SKILLNET-NLG: GENERAL-PURPOSE NATURAL LANGUAGE GENERATION WITH A SPARSELY ACTIVATED APPROACH

Junwei Liao ⋆ Duyu Tang † Fan Zhang † Shuming Shi †

⋆University of Electronic Science and Technology of China †Tencent AI Lab

ABSTRACT

We present SkillNet-NLG, a sparsely activated approach that handles many natural language generation tasks with one model. Different from traditional dense models that always activate all the parameters, SkillNet-NLG selectively activates relevant parts of the parameters to accomplish a task, where the relevance is controlled by a set of predefined skills. The strength of such model design is that it provides an opportunity to precisely adapt relevant skills to learn new tasks effectively. We evaluate on Chinese natural language generation tasks. Results show that, with only one model file, SkillNet-NLG outperforms previous best performance methods on four of five tasks. SkillNet-NLG performs better than two multitask learning baselines (a dense model and a Mixture-of-Expert model) and achieves comparable performance to task-specific models. Lastly, SkillNet-NLG surpasses baseline systems when adapted to new tasks.

Index Terms— natural language generation, multitask model, sparsely activated model, skill network

1. INTRODUCTION

The flexibility of Transformer [1] facilitates the development of multitask models that use one model to handle multiple tasks [2, 3, 4]. These models are typically “dense” — all the model parameters are activated for all the tasks. However, it is unclear what skills are learned in which parts of the parameters. Even though tackling different tasks requires different skills [5, 6], dense models do not allow us to carry out subtle operations to choose different skills for different tasks. Moreover, when adapting a well-trained dense model to learn new tasks, all the encoded “vague” skills are transferred blindly, regardless of their relevance to the tasks.

In this work, we propose a general-purpose natural language generation model called SkillNet-NLG. The basic idea is that the model includes multiple skill modules, each of which stands for a particular skill defined in Table 1. Instead of activating all the parameters as in traditional dense models, we only activate relevant skills for a downstream task.

Fig. 1. An illustration of our sparsely activated model SkillNet-NLG for dialogue generation. Each pillar represents a skill module and pillars filled in color (e.g., yellow, green, red and purple) are activated.

As depicted in Figure 1, for the task of dialogue generation, SkillNet-NLG requires the ability to generate open-ended language ($S_1$), understand the conversational context ($S_3$) and understand natural language questions ($S_5$). Therefore, the skill modules related to $S_1$, $S_3$, $S_5$ and $S_6$ are activated. The remaining modules ($S_2$ and $S_4$) are not activated. We develop SkillNet-NLG based on Transformer [1] with an encoder-decoder structure. We modify every other layer in both Transformer encoder and decoder through replacing one feed forward network (FFN) layer with multiple FFN layers, each of which corresponds to a skill.

We conduct extensive experiments on Chinese natural language generation tasks. We define a general skill $S_6$, which works as a default skill and is always activated.

1 Work done as an intern at Tencent

2 Our approach is language agnostic. We leave the extension of SkillNet-NLG to more languages in the future.
model handles a task, only the FFN layers corresponding to relevant skills are activated. For example, for the task of dialogue generation, we only activate $S_1$, $S_3$, $S_5$ and $S_6$. The remaining modules ($S_2$ and $S_4$) are not activated. For a particular FFN layer $\text{FFN}_k$, it works same as the original FFN layer and produces skill-specific representations as follows,

$$h_k = \text{FFN}_k(\text{Attention}(h_{in})).$$

Since the size of the set of activated modules is variable, we compute the output representations using the average pooling as follows,

$$h_{out} = \frac{1}{|S|} \sum_{k=1}^{|S|} h_k,$$

where $S$ is the set of activated skills. For the the task of dialogue generation, as shown in Figure 1, $S = \{S_1, S_3, S_5, S_6\}$. The remaining operations in SkillNet-NLG are same as the original Transformer. Following [7], we only make the above changes in every other Transformer layer to avoid adding too many parameters.

2.3. Model Training

The model is trained on the mixing of training samples from all tasks. In each iteration, a mini-batch is selected from one task. A task-specific prefix is appended to the input. The model computes the cross-entropy loss between the generated text and the reference text to update the model parameters. Since the training data of different tasks are unbalanced, we follow [3] and adopt a temperature-scaled mixing strategy for data sampling. Specifically, we sample mini-batches from $N$ tasks according to probability $\{p_1, \ldots, p_N\}$:

$$p_i = \frac{D_i^\lambda}{\sum_{j=1}^N D_j^\lambda} \quad \text{with} \quad D_i = \min(n_i, K),$$

where $n_i$ is the number of training examples for the $i$-th task. $K$ is the artificial data set size limit to avoid some task’s data set becoming too large to crowd out most of the batches. $T$ is the sampling temperature. The distribution is equivalent to original data distribution for $T = 1$ and is close to the uniform distribution for larger value (e.g., $T = 1024$).

3. EXPERIMENTS

3.1. Experimental Setup

3.1.1. Task Datasets

We consider five tasks for multitask training and three tasks for adaptation. Table 2 provides all task related information including task-skill definitions, dataset statistics and evaluation metrics. **LCSTS** [8] is a large scale Chinese short text summarization dataset collected from Sina Weibo. **AdGen**

| Skill | Definition |
|-------|------------|
| $S_1$ | open-ended text generation |
| $S_2$ | non-open-ended text generation |
| $S_3$ | understand the conversational contexts |
| $S_4$ | generate text from structured data |
| $S_5$ | understand natural language questions |
| $S_6$ | generic skill |

Table 1. Skills and definitions of SkillNet-NLG.
3.1.2. Baselines

In our experiments, we compare the proposed SkillNet-NLG with the following approaches: (i) **Task-specific**: We fine-tune all the parameters of our BART model\(^9\) for each task individually. As a result, we get a total of five task-specific models for five tasks. (ii) **Dense**: We fine-tune the BART model jointly on five tasks using the same training approach with SkillNet-NLG (§2.3). (iii) **MoE**: We train a Mixture-of-Experts (MoE) baseline [7] with the same amount of six experts. For each token, we use a gating function to selectively activate the top-2 experts. The parameters of the model are initialized with our BART model and learned jointly on five tasks using the presented multitask training approach (§2.3).

3.1.3. Experimental Details

We build our SkillNet model using the implementation of BART-large by HuggingFace’s Transformers\(^{6}\) [15], which has 12 encoder layers, 12 decoder layers, 1024 hidden state dimensions and 4096 FFN dimensions. All the skill modules are initialized with FFN layers from our pre-trained Chinese BART. We conduct multi-task training for 100k steps with maximum source length of 512, maximum target length of 200 and batch size of 512. We use Adam [16] as the optimizer with \(\beta_1 = 0.9, \beta_2 = 0.999, \epsilon = 1e^{-8}\). The learning rate is warmed up over the first 10k steps to a peak value of \(3e^{-5}\), and then linearly decayed. We set the size limit \(K = 2^{21}\) and the sampling temperature \(T = 4\) after searching in \(\{1, 2, 4, 8, 16, 1024\}\). In inference stage, we use the beam search decoding and set the beam size to 4 for all tasks. Table 3 shows the specific hyper-parameters used to train three new tasks. Other training parameters are the same as for multitask training.

### 3.2. Overall Results

Table 4 shows the results of the baselines as well as SkillNet-NLG on five tasks. We average task scores as a reference to the overall performance. Overall, SkillNet-NLG performs better than task-specific fine-tuning and two multitask learning baselines (i.e., Dense and MoE) in terms of the average score. With only one model, SkillNet-NLG outperforms previous best methods on four of five tasks, demonstrating the effectiveness of the sparsely activated approach.

### 3.3. Adaptation to New Tasks

In this section, we adapt models that are well-trained on five tasks to new tasks separately.

Table 5 shows the results of different models on three new tasks. We can see that SkillNet-NLG outperforms task-specific fine-tuning and two multitask baselines. SkillNet-NLG achieves comparable performance with [19] on ZhiHu, which uses external knowledge base. SkillNet-NLG achieves a 1.22 improvement compared to the LongLM\(_{large}\), which has larger number (i.e., one billion) of parameters and is pre-trained on a large-scale in-domain data.

### 3.4. Ablation Study

In this work, we define the skills needed to accomplish a task by human experts (Table 2). Below we compare different task-skill definitions to verify the effectiveness of the proposed task-skill definition.

First, we compare four different task-skill definitions for five tasks used in multitask training: (1) **Predefined skills** uses the skill definition in Table 2; (2) **One skill per task** assigns only one skill for each task. There are no overlapped skills among tasks; (3) **Random skills** allocates randomly chosen skills for each task. Since the activated parameters of the model depend on the number of skills, for a fair comparison, the number of randomly assigned skills for each task is consistent with the number of skills defined in Table 2; (4) **In predefined skills,** the generic skill \(S_g\) works as a default skill and is always activated. Similarly, we assign random skills like (3) but always include the generic skill for each task. As shown in Table 6, the performances of all other skill definitions (2)-(4) decrease compared with predefined skills, indicating the effectiveness of the proposed task-skill definition. Moreover, (4) outperforms (3) due to the included generic skill that can learn the common ability required by all tasks.

Next, we compare six different task-skill definitions for three adapted tasks in Table 7, where (2) **Unrelated skills**
Table 2. Task related information including task-skill definitions, dataset statistics and evaluation metrics. Relevant skills (defined in Table 1) for each task are marked with ticks.

| Task                | Skills | Dataset (Train / Dev / Test) | Metrics |
|---------------------|--------|-----------------------------|---------|
| Text Summarization  | ✓ ✓ ✓ | LCSTS (2160k / 30k / 725)   | ROUGE-L |
| Advertisement Gen.   | ✓ ✓ ✓ | AdGen (114k / 1k / 3k)      | BLEU-4  |
| Question Answering   | ✓ ✓ ✓ | MATINF (740k / 100k / 210k)| ROUGE-L |
| Dialogue Gen.        | ✓ ✓ ✓ | KdConv (63k / 9k / 9k)      | BLEU-4  |
| Grammatical Error Cor. | ✓ ✓ | NLPCC (1200k / 5k / 2k)    | F0.5    |

New tasks for fine-tuning well-trained multitask models

| Task                | Skills | Dataset (Train / Dev / Test) | Metrics |
|---------------------|--------|-----------------------------|---------|
| Topic-to-Essay Gen. | ✓ ✓ ✓ | ZhiHu (27k / 300 / 2.3k)    | BLEU-2  |
| Paraphrase Gen.     | ✓ ✓ ✓ | PKUPB (490k / 10k / 10k)    | BLEU-4  |
| Story Gen.          | ✓ ✓ ✓ | OutGen (1456 / 242 / 729)   | BLEU-2  |

Table 3. Training parameters for fine-tuning well-trained SkillNet-NLG on new tasks.

| ZhiHu | PKUPB | OutGen |
|-------|-------|--------|
| Epochs   | 16    | 6      | 16     |
| Batch size | 128   | 64     | 64     |
| Learning rate | 3e-5  | 3e-5   | 5e-5   |
| Max source length | 30    | 140    | 100    |
| Max target length | 170   | 140    | 310    |

Table 4. Test results on the five task datasets during multitask training. Avg is the average score of all tasks. † indicates the score from CPT-Large [9]. ‡ indicates the score from mBART-Large [17]. * indicates the score from Mask GEC [18].

| Task        | ZhiHu | PKUPB | OutGen | Avg |
|-------------|-------|-------|--------|-----|
| LCSTS       | 41.87 | 10.63 | 20.51  | 18.50 |
| AdGen       | 20.73 | 20.76 | 36.68  | 26.27 |
| MATINF      | 18.50 | 15.75 | 15.75  | 15.75 |
| KdConv      | 36.42 | 24.77 | 24.77  | 24.77 |
| NLPCC       | 25.70 |       |        |      |
| SkillNet-NLG| 42.40 | 10.80 | 20.73  | 20.76 |

Table 5. Test results on three new task datasets. Results with † are from SCTKG(Gold-Senti) [19]. Results with ‡ are from LongLM_{large} [14].

| Task        | ZhiHu | PKUPB | OutGen | Avg |
|-------------|-------|-------|--------|-----|
| ZhiHu       | 11.02 †|       | 24.77  |     |
| PKUPB       | 10.56 | 31.88 | 25.23  |     |
| OutGen      | 10.53 | 31.93 | 24.47  |     |
| SkillNet-NLG| 10.98 | 32.02 | 25.99  |     |

4. CONCLUSION

In this work, we present a general-purpose model called SkillNet-NLG. It deals with multiple natural language generation tasks with one model. The key feature of our approach is that it is sparsely activated guided by a set of predefined skills. Only the parameters of relevant skills are activated. The advantage of such model design is that it enables us to only transfer relevant skills to learn new tasks. Experimental results on Chinese NLG tasks verify the effectiveness of our approach. In the future, we plan to adapt the model to more languages and even more modalities.
5. REFERENCES

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin, “Attention is all you need,” in Proc. of NeurIPS, 2017.

[2] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao, “Multi-task deep neural networks for natural language understanding,” in Proc. of ACL, 2019.

[3] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al., “Exploring the limits of transfer learning with a unified text-to-text transformer,” J. Mach. Learn. Res., vol. 21, no. 140, pp. 1–67, 2020.

[4] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer, “BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension,” in Proc. of ACL, 2020.

[5] Jeff Dean, “Introducing pathways: A next-generation ai architecture,” in Google Blog, 2021.

[6] Fan Zhang, Duyu Tang, Yong Dai, Cong Zhou, Shuangzhi Wu, and Shuming Shi, “Skillnet-nlu: A sparsely activated model for general-purpose natural language understanding,” arXiv preprint arXiv:2203.03312, 2022.

[7] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen, “Gshard: Scaling giant models with conditional computation and automatic sharding,” arXiv preprint arXiv:2006.16668, 2020.

[8] Baotian Hu, Qingcai Chen, and Fangze Zhu, “LCSTS: A large scale Chinese short text summarization dataset,” in Proc. of EMNLP, 2015.

[9] Yunfan Shao, Zhichao Geng, Yitao Liu, Junqi Dai, Fei Yang, Li Zhe, Hujun Bao, and Xipeng Qiu, “Cpt: A pre-trained unbalanced transformer for both chinese language understanding and generation,” arXiv preprint arXiv:2109.05729, 2021.

[10] Canwen Xu, Jiaxin Pei, Hongtao Wu, Yiyu Liu, and Chenliang Li, “MATINF: A jointly labeled large-scale dataset for classification, question answering and summarization,” in Proc. of ACL, 2020.

[11] Hao Zhou, Chujie Zheng, Kaili Huang, Minlie Huang, and Xiaoyan Zhu, “KdConv: A Chinese multi-domain dialogue dataset towards multi-turn knowledge-driven conversation,” in Proc. of ACL, 2020.