAMIC CONV        | 29.7              | –    | 43.2              | –    |
| <i>Our NLG Systems</i>      |                     |                   |      |                   |      |
| <i>This work</i>            | LSTM                | 26.5              | –    | 40.6              | –    |
|                             | + Data Rejuvenation | 27.0 <sup>↑</sup> | +0.5 | 41.1 <sup>↑</sup> | +0.5 |
|                             | TRANSFORMER-BASE    | 27.5              | –    | 40.2              | –    |
|                             | + Data Rejuvenation | 28.3 <sup>↑</sup> | +0.8 | 41.0 <sup>↑</sup> | +0.8 |
|                             | TRANSFORMER-BIG     | 28.4              | –    | 42.4              | –    |
|                             | + Data Rejuvenation | 29.2 <sup>↑</sup> | +0.8 | 43.0 <sup>↑</sup> | +0.6 |
|                             | + Large Batch       | 29.6              | –    | 43.5              | –    |
|                             | + Data Rejuvenation | 30.3 <sup>↑</sup> | +0.7 | 44.0 <sup>↑</sup> | +0.5 |
|                             | DYNAMIC CONV        | 29.7              | –    | 43.3              | –    |
|                             | + Data Rejuvenation | 30.2 <sup>↑</sup> | +0.5 | 43.9 <sup>↑</sup> | +0.6 |

TABLE IV

COMPARISON WITH OTHER DATA MANIPULATION APPROACHES. RESULTS ARE REPORTED ON WMT14 EN ⇒ DE TEST SET

| Model                     | BLEU | △    |
|---------------------------|------|------|
| TRANSFORMER-BASE          | 27.5 | –    |
| + Data Rejuvenation       | 28.3 | +0.8 |
| + Curriculum Learning     | 27.4 | -0.1 |
| + Data Rejuvenation       | 27.9 | +0.4 |
| + Data Diversification-BT | 26.9 | -0.6 |
| + Data Rejuvenation       | 27.9 | +0.4 |
| + Data Diversification-FT | 28.1 | +0.6 |
| + Data Rejuvenation       | 28.5 | +1.0 |
| + Data Denoising          | 28.1 | +0.6 |
| + Data Rejuvenation       | 28.6 | +1.1 |

### C. Data Redundancy in Computer Vision

Our work is partially inspired by existing studies on data redundancy in computer vision. For example, Birodkar *et al.* [30] demonstrate the data redundancy issue in large-scale image recognition datasets, e.g., CIFAR-10 [31] and ImageNet [32] such that selecting a subset of training examples (with 10% removed) can perform on par with the full training examples. Vodrahalli *et al.* [33] also find that a small subset of MNIST [34] is sufficient for training a well-performing model. In our work, we empirically confirm these findings on the large-scale NLP datasets. More importantly, we propose to rejuvenate the inactive examples to further improve the model performance.

## III. OUR APPROACH: DATA REJUVENATION

In this section, we introduce the framework of our *data rejuvenation* approach. As illustrated in Fig. 1, our approach consists of three steps:

- Train an *identification model* on the original training examples to recognize inactive examples.

- Train a *rejuvenation model* on the active examples to rejuvenate the inactive examples.
- Combine the rejuvenated examples with the active examples to train the final NLG model.

There are many possible ways to implement the general idea of data rejuvenation. The aim of this paper is not to explore this whole space but simply to show that one fairly straightforward implementation works well and that data rejuvenation helps.

### A. Identification Model

We leverage the output probabilities of NLG models, which implicitly learn the statistics of training examples, to implement the identification model. Here, we focus on general sequence-to-sequence NLG tasks that generate a target sentence based on a source sentence. The training objective of the NLG model is to maximize the log-likelihood of the training data  $\{\mathbf{x}^n, \mathbf{y}^n\}_{n=1}^N$ :

$$L(\theta) = \sum_{n=1}^N \log P(\mathbf{y}^n | \mathbf{x}^n). \quad (1)$$

For each sentence pair  $(\mathbf{x}, \mathbf{y})$ , the trained NLG model assigns a sentence-level probability  $P(\mathbf{y} | \mathbf{x})$  to it, representing the confidence of the model to generate the target sentence  $\mathbf{y}$  from the source sentence  $\mathbf{x}$  [19], [35]. Intuitively, a training example with a low sentence-level probability is less likely to provide useful information for improving model performance, and thus can be regarded as an inactive example.

As a result, we adopt the sentence-level probability  $P(\mathbf{y} | \mathbf{x})$  as the metric to measure the activeness level of each training example:

$$I(\mathbf{y} | \mathbf{x}) = \left[ \prod_{t=1}^T p(y_t | \mathbf{x}, \mathbf{y}_{<t}) \right]^{1/T}, \quad (2)$$

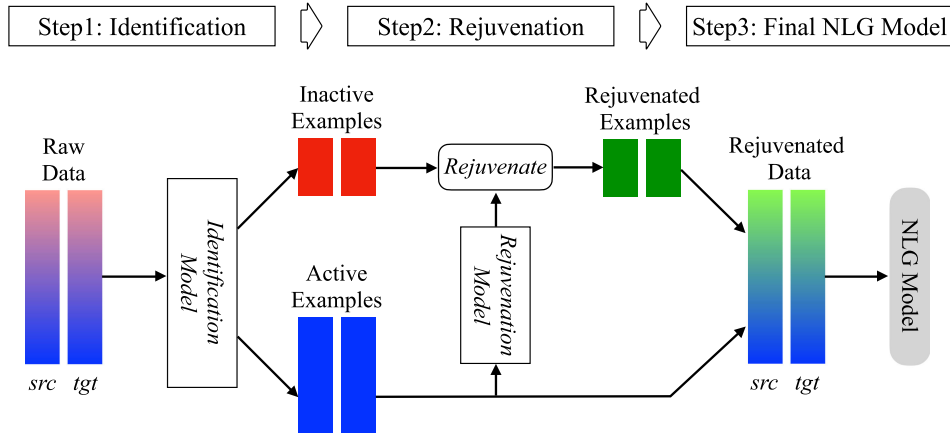


Fig. 1. Framework of data rejuvenation. The inactive examples from the original training data are identified by the *identification model*, then rejuvenated by the *rejuvenation model*. The rejuvenated examples along with the active examples are used together to train the NLG model.

where  $T$  denotes the number of target words in the training example. As seen, we also normalize  $I(y|x)$  by  $T$  to avoid length bias. Finally, in experiments, we train an NLG model on the original training data as the identification model and adopt it to score each training example. A certain percent of training examples with the lowest sentence-level probabilities are treated as inactive examples.

### B. Rejuvenation Model

For the rejuvenation model, we explore the pseudo labeling techniques due to their success in data augmentation for NLG tasks. Concretely, we re-generate either the source sentence of inactive examples by back-translation [36] or the target sentence by forward-translation [37]. Accordingly, the rejuvenation model will be a backward NLG model or a forward NLG model, respectively. We train the rejuvenation model on the active examples to exclude the potentially negative effects of inactive examples. The rejuvenation model re-activates each inactive example by translating the source (for forward-translation) or target (for back-translation) sentence to produce a synthetic parallel example. Benefiting from the knowledge distillation based on active examples, the rejuvenated examples contain simpler and more correct patterns than the original inactive examples [38], making them easier to be learned by NLG models. As shown in Fig. 6, the quality of rejuvenated examples is generally good, with simpler word choices (i.e., lower frequency rank), better alignments (i.e., higher coverage), and easier translation mappings (i.e., lower uncertainty). Therefore, by default, we do not set rules to filter out the potentially low-quality translated data. In our preliminary experiments, we also try removing rejuvenated examples (about 39% of all rejuvenated examples) with coverage values lower than 0.9. This results in a slight performance drop of our data rejuvenation approach. We speculate that the filtering process removes too many rejuvenated examples, which may contain important knowledge. In addition, we also need carefully designed filtering criteria, without which we recommend to use all the rejuvenated data.

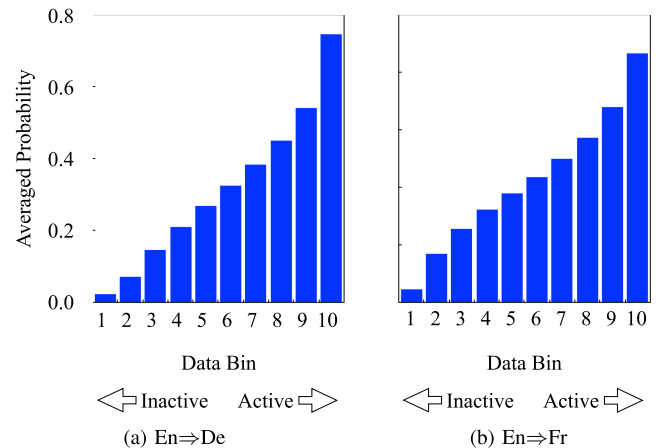


Fig. 2. Probability diagram on WMT14 (a)  $En \Rightarrow De$  and (b)  $En \Rightarrow Fr$  training data. Training examples in smaller bins (e.g., 1, 2) are regarded as inactive examples due to their lower probabilities.

## IV. HIGH-RESOURCE TRANSLATION TASK

First of all, we validated our findings and approach on high-resource translation tasks. This section includes the results of ablation studies, main experiments, and analysis.

### A. Experimental Setup

1) *Data*: For high-resource translation tasks, we chose WMT14 English  $\Rightarrow$  German<sup>2</sup> ( $En \Rightarrow De$ ) and English  $\Rightarrow$  French<sup>3</sup> ( $En \Rightarrow Fr$ ) datasets for experiments. Concretely, the WMT14  $En \Rightarrow De$  dataset contains about 4.5 M sentence pairs and the  $En \Rightarrow Fr$  dataset contains 35.5 M, respectively. For both language pairs, we applied byte-pair encoding (BPE) [39] with 32 K merge operations.

2) *Architecture*: We demonstrated the proposed approach on a variety of representative NLG architectures:

<sup>2</sup>[Online]. Available: <https://nlp.stanford.edu/projects/nmt/>

<sup>3</sup>[Online]. Available: <http://www.statmt.org/wmt14/>



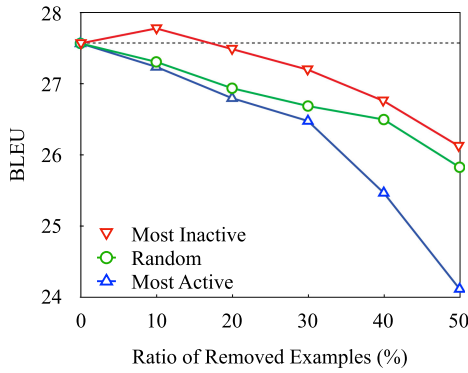


Fig. 3. Translation performance of the NLG model trained on the training data with the most inactive examples removed. For comparison, results of the most active examples and randomly sampled examples are also presented.

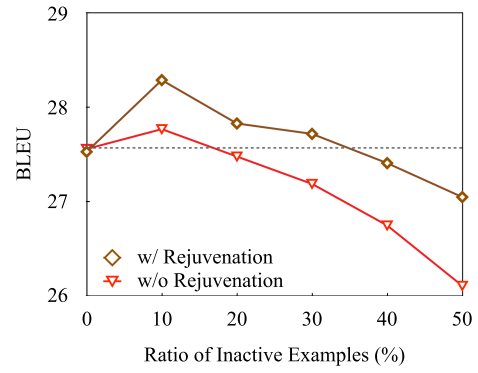


Fig. 5. Effect of the ratio of examples labeled as inactive examples on WMT14 En → De translation task. We used forward-translation as the rejuvenation strategy and trained the final NLG model on the combination of rejuvenated examples and active examples from scratch.

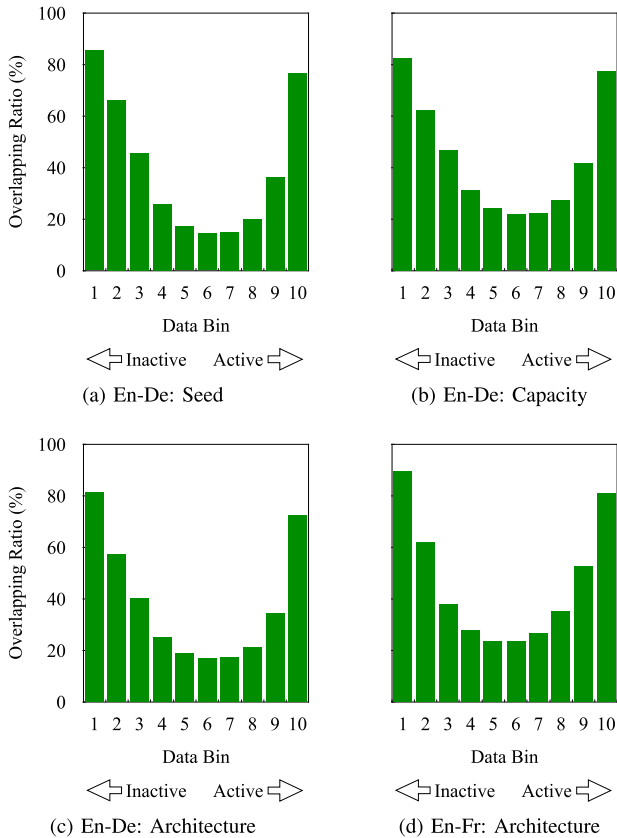


Fig. 4. Ratio of examples that are shared by different model variants: random seed (a), model capacity (b), model architecture on WMT14 En → De (c) and En → Fr (d) datasets. A high overlapping ratio for most inactive examples (i.e., 1<sup>st</sup> data bin) demonstrates that the identified inactive examples are not model-specific.

- LSTM [20] that is implemented in the TRANSFORMER framework. We follow Domhan *et al.* [20] to implement LSTM by replacing the self-attention (SAN) layers in TRANSFORMER-BASE with LSTM layers. Specifically, we use a bidirectional LSTM for each layer of the encoder, and a unidirectional LSTM for each layer of the decoder. Each

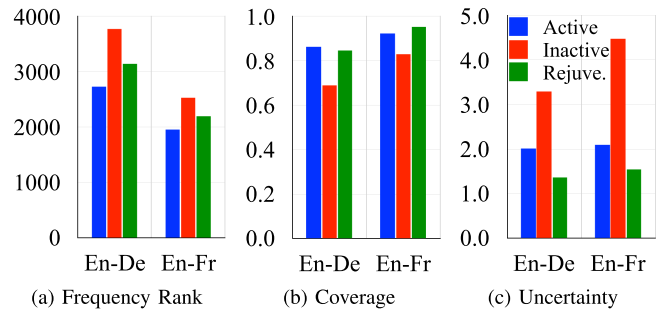


Fig. 6. Linguistic properties of different training examples on WMT14 En → De and En → Fr training data: frequency rank (↑ more difficult), coverage (↓ more difficult), and uncertainty (↑ more difficult).

bidirectional LSTM layer is followed by a fully-connected layer with ReLU as the activation function.

- TRANSFORMER [3] that is implemented with only attention mechanisms for encoder, decoder, and encoder-decoder attention networks. The TRANSFORMER model may be configured with different capacities, for example, TINY, BASE and BIG (Section IV-B).
- DYNAMICCONV [21] that is implemented with lightweight and dynamic convolutions, which can perform competitively to the best reported TRANSFORMER-BIG results. Here the DYNAMICCONV has a similar capacity with TRANSFORMER-BIG.

To implement the above NLG models, we utilized the open-source toolkit Fairseq [40]. The models were trained under the same settings in the original works. Briefly, we trained the LSTM model for 100 K steps with a batch size of 32 K (4096 × 8) tokens. For TRANSFORMER, the BASE model followed the same setting as LSTM while the BIG model was trained with the same batch size but for 300 K steps. The DYNAMICCONV model was trained with a large batch size of 459 K (3584 × 128) tokens for 30 K steps. We selected the model with the lowest perplexity on the validation set as the final model. We evaluated the translation performance of the NLG models by case-sensitive BLEU score [41]. The translations were generated by beam search with

a beam width of 4 and a length penalty of 0.6 for both WMT14 En  $\Rightarrow$  De and En  $\Rightarrow$  Fr translation tasks.

The other parts of this section are organized as below: We first conducted ablation studies on the identification model (Section IV-B) and rejuvenation model (Section IV-C) on the WMT14 En  $\Rightarrow$  De dataset with TRANSFORMER-BASE. Then we reported the translation performance on different model architectures and language pairs, as well as the comparison with previous studies (Section IV-D). Finally, we presented extensive analyses to understand the inactive examples and the proposed data rejuvenation approach (Section IV-E).

### B. Reasonableness of Identification Model

We investigated the reasonableness of the identification model in this section, including the performance contribution of identified inactive examples and the sensitivity to model-specific factors.

1) *Identified Inactive Examples*: As described in Section III-A, we ranked the training examples by their sentence-level output probabilities assigned by a trained NLG model. Following Wang *et al.* [35], we partitioned the training examples into 10 equal bins (i.e., each bin contains 10% of training examples) according to the ranking of their probabilities and reported the averaged probability of each bin in Fig. 2. We regarded the examples in the 1<sup>st</sup> data bin as inactive examples, which have much lower probabilities than the other ones.

2) *Performance Contribution*: In this experiment, we studied if the identified inactive examples satisfy our definition, i.e., those examples that only marginally contribute to or even inversely harm the performance of NLG models. According to this definition, we can remove the inactive examples without harming the performance, as they cannot provide useful information to the NLG models. Therefore, here we removed a certain percentage of examples with the least probabilities (e.g., most inactive examples) from the training data, and evaluated the performance of the NLG model that is trained on the remaining data. We also conducted the same experiments for the active examples and randomly selected examples.

Fig. 3 shows the contribution of these three kinds of training examples to translation performance. Generally, the performance drop grows up with the increased portion of training examples being removed. However, the declining trend of the inactive examples is more gentle than that of the randomly selected examples, and the steepest comes from the active examples. These results meet our expectation for the inactive examples and active examples, demonstrating the reasonableness of the identification model. In addition, we can observe that the translation performance does not degrade when removing 10% of the most inactive examples, which is consistent with the finding of Birodkar *et al.* [30] on the CV datasets.

3) *Model-Specific Factors*: In this experiment, we studied the sensitivity of the identification model to model-specific factors. Since we counted on a pre-trained NLG model for the identification of inactive examples, one question naturally arises: *are the identified inactive examples model-specific?* For example, with different NLG models, we may obtain inactive

examples that come from different parts of the training data. We analyzed this question by constructing NLG model variants for identification with three common factors that can significantly affect the performance, including random seeds, model capacity, and model architectures. For *random seeds*, we set it to “1,” “12,” “123,” “1234,” and “12345” when training the TRANSFORMER-BASE model. For *model capacity*, we configured the TRANSFORMER model with varied numbers of attention heads and hidden sizes to obtain the TINY ( $3 \times 256$ ), BASE ( $6 \times 512$ ), and BIG ( $6 \times 1024$ ) models. As for *model architectures*, we utilized the aforementioned architectures in Section IV-A. We adopted each of the above model variants as the identification model to rank the training examples and split them into data bins. For each data bin, we calculated the ratio of examples that are shared by different model variants (e.g., different random seeds). Generally, a high overlapping ratio denotes that the identified examples are more agreed by different models, which suggests the examples are not model-specific.

We reported the results in Fig. 4. As seen, there is always a high overlapping ratio (over 80%) for the 1<sup>st</sup> data bin, i.e., the most inactive examples, across model variants and language pairs. It suggests that the identified inactive examples are highly consistent, demonstrating that *the inactive examples are invariant to specific models but mainly depend on the data distribution itself*. At the meantime, we noticed another interesting phenomenon that the 10<sup>th</sup> data bin, i.e., the most active examples, also holds a high agreement by model variants. The overlapping ratios of all model variants (i.e., seeds, capacities, and architectures, 9 models in total) on the En  $\Rightarrow$  De dataset are 70.9%, and 62.5% for the most inactive and (most) active examples, respectively. These results show that deep learning methods share a common ability to learn from the training examples, especially on the most inactive and active examples.

### C. Rejuvenation of Inactive Examples

In this section, we evaluated the rejuvenation model from various aspects, including the ratio of training data treated as inactive examples, the effect of different rejuvenation strategies, and comparing inactive examples with randomly selected examples under rejuvenation.

1) *Ratio for Inactive Examples*: In this experiment, we investigated how the rejuvenation model performs when we treat different ratios of training data as inactive examples. As aforementioned, we treated  $R\%$  of examples with the least sentence-level probabilities as the inactive examples. Now, we increased the ratio  $R\%$  from 10% to 50% and showed the performance after rejuvenation. The results are shown in Fig. 5. Obviously, the rejuvenation of the inactive examples consistently outperforms the non-rejuvenated counterpart, demonstrating the necessity of data rejuvenation. As for the rejuvenation model, the BLEU score declines with the increase of  $R$ . It is intuitive as examples with relatively higher probabilities (e.g., beyond the 10% most inactive examples) can provide useful information for NLG models, and rejuvenating them may inversely harm the translation performance. As a result, in the following experiments,

we treat 10% examples with the least probabilities as inactive examples by default.

2) *Effect of Rejuvenation Strategies*: In this experiment, we studied the effect of different rejuvenation strategies, and reported the results in Table I. It is surprising to find that the back-translation strategy does not improve the model performance. The reason could be that the inactive examples are identified by a forward-translation model (Section III-A), indicating that these inactive examples are more difficult for NLG models to generate from the source side to the target side, rather than in the reverse direction. Accordingly, we conjecture that the forward translation strategy may alleviate this problem by constructing a synthetic example, such that each source side is paired with a simpler target side. As expected, combining these two strategies cannot further improve translation performance. Based on the above results, in the following experiments, we use forward translation as the default rejuvenation strategy.

3) *Inactive Examples or Random Examples*: At last, we investigated if the rejuvenation model can be applied to any training examples. Concretely, we attempted to answer the question: *does the improvement indeed come from data rejuvenation, or just from forward translation?* To analyze this question, we randomly selected 10% training examples as the inactive examples and applied data rejuvenation with the forward translation strategy. The results are listed in Table II. Clearly, removing 10% random examples inversely harms the translation performance, and rejuvenating them leads to a further decrease of performance. In contrast, data rejuvenation over the inactive examples identified by us improves performance as expected. These results suggest that rejuvenation is mainly effective for the inactive examples, further demonstrating the importance of identifying the inactive examples.

#### D. Main Results

1) *Comparison With Vanilla Models*: Table III shows the results of the proposed data rejuvenation approach across model architectures and language pairs. The baseline TRANSFORMER models trained by us achieve better results than that reported in previous work [3], particularly on the large-scale En  $\Rightarrow$  Fr dataset. We also trained the TRANSFORMER-BIG model with 459 K tokens per batch (denoted as “+ Large Batch”) as a stronger baseline, since Ott *et al.* [42] showed that models of larger capacity benefit from training with large batches. We tested statistical significance of our approach over the baselines with paired bootstrap resampling [43] using `compare-mt`<sup>4</sup> [44] with 1000 re-samples.

Obviously, the proposed data rejuvenation approach consistently and significantly improves translation performance in all cases, demonstrating the effectiveness and universality of our approach. Most importantly, our approach achieves significant improvements without introducing any additional data, modifying model architectures or customizing training strategies. It makes the approach robustly applicable to most existing NLG systems.

2) *Comparison With Previous Work*: Clearly, our *data rejuvenation* approach belongs to the family of data manipulation. To further emphasize the advantage of our approach, we compare it with three widely-used manipulation strategies, i.e., curriculum learning [15], data diversification [18], and data denoising [13].

First, for curriculum learning, we followed Zhang *et al.* [15] to group training samples into 10 shards, each shard contains samples with similar difficulties. We reused the sentence-level output probability from our identification model as the difficulty criterion, where the higher probability denotes the easier sample. During training, we used the easiest shard (i.e., shard 1) at the beginning and appended harder shards with the easier shards as the next epochs afterwards (e.g., shard 1 to 2 for epoch 2, shard 1 to 3 for epoch 3, etc.). Because, we find that feeding the shards separately will make the training process unstable with very fluctuated losses. After 10 epochs, we trained the NLG models on the whole training data. When incorporating our data rejuvenation approach, we regard the rejuvenated data (i.e., from the original shard 10) as the easiest shard, so that the rejuvenated data can involve in the whole training process.

Second, for data diversification, we adopted both forward-translation (FT) [37] and back-translation (BT) [36] strategies to construct synthetic data from the original training data without introducing any monolingual data. The combination of the original and the synthetic data is used to train the final NLG model. Here, “Data Diversification-FT” is similar to our approach except that it forward-translates all the original training examples while we only forward-translate the identified inactive examples (10% of the training data).

Third, for data denoising, we ranked the training data according to the noise metric by Wang *et al.* [13], which requires a set of trusted examples. Accordingly, we used WMT newstest 2010-2011 as the trusted data, which consists of 5492 examples. The trained NLG model on the raw data was regarded as the noisy model, which was then fine-tuned on the trusted data to obtain the denoised model. For each training example, a noise score is calculated based on the noisy and denoised models, which is then used for instance sampling during training.

Table IV shows the results of these comparative experiments on the WMT14 En  $\Rightarrow$  De test set. All the approaches improve the translation performance individually except for curriculum learning and data diversification with back-translation. For curriculum learning, the reason could be that it is very sensitive to specific curriculum hyper-parameters in practice and the “easy to hard” ordering strategy is not always effective [15]. In contrast, while our approach needs an identification model (and a rejuvenation model which can reuse a strong identification model), it does not require any tuning of the model and training strategy, which is very simple and effective. For data diversification with back-translation, we speculate that it cannot reduce the noises in the target sentences. Nonetheless, our approach performs on par with these manipulation approaches and can obtain further improvement on top of them, indicating that data rejuvenation is complementary to them.

In addition, we compared the inactive examples identified by us and the noisiest examples (also 10% of the training examples) identified by the data denoising approach. The overlapping ratio

<sup>4</sup>[Online]. Available: <https://github.com/neulab/compare-mt>

TABLE V  
NEW TESTING RULE ON WMT19 EN  $\Rightarrow$  DE DATASETS, EVALUATED ON  
NEWSTEST 2019 AND NEWSTEST 2020

| Model                      | newstest2019 |          | newstest2020 |          |
|----------------------------|--------------|----------|--------------|----------|
|                            | BLEU         | $\Delta$ | BLEU         | $\Delta$ |
| TRANSFORMER-BIG            | 41.1         | -        | 33.7         | -        |
| + <i>Data Rejuvenation</i> | 43.0         | +1.9     | 35.5         | +1.8     |

of these two sets of examples is only 32%, indicating that the inactive examples are not necessarily noisy examples. For a deeper understanding of the inactive example, we will conduct more detailed analyses on linguistic properties of the inactive examples in Section IV-E1.

3) *Random Seeds*: Some researchers may doubt if the improvement achieved by our approach comes from lucky random starts. To dispel this doubt, we conducted experiments on the En  $\Rightarrow$  De dataset using the TRANSFORMER-BASE model with three random seeds (i.e., 1, 12, and 123). Our approach consistently outperforms the baseline model in all cases (i.e., 28.3/27.5, 28.2/27.4, and 27.9/27.1), demonstrating the effectiveness of our approach.

4) *Source Language*: Some researchers may have questions about the language pairs used in the experiments that both language pairs have English as the source language, which could determine the rejuvenation strategy. To demonstrate the universality of our approach across language directions, we conducted an experiment on the WMT14 De-En translation task. The TRANSFORMER-BASE model achieved a BLEU score of 31.2, and the data rejuvenation approach improves performance by +0.6 BLEU point.

5) *New Testing Setup*: Starting from WMT2019 [45], the test sets only include naturally occurring text at the source-side to make a more realistic scenario for practical translation usage. We thus evaluated our approach under this new testing setup. Specifically, we trained TRANSFORMER-BIG models on the WMT19 En  $\Rightarrow$  De datasets with 36.8 M sentence pairs and evaluated on newstest2019 and newstest2020. The results are listed in Table V. As seen, our data rejuvenation approach achieves +1.9 and +1.8 improvements of BLEU score on the two test sets, respectively, demonstrating that our approach is even more effective under this new testing setup.

## E. Analysis

1) *Analysis on Inactive Examples*: In this section, we performed an extensive study to understand inactive examples in terms of linguistic properties, inactive examples v.s. human translations and case study. Unless otherwise stated, all experiments were conducted on the En  $\Rightarrow$  De and En  $\Rightarrow$  Fr datasets with TRANSFORMER-BASE.

*Linguistics Properties*: To obtain a deeper understanding of the inactive examples, we first compare them with active examples and rejuvenated examples in terms of *three* linguistics properties, including frequency rank, coverage, and uncertainty.

- *Frequency Rank*: We adopted frequency rank to measure the rarity of words in the target sentences. In the target vocabulary, words are sorted in the descending order of their frequencies in the whole training data, and the frequency rank of a word is its position in the dictionary. Therefore, the higher the frequency rank is, the more rare the word is in the training data. We report the averaged frequency rank of each of the three subsets. The larger frequency rank of inactive examples indicates that they contain more rare words, which makes them more difficult to be learned by NLG models than the active examples.
- *Coverage*: We adopted coverage to measure the ratio of source words being aligned by any target words [46]. Firstly, we trained an alignment model on the training data by *fast-align*<sup>5</sup> [47], and force-aligned the source and target sentences of each subset. Then, we calculated the coverage of each source sentence, and reported the averaged coverage of each subset. The lower coverage of inactive examples indicates that they are not well aligned as the active examples, which also makes them more difficult for NLG models to learn.
- *Uncertainty*: Uncertainty measures the level of multimodality of a parallel corpus [48]. We adopted uncertainty to reflect the number of possible translations at target side for a source sentence. We used the corpus level uncertainty, which measures the complexity of each subset. Corpus level uncertainty is simplified as the sum of entropy of target words conditioned on the aligned source words denoted  $H(y|x = x_t)$ . Therefore, an alignment model is also required. To prevent uncertainty from being dominated by frequent words, we followed Zhou *et al.* [48] to calculate uncertainty by averaging the entropy of target words conditioned on a source word denoted  $\frac{1}{|V_x|} \sum_{x \in V_x} H(y|x)$ . The larger uncertainty of inactive examples indicates that there are more possible translations for each source sentence within. In other words, inactive examples contain more complex patterns, which are more difficult to be learned by NLG models.

Fig. 6 depicts the results. In summary, the linguistic properties consistently suggest that inactive examples are more difficult than those active ones. By rejuvenation, the inactive examples are transformed into much simpler patterns such that NLG models are able to learn from them.

*Inactive Examples v.s. Human Translations*: In Table I, we demonstrated that forward translation performs better than back-translation for rejuvenation. Then, we wondered if examples with formats as that generated by back-translation, i.e., human translations from target to source, are more likely to be inactive examples. For simplicity, we named such examples as source-translated whereas source-natural otherwise. The information of source-translated/natural examples is unavailable for training examples, but fortunately is provided for test sets<sup>6</sup>. For example, the test sets of En  $\Rightarrow$  De and En  $\Rightarrow$  Fr contain 1500 source-translated and 1503 source-natural examples, respectively. We

<sup>5</sup>[Online]. Available: [https://github.com/clab/fast\\_align](https://github.com/clab/fast_align)

<sup>6</sup>[Online]. Available: <https://www.statmt.org/wmt14/test-full.tgz>



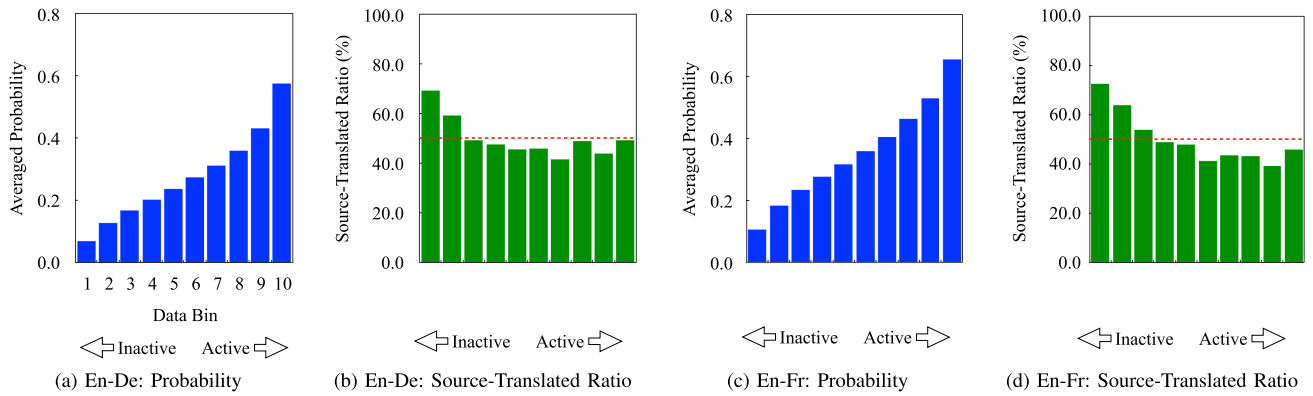


Fig. 7. Probability and ratio of source-translated examples over the data bins of WMT14 En => De and En => Fr test sets.

TABLE VI  
INACTIVE EXAMPLES FROM THE TRAINING SETS OF WMT14 EN => DE AND EN => FR. X, Y AND Y' REPRESENT THE SOURCE SENTENCE, TARGET SENTENCE, AND THE REJUVENATED TARGET SENTENCE, RESPECTIVELY. Y AND Y' ARE ALSO TRANSLATED INTO ENGLISH (=>EN:) BY GOOGLE TRANSLATE FOR REFERENCE. FOR EITHER EXAMPLE, THE UNDERLINED PHRASES CORRESPOND TO THE SAME CONTENT

| Side   | Sentence   |
|--------|--|
| En=>De | X The Second World War <u>finished the destruction of the first</u> .  |
|        | Y Der zweite Weltkrieg <u>tat dann das seine und zerstörte den Rest</u> .<br>=>En: The Second World War <u>then did his and destroyed the rest</u> .                             |
|        | Y' Der Zweite Weltkrieg <u>beendete die Zerstörung des ersten</u> .<br>=>En: The Second World War <u>ended the destruction of the first</u> .                                    |
| En=>Fr | X Anything <u>denied by the latter</u> was effectively confirmed as true .   |
|        | Y Tout ce que <u>démentait cette agence</u> se révélait dans la pratique bien réel .<br>=>En: Everything that <u>this agency denied</u> turned out to be very real in practice . |
|        | Y' Toute chose <u>niée par ce dernier</u> a été effectivement confirmée comme vraie .<br>=>En: Anything <u>denied by the latter</u> has actually been confirmed to be true .     |

split the examples of the two test sets into 10 data bins according to the sentence-level probability of the identification model (i.e., TRANSFORMER-BASE), and then calculate the ratio of source-translated examples in each bin. We reported the results in Fig. 7. As seen, the ratios of source-translated examples in 1<sup>st</sup> and 2<sup>nd</sup> bins significantly exceed that in the whole test sets (i.e., 1500/3003), suggesting that human translations from target to source are more likely to be inactive examples. This study also enhanced our understanding of why the back-translation strategy did not work for rejuvenation.

*Case Study:* By inspecting the inactive examples, we find that the target sentences tend to be paraphrases of the source sentences rather than direct translations. We provide two cases in Table VI. In the first case, the target sentence does not translate “finished the destruction of the first” in the source

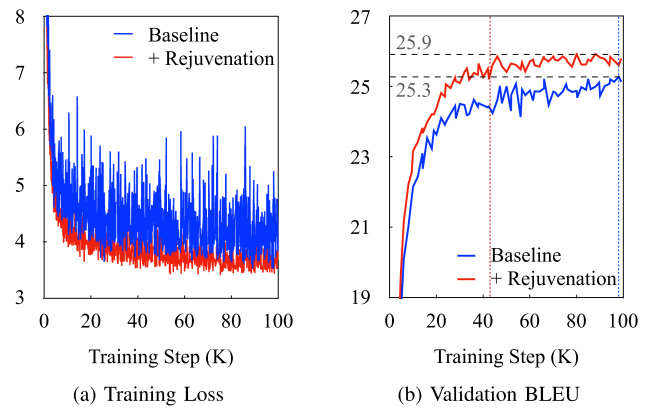


Fig. 8. Learning curves on the En => De dataset.

sentence directly but rephrases it as “tat dann das seine und zerstörte den Rest,” meaning “then did his and destroyed the rest” (that was not destroyed by The First World War). As for the second case, “denied by the latter” uses the passive voice but its corresponding phrase in the target sentence is in the active voice. These observations indicate that the inconsistent structures or expressions between the source and target sentences could make the examples difficult for NLG models to learn well.

2) *Analysis on Data Rejuvenation:* In this section, we performed analyses on the NLG models trained with data rejuvenation in terms of learning stability, generalization capability, and the strategy for speeding up the pipeline. Unless otherwise stated, all experiments were conducted on the WMT14 En => De dataset with the TRANSFORMER-BASE model.

*Learning Stability:* First, we studied how data rejuvenation improved translation performance from the perspective of the optimization process. We presented the training loss and validation BLEU score in Fig. 8. From the training loss, we observed that the proposed data rejuvenation approach makes the model converge faster with much less fluctuation than the baseline model during the whole training process. Correspondingly, the BLEU score on the validation set is significantly boosted with our approach. These results suggest that data rejuvenation is able to accelerate and stabilize the training process.

*Generalization Capability:* Second, we investigated how data rejuvenation affected the generalization capability of NLG models with two measures, namely, Margin [49] and Gradient

TABLE VII  
RESULTS OF GENERALIZATION CAPABILITY ON THE WMT14 EN  $\Rightarrow$  DE DATASET. LARGER MARGIN/GSNR VALUES DENOTE BETTER GENERALIZATION CAPABILITY

| Model                      | Margin | GSNR   |
|----------------------------|--------|--------|
| TRANSFORMER-BASE           | 0.68   | 5.2e-3 |
| + <i>Data Rejuvenation</i> | 0.71   | 8.5e-3 |

Signal-to-Noise Ratio (GSNR) [50]. The two measures are introduced as below:

- *Margin*: Margin [49] is a classic concept in support vector machine, measuring the geometric distance between the support vectors and the decision boundary. To apply margin for NLG models, we followed Li *et al.* [51] to compute word-wise margin, which is defined as the probability of the correctly predicted word minus the maximum probability of other word types. We computed the word-wise margin over the training set and reported the averaged value.
- *GSNR*: The GSNR metric [50] is proposed to positively correlate with generalization performance. The calculation of a parameter's GSNR is defined as the ratio between its gradient's squared mean and variance over the data distribution. For NLG models, we computed GSNR of each parameter and reported the averaged value over all the parameters.

Table VII lists the results, in which the GSNR values are at the same order of magnitude as that reported by Liu *et al.* [50]. Compared with the baseline model trained on the raw data, the model trained with our *data rejuvenation* has larger Margin and GSNR, suggesting that *data rejuvenation* is able to improve the generalization capability of the final NLG models.

*Acceleration*: As shown in Fig. 1, the pipeline of data rejuvenation is time-consuming: training the identification and rejuvenation models in sequence as well as the scoring and rejuvenating procedures make the time cost of data rejuvenation more than 3X that of the standard NLG system. To save the time cost, a promising strategy is to let the identification model take the responsibility of rejuvenation. Therefore, we used the TRANSFORMER-BIG model with the large batch configuration trained on the raw data to accomplish both identification and rejuvenation. The resulting data is used to train two final models, i.e., TRANSFORMER-BIG and DYNAMICCONV.

Table VIII lists the results. With almost no decrease of translation performance, the time cost of data rejuvenation is reduced by about 33%. This makes the total time cost comparable to those data manipulation or augmentation techniques that require additional NLG systems, such as data diversification [18] and back-translation [36]. In addition, the superior performance of DYNAMICCONV (i.e., 30.4) further demonstrates the high agreement of inactive examples across architectures.

## V. MEDIUM/LOW-RESOURCE TRANSLATION TASK

In this section, we investigated whether our data rejuvenation approach also worked for medium/low-resource translation tasks. While these tasks are supported by a limited size of

TABLE VIII  
RESULTS OF SPEEDING UP (“REJ.-BIG”) ON THE WMT14 EN  $\Rightarrow$  DE DATASET. “TIME” DENOTES THE TIME OF THE WHOLE PROCESS USING 4 NVIDIA TESLA V100 GPUS

| Method     | TRANS.-BIG |      | DYN.CONV |      |
|------------|------------|------|----------|------|
|            | BLEU       | Time | BLEU     | Time |
| Standard   | 29.6       | 32h  | 29.7     | 31h  |
| Rejuvenate | 30.3       | +65h | 30.2     | +62h |
| Rej.-Big   | 30.2       | +33h | 30.4     | +32h |

TABLE IX  
EVALUATION OF TRANSLATION PERFORMANCE ON WMT16 RO  $\Rightarrow$  EN AND IWSLT14 DE  $\Rightarrow$  EN DATASETS

| Data                       | WMT16 Ro $\Rightarrow$ En |          | IWSLT14 De $\Rightarrow$ En |          |
|----------------------------|---------------------------|----------|-----------------------------|----------|
|                            | BLEU                      | $\Delta$ | BLEU                        | $\Delta$ |
| VANILLA MODEL              | 33.8                      | -        | 34.6                        | -        |
| + Remove Inactive          | 33.7                      | -0.1     | 34.3                        | -0.3     |
| + <i>Data Rejuvenation</i> | 34.2                      | +0.4     | 34.9                        | +0.3     |

training examples, there still could be inactive examples that cannot be learned well by NLG models.

1) *Dataset*: To investigate this problem, we conducted experiments on smaller translation datasets, including WMT16 Romanian  $\Rightarrow$  English (Ro  $\Rightarrow$  En) [52] and IWSLT14 German  $\Rightarrow$  English (De  $\Rightarrow$  En) [53], which consist of 608 K and 160 K sentence pairs for training, respectively. We also applied BPE [39] with 32 K merge operations for WMT16 Ro  $\Rightarrow$  En and 10K<sup>7</sup> for IWSLT14 De  $\Rightarrow$  En datasets, respectively.

2) *Model*: For WMT16 Ro  $\Rightarrow$  En, we trained a TRANSFORMER-BASE model for 50 K steps with 16 K (4096  $\times$  4) tokens per batch. For IWSLT14 De  $\Rightarrow$  En, we adopted the TRANSFORMER-IWSLT-DE-EN model implemented in Fairseq<sup>8</sup>, which contains less attention heads (i.e., 4) and a smaller dimension (i.e., 1024) for the feed-forward network (FFN). We trained the TRANSFORMER-IWSLT-DE-EN model for 50 K steps with 4,096 tokens per batch. The dropout was set to 0.3 for both models to prevent overfitting. We selected the model with the best perplexity on the validation set as the final model.

For evaluation, we generated translations by beam search with beam width 4 for both language pairs. The length penalty was 1.0 for WMT16 Ro  $\Rightarrow$  En and 0.6 for IWSLT14 De  $\Rightarrow$  En. The experimental results were reported in case-sensitive BLEU score [41].

3) *Experimental Results*: We reported the results in Table IX. As seen, the proposed data rejuvenation approach can still bring some improvements over the baseline models by +0.3 to +0.4 BLEU scores, which indicated that rejuvenating the inactive examples in small datasets also benefited the translation performance. However, due to the size reduction of datasets, we

<sup>7</sup>[Online]. Available: <https://github.com/pytorch/fairseq/blob/v0.9.0/examples/translation/prepare-iwslt14.sh>

<sup>8</sup>[Online]. Available: <https://github.com/pytorch/fairseq/blob/v0.9.0/fairseq/models/transformer.py>

TABLE X  
EVALUATION OF TEXT SUMMARIZATION PERFORMANCE ON THE GIGAWORD DATASET

| Model                      | RG-1        | RG-2        | RG-L        | BLEU        |
|----------------------------|-------------|-------------|-------------|-------------|
| <b>Gigaword</b>            |             |             |             |             |
| VANILLA MODEL              | 35.1        | 18.1        | 33.5        | 13.4        |
| + Remove Inactive          | 35.1        | 18.2        | 33.4        | 14.3        |
| + <i>Data Rejuvenation</i> | <b>35.4</b> | <b>18.5</b> | <b>33.7</b> | <b>14.4</b> |
| <b>X-Sum</b>               |             |             |             |             |
| VANILLA MODEL              | 20.2        | 4.1         | 16.2        | 2.8         |
| + Remove Inactive          | <b>20.9</b> | <b>4.3</b>  | <b>16.8</b> | <b>2.9</b>  |
| + <i>Data Rejuvenation</i> | 19.2        | 3.2         | 15.6        | 2.0         |

observed a slight drop of performance from models trained on active examples, which suggested that the size of datasets may still limit the potential of our approach.

## VI. TEXT SUMMARIZATION TASK

The most important feature of our data rejuvenation approach is that, we focus on refining the training data without touching the modification of model architectures or training strategies. Therefore, it is a general approach that can be easily extended to other NLG tasks. Here, we evaluated our approach on another popular NLG task, i.e., text summarization [7]–[11], [54], [55].

### 1) Dataset

We conducted experiments on the preprocessed data<sup>9</sup> from the Annotated English Gigaword (Gigaword) [54], consisting of 3.8 M, 189 K, and 1951 sentence pairs for training, validation, and test sets. Following Song *et al.* [8]<sup>10</sup>, we replaced the token “UNK” in the test set with “unk” to be consistent with the training set. We also tested our approach on a low-resource dataset XSum [22], which consists of 214 K, 11 K, and 11 K sentence pairs for training, validation, and test sets. We applied BPE [39] with 32 K merge operations for both datasets.

### 2) Model

For Gigaword, we trained a TRANSFORMER-BASE model with the default configurations. The dropout is set to 0.3 and the warmup step of learning rate is set to 4000. We trained the model for 100 K steps with 32 K (4096 × 8) tokens per batch and selected the model by the best perplexity. For XSum, we trained a TRANSFORMER-IWSLT-DE-EN model as for the IWSLT14 De ⇒ En MT task. For evaluation, we generated summarizations by beam search with beam width 5 with no penalty and reported ROUGE-1,2,L scores [56] and the BLEU score.

### 3) Experimental Results

We reported the results in Table X. For Gigaword, our data rejuvenation approach made non-trivial improvements over the baseline, especially in terms of BLEU score. This further demonstrated the universality of our approach. As for XSum, removing inactive examples brings noticeable improvements in terms of

TABLE XI  
INACTIVE EXAMPLES FROM THE TRAINING SET OF GIGAWORD. X, Y AND Y' DENOTE THE INPUT TEXT, THE REFERENCE SUMMARIZATION AND THE REJUVENATED SUMMARIZATION, RESPECTIVELY. THE UNDERLINED PHRASES ARE THE KEY POINTS OF EACH INPUT TEXT

|    | Side | Sentence  |
|----|------|---|
| #1 | X    | as u.s. agriculture officials work to strengthen the nation's surveillance for mad cow disease , <u>critics</u> continue to <u>poke holes</u> in the current system . |
|    | Y    | critics say voluntary mad cow testing does n ' t equal surveillance   |
|    | Y'   | critics say u.s. mad cow surveillance system is flawed  |
| #2 | X    | cuba is aiming to achieve total <u>domination</u> in the <u>boxing tournament</u> of the athens <u>olympic games</u> , targeting gold in all # # weight categories .  |
|    | Y    | cubans to dominate olympic ring in athens   |
|    | Y'   | cuba aims for total domination in athens boxing   |
| #3 | X    | logan international airport thursday became one of six big us airports testing a new <u>hand - held unk detection scanning system</u> .                               |
|    | Y    | logan testing explosives testing device   |
|    | Y'   | logan tests hand - held scanning system   |

ROUGE scores, indicating the difficulty of inactive examples. However, applying data rejuvenation to inactive examples inversely harms the performance. The main reason is that the XSum dataset is too small to learn a strong rejuvenation model, such that the quality of rejuvenated data cannot be guaranteed. Moreover, the different behaviors of NLG models on XSum and IWSLT14 De ⇒ En may result from the task differences. The goal of NLG models on IWSLT14 De ⇒ En is to translate all source information into the target language while on XSum is to extract the key information selectively. Thus, the latter task will become much more difficult when the source inputs contain a large context (e.g., thousands of tokens), which is exactly the situation of XSum.

We also presented some inactive examples from the training set of Gigaword, as shown in Table XI. Generally, in the inactive examples, the reference summarizations either deviate from or miss key points of the input text. For example #1, the reference summarization talks about the relationship between “voluntary mad cow testing” and “surveillance” while the input text is about the “holes” of the “nation’s surveillance for mad cow disease”. For example #2, the reference summarization misses the game name “boxing” that Cuba aims to win. For example #3, the key characteristic of the detection scanning system in the input text is “hand - held,” which however is missing in the reference summarization. Besides, since there is a “unk” describing the system, we do not know if it is an “explosives testing device,” which actually induces the inconsistency between the input text and the reference summarization. As shown, these problems can be fixed by rejuvenation to a considerable extent.

## VII. CONCLUSION

In this study, we propose data rejuvenation to exploit the inactive training examples for natural language generation (NLG). Our study demonstrates the existence of inactive examples in large-scale NLG datasets, which mainly depends on the data

<sup>9</sup>[Online]. Available: <https://github.com/harvardnlp/sent-summary>

<sup>10</sup>[Online]. Available: <https://github.com/microsoft/MASS/blob/master/MASS-unsupNMT/get-data-gigaword.sh>



distribution. We propose a general framework to rejuvenate the inactive examples to improve the training of NLG models, and achieve significant improvements on state-of-the-art models (e.g., TRANSFORMER and DYNAMICCONV) on benchmark datasets without modifying the model architecture and training strategies. We conduct extensive analyses to understand the properties of inactive examples and the proposed data rejuvenation approach. We successfully validate the data rejuvenation approach on various NLG tasks, including the machine translation tasks with high/medium/low resources and the text summarization tasks, which demonstrates the effectiveness and universality of our approach.

In the future, we plan to explore advanced identification and rejuvenation models that can better reflect the learning abilities of NLG models. Further, it is also important to conduct similar studies of distinguishing examples when utilizing large-scale unlabeled data.

## REFERENCES

- [1] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proc. Int. Conf. Learn. Representations*, 2015.
- [2] M.-T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2015, pp. 1412–1421.
- [3] A. Vaswani *et al.*, "Attention is all you need," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6000–6010.
- [4] X. Wang, Z. Tu, and M. Zhang, "Incorporating statistical machine translation word knowledge into neural machine translation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 12, pp. 2255–2266, Dec. 2018.
- [5] K. Chen, R. Wang, M. Utiyama, E. Sumita, and T. Zhao, "Neural machine translation with sentence-level topic context," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 27, no. 12, pp. 1970–1984, Dec. 2019.
- [6] K. Chen *et al.*, "Towards more diverse input representation for neural machine translation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 28, pp. 1586–1597, 2020.
- [7] P. Li, W. Lam, L. Bing, and Z. Wang, "Deep recurrent generative decoder for abstractive text summarization," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2017, pp. 2091–2100.
- [8] K. Song, X. Tan, T. Qin, J. Lu, and T.-Y. Liu, "MASS: Masked sequence to sequence pre-training for language generation," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 5926–5936.
- [9] S.-Q. Shen *et al.*, "Zero-shot cross-lingual neural headline generation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 26, no. 12, pp. 2319–2327, Dec. 2018.
- [10] Y. Gao, Y. Xu, H. Huang, Q. Liu, L. Wei, and L. Liu, "Jointly learning topics in sentence embedding for document summarization," *IEEE Trans. Knowl. Data Eng.*, vol. 32, no. 4, pp. 688–699, Apr. 2020.
- [11] Q. Liu, L. Chen, Y. Yuan, and H. Wu, "History reuse and bag-of-words loss for long summary generation," *IEEE/ACM Trans. Audio, Speech, Lang. Process.*, vol. 29, pp. 2551–2560, 2021.
- [12] P. Koehn and R. Knowles, "Six challenges for neural machine translation," in *Proc. 1st Workshop Neural Mach. Transl.*, 2017, pp. 28–39.
- [13] W. Wang, T. Watanabe, M. Hughes, T. Nakagawa, and C. Chelba, "Denosing neural machine translation training with trusted data and online data selection," in *Proc. 3rd Workshop Neural Mach. Transl.*, 2018, pp. 133–143.
- [14] O. Dušek, D. M. Howcroft, and V. Rieser, "Semantic noise matters for neural natural language generation," in *Proc. 12th Int. Conf. Natural Lang. Gener.*, 2019, pp. 421–426.
- [15] X. Zhang *et al.*, "An empirical exploration of curriculum learning for neural machine translation," 2018, *arXiv:1811.00739*.
- [16] E. A. Platanios, O. Stretcu, G. Neubig, B. Poczos, and T. Mitchell, "Competence-based curriculum learning for neural machine translation," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2019, pp. 1162–1172.
- [17] L. Shen and Y. Feng, "CDL: Curriculum dual learning for emotion-controllable response generation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 556–566.
- [18] X.-P. Nguyen, S. Joty, W. Kui, and A. T. Aw, "Data diversification: An elegant strategy for neural machine translation," in *Proc. 34th Int. Conf. Neural Inf. Process. Syst.*, 2020, pp. 10 018–10 029.
- [19] A. Kumar and S. Sarawagi, "Calibration of encoder decoder models for neural machine translation," 2019, *arXiv:1903.00802*.
- [20] T. Domhan, "How much attention do you need? A granular analysis of neural machine translation architectures," in *Proc. 56th Annu. Meeting Assoc. Comput. Linguistics*, 2018, pp. 1799–1808.
- [21] F. Wu, A. Fan, A. Baevski, Y. N. Dauphin, and M. Auli, "Pay less attention with lightweight and dynamic convolutions," in *Proc. Int. Conf. Learn. Representations*, 2019.
- [22] S. Narayan, S. B. Cohen, and M. Lapata, "Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2018, pp. 1797–1807.
- [23] W. Jiao, X. Wang, S. He, I. King, M. Lyu, and Z. Tu, "Data rejuvenation: Exploiting inactive training examples for neural machine translation," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2020, pp. 2255–2266.
- [24] M. P. Kumar, B. Packer, and D. Koller, "Self-paced learning for latent variable models," in *Proc. 23rd Int. Conf. Neural Inf. Process. Syst.*, 2010, pp. 1189–1197.
- [25] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 761–769.
- [26] H.-S. Chang, E. Learned-Miller, and A. McCallum, "Active bias: Training more accurate neural networks by emphasizing high variance samples," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 1003–1013.
- [27] T. Kocmi and O. Bojar, "Curriculum learning and minibatch bucketing in neural machine translation," in *Proc. Recent Adv. Natural Lang. Process.*, 2017, pp. 379–386.
- [28] W. Wang, I. Caswell, and C. Chelba, "Dynamically composing domain-data selection with clean-data selection by "co-curricular learning" for neural machine translation," in *Proc. 57th Annu. Meeting Assoc. Comput. Linguistics*, 2019, pp. 1282–1292.
- [29] X. Liu, H. Lai, D. F. Wong, and L. S. Chao, "Norm-based curriculum learning for neural machine translation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 427–436.
- [30] V. Birodkar, H. Mobahi, and S. Bengio, "Semantic redundancies in image-classification datasets: The 10% you don't need," 2019, *arXiv:1901.11409*.
- [31] A. Krizhevsky *et al.*, "Learning multiple layers of features from tiny images," *Tech. Rep. 7*, 2009.
- [32] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.
- [33] K. Vodrahalli, K. Li, and J. Malik, "Are all training examples created equal? An empirical study," 2018, *arXiv:1811.12569*.
- [34] Y. LeCun, C. Cortes, and C. Burges, "MNIST handwritten digit database," ATT Labs, 2010. [Online]. Available: <http://yann.lecun.com/exdb/mnist>
- [35] S. Wang, Z. Tu, S. Shi, and Y. Liu, "On the inference calibration of neural machine translation," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 3070–3079.
- [36] R. Sennrich, B. Haddow, and A. Birch, "Improving neural machine translation models with monolingual data," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 86–96.
- [37] J. Zhang and C. Zong, "Exploiting source-side monolingual data in neural machine translation," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2016, pp. 1535–1545.
- [38] S. Edunov, M. Ott, M. Ranzato, and M. Auli, "On the evaluation of machine translation systems trained with back-translation," in *Proc. 15th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 2836–2846.
- [39] R. Sennrich, B. Haddow, and A. Birch, "Neural machine translation of rare words with subword units," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 1715–1725.
- [40] M. Ott *et al.*, "Fairseq: A fast, extensible toolkit for sequence modeling," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics (Demonstrations)*, 2019, pp. 48–53.
- [41] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: A method for automatic evaluation of machine translation," in *Proc. 40th Annu. Meeting Assoc. Comput. Linguistics*, 2002, pp. 311–318.
- [42] M. Ott, S. Edunov, D. Grangier, and M. Auli, "Scaling neural machine translation," in *Proc. 3rd Workshop Neural Mach. Transl.*, 2018, pp. 1–9.



- [43] P. Koehn, "Statistical significance tests for machine translation evaluation," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2004, pp. 388–395.
- [44] G. Neubig, Z.-Y. Dou, J. Hu, P. Michel, D. Pruthi, and X. Wang, "compartmt: A tool for holistic comparison of language generation systems," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics (Demonstrations)*, 2019, pp. 35–41.
- [45] L. Barrault *et al.*, "Findings of the 2019 conference on machine translation," in *Proc. 4th Workshop Neural Mach. Transl.*, 2019, pp. 1–61.
- [46] Z. Tu, Z. Lu, Y. Liu, X. Liu, and H. Li, "Modeling coverage for neural machine translation," in *Proc. 54th Annu. Meeting Assoc. Comput. Linguistics*, 2016, pp. 76–85.
- [47] C. Dyer, V. Chahuneau, and N. A. Smith, "A simple, fast, and effective reparameterization of IBM model 2," in *Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics*, 2013, pp. 644–648.
- [48] C. Zhou, G. Neubig, and J. Gu, "Understanding knowledge distillation in non-autoregressive machine translation," in *Proc. Int. Conf. Learn. Representations*, 2019.
- [49] P. L. Bartlett, D. J. Foster, and M. J. Telgarsky, "Spectrally-normalized margin bounds for neural networks," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 6241–6250.
- [50] J. Liu, G. Jiang, Y. Bai, T. Chen, and H. Wang, "Understanding why neural networks generalize well through GSNR of parameters," in *Proc. Int. Conf. Learn. Representations*, 2020.
- [51] G. Li, L. Liu, G. Huang, C. Zhu, and T. Zhao, "Understanding data augmentation in neural machine translation: Two perspectives towards generalization," in *Proc. Conf. Empir. Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 5689–5695.
- [52] R. Sennrich, B. Haddow, and A. Birch, "Edinburgh neural machine translation systems for WMT 16," in *Proc. 1st Workshop Neural Mach. Transl.*, 2016, pp. 371–376.
- [53] T. He *et al.*, "Layer-wise coordination between encoder and decoder for neural machine translation," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 7955–7965.
- [54] A. M. Rush, S. Chopra, and J. Weston, "A neural attention model for abstractive sentence summarization," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2015, pp. 379–389.
- [55] J. Suzuki and M. Nagata, "Cutting-off redundant repeating generations for neural abstractive summarization," in *Proc. 15th Annu. Meeting Assoc. Comput. Linguistics*, 2017, pp. 291–297.
- [56] C.-Y. Lin, "ROUGE: A package for automatic evaluation of summaries," in *Proc. Text Summarization Branches Out*, 2004, pp. 74–81.



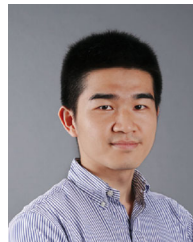
**Wenxiang Jiao** received the B.S. and M.phil. degrees from Nanjing University, Nanjing, China, in 2015 and 2017, respectively. He is currently working toward the Ph.D. degree with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, advised by Prof. Irwin King and Prof. Michael R. Lyu. His research interests mainly include natural language processing and deep learning techniques, focusing on directions like representation learning, conversation analysis, and neural machine translation.



**Xing Wang** received the Ph.D. degree from Soochow University, Suzhou, China, in 2018. He is currently a Senior Researcher with the Tencent AI Lab, Shenzhen, China. His research interests include statistical machine translation, neural machine translation, and biomedical NLP.



**Shilin He** received the Ph.D. degree from the Chinese University of Hong Kong, Hong Kong, in 2020. He is currently a Researcher with Microsoft Research Asia. His research interests include the interpretability-driven intelligent software reliability engineering, including AIOps, log analysis, model interpretability, and neural machine translation.



**Zhaopeng Tu** received the Ph.D. degree from the Institute of Computing Technology, Chinese Academy of Sciences, Nationwide, China, in 2013. He is currently a Principal Researcher with Tencent AI Lab, Shenzhen, China. He was a Postdoctoral Researcher with the University of California, Davis, CA, USA, from 2013 to 2014. He was a Researcher with Huawei Noah's Ark Lab, Hong Kong, from 2014 to 2017. He is currently working on neural machine translation and Seq2Seq learning for other NLP tasks, such as dialogue and question answering.



**Irwin King** (Fellow, IEEE) received the B.S. degree in engineering and applied science from the California Institute of Technology, Pasadena, CA, USA, and the M.S. and Ph.D. degrees in computer science from the University of Southern California, Los Angeles, CA, USA. He is currently a Professor with the Department of Computer Science and Engineering, and a former Associate Dean (Education), Faculty of Engineering with The Chinese University of Hong Kong, Hong Kong. His research interests include machine learning, social computing, web intelligence, data mining, and multimedia information processing. He was elected to IEEE Fellow (2019) for his contributions to theory and applications of machine learning in social computing.



**Michael R. Lyu** (Fellow, IEEE) received the B.S. degree in electrical engineering from National Taiwan University, Taipei, China, in 1981, the M.S. degree in computer science from the University of California, Santa Barbara, CA, USA, in 1985, and the Ph.D. degree in computer science from the University of California, Los Angeles, CA, USA, in 1988. He is currently a Professor with the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong.

His research interests include software engineering, dependable computing, distributed systems, cloud computing, mobile networking, Big Data, and machine learning. He was elected to IEEE Fellow (2004), AAAS Fellow (2007), and ACM Fellow (2015) for his contributions to software reliability engineering and software fault tolerance.