A Learnable Skill Combination Strategy for Multi-task Learning in Natural Language Understanding

Zhe Yang [1], Yi Huang [1,2] *, Yaqin Chen [1], Mengfei Guo [1], Xiaoting Wu [1], Junlan Feng [1]
[1] JIUTIAN Research; [2] Department of Computer Science and Technology, Tsinghua University, China
{yangzhe,huangyi,chenyaqin,guomengfei,wuxiaoting,fengjunlan}@cmjt.chinamobile.com

Abstract

In the realm of domain-specific natural language understanding (NLU) tasks, acquiring high-quality labeled data is often arduous, thereby posing significant challenges for effective model training. Multi-task learning (MTL) addresses these limitations by jointly optimizing multiple tasks within a unified framework. In this paper, we introduce a novel sparse NLU multi-task learning framework that decomposes the language model into modular skill components and employs a dynamic, learnable skill-combination mechanism to adaptively handle diverse tasks. Extensive experiments on benchmark NLU datasets demonstrate that our proposed method surpasses conventional multi-task learning approaches in performance.

1 Introduction

Recent advancements in multi-task learning (MTL) for natural language understanding (NLU) have significantly enhanced data efficiency and inter-task correlation, effectively mitigating both the cold-start predicament and overfitting issues (Pilault et al., 2021; Zhang et al., 2023). MTL methodologies are broadly classified into "hard-parameter sharing" and "soft-parameter sharing" paradigms (Chen et al., 2024). The former employs a fully shared representation network across all tasks, whereas the latter adopts a selective parameter-sharing mechanism to balance task-specific and shared learning.

In this paper, we introduce a MTL framework that dynamically selects skill indices through embedding similarity computation between skills and the target task. Specifically, we incorporate dedicated **embedding layers** for both tasks and skills, enabling their representations to be adaptively learned in a latent semantic space. To optimize skill selection, we employ a **differentiable thresholding mechanism** that compares task-skill similarity scores against a pre-defined threshold, thereby facilitating end-to-end gradient propagation across all model components. This approach ensures synchronized parameter updates while maintaining sparsity. Consequently, our framework achieves fine-grained task-skill information fusion through a principled index selection strategy, effectively eliminating empirical biases and enhancing the robustness of multi-task learning. Our contribution are summarized as follows:

- We propose an adaptive skill composition mechanism that leverages dual embedding spaces for both skills and tasks (Equations 2 and 3). This framework optimizes embedding parameters through joint multi-task optimization (Equations 10 and 11), enabling data-driven skill integration as opposed to the heuristic manual configuration characteristic of Skill-Net(Zhang et al., 2022) (seeing in Section B).

- We devise a differentiable optimization function that enforces synchronous skill updates within each task, effectively mitigating the load imbalance issue inherent in MMoE(Ma et al., 2018). Furthermore, our framework accommodates dynamic skill cardinality across different tasks - a critical flexibility that MMoE's rigid top-k selection mechanism fundamentally lacks (seeing in Section B).

2 Method

This section delineates the architectural framework of our novel MTL approach, as depicted in Figure 1. First, Section 2.1 presents the skill module architecture. Subsequently, Section 2.2 elaborates on our adaptive skill combination strategy, employing a differentiable thresholding mechanism that evaluates task-skill similarity against learned activation

*Corresponding Author: Yi Huang

Findings of the Association for Computational Linguistics: ACL 2026, pages 37839–37845

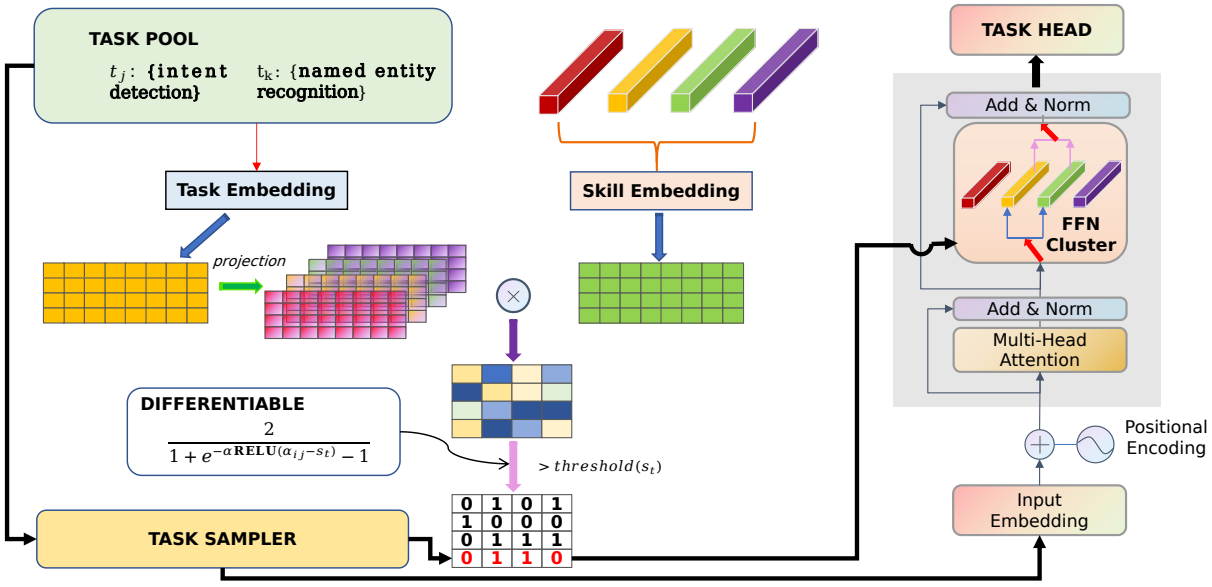

Figure 1: The overall framework of the proposed method. Original FFN structure in the encoder is extent to a FFN cluster which representing skills in the paper (Right Part). Skills activation is derived from similarity calculation between embedding of tasks and skills (Left Part).

thresholds. We introduce specialized classifier architectures tailored for representative downstream tasks in Section 2.3.

2.1 Skill modules

Inspired by MoE and Skill-Net, we adapt original language model to skill specific structures. Concretely, a typical pre-trained language model, e.g., BERT (Devlin et al., 2019), is comprised of multiple multi-head attention layers (MHA) and exercises the linkage between adjacent layers by a feed-forward network (FFN). We modify the FFN matrix into a FFN cluster for task j with each item representing a special skill, and obtain the final FFN output via the mean value calculation, i.e.,

$$\mathcal{C}_j(x) = \frac{\sum_{1\le i\le N}(\alpha_{ij} x W_i + b_i)}{\sum_{1\le i\le N}\alpha_{ij}}, \qquad (1)$$

where W_i and b_i mean the weight matrix and bias of the skill i, and α_{ij} is the coefficient to measure the contribution the skill i is on the current task j. N is the skill amount.

Distinguished from MoE method, where the weight value (i.e., α_{ij}) for an FFN routine is derived from the gate mechanism on current data, we reckon it via embedding-similarity calculation between current task type and the skill. Specifically, we attach corresponding embedding layers, i.e., E_t and E_s, for both task types and skills, as is described in Equation 2, in which n represents the number of the multiple tasks and d is the embedding dimension:

$$E_t = \mathbf{EMB}(n, d), \;\; E_s = \mathbf{EMB}(N, d). \qquad (2)$$

Furthermore, the similarity score between task type j and skill i is obtained through normalization on vectors' dot product result.

$$\alpha_{ij} = \sigma((W_{ij} E_t(j) + b_{ij}) \cdot E_s(i)), \qquad (3)$$

where $\sigma(.)$ is the *sigmoid* function, W_{ij} and b_{ij} constitute the projection network to map the task embedding into space of the skill i.

We instroduce an embedding constraint function that enforces orthogonality among skill representations, analogous to BigGAN(Brock et al., 2019), thus impels each skill to learn an exact aspect of natural language knowledge:

$$l_{orth} = \frac{|E_s E_s^T \odot (1 - I)|_F^2}{N(N-1)}, \qquad (4)$$

where E_s^T means the transpose of skill embedding matrix, I is the identity matrix and $\odot$ counts the Hadamard product between two matrices.

2.2 Skill combination strategy

α_{ij} mentioned in Section 2.1 gauges the donation that the skill i makes to task j, however, we desire a modification for it to express as either 1 or 0, which demonstrates the skill is selected as an item of the FFN cluster for task j or not, correspondingly. We

Model	TNEWS	ChnSentiCorp	AFQMC	OCNLI	OntoNotesEE	CMRC2018	Average Result
Single task fine-tuning	*52.74*	*91.42*	*68.49*	*68.98*	*58.85*	*55.60*	*66.01*
MT-DNN	**51.78**	**90.41**	65.73	67.97	49.84	57.62	63.89
MMoE	51.24	89.91	65.66	68.81	44.92	59.43	63.34
Skill-net	51.17	89.07	**67.68**	68.24	59.65	59.10	65.82
Our Model	51.29	90.08	67.10	**69.02**	**60.32**	**60.56**	**66.40**

Table 1: Evaluation results on six common NLU datasets (F1 score, %)

pre-define a threshold s_t and re-assign the value for α_{ij} with the indicator function:

$$\alpha_{ij}^{'} = \mathbb{I}(\alpha_{ij} > s_t). \tag{5}$$

However, the inherent non-differentiability of the indicator function across the entire FFN cluster impedes effective gradient backpropagation during model optimization. To address this limitation, we propose a differentiable approximation that serves as a surrogate for the original function while maintaining training stability:

$$\alpha_{ij}^{*} = \frac{2}{1 + e^{-\alpha \mathbf{ReLU}(\alpha_{ij} - s_t)}} - 1 \approx \alpha_{ij}^{'}, \tag{6}$$

where α is a positive number, **ReLU(.)** is the *Rectified Linear Unit* activation function. Obviously, if α is large in value, i.e., $\alpha \to +\infty$, the equation is satisfied that:

$$\alpha_{ij}^{*} = \begin{cases} 1 & \alpha_{ij} > s_t \\ 0 & \alpha_{ij} \leq s_t \end{cases} \tag{7}$$

Similar to Skill-Net, with this modification the FFN cluster in Equation 1 will be commuted into:

$$\begin{aligned} \mathcal{C}_j(x) &= \frac{\sum_{1 \leq i \leq N} (\alpha_{ij}^{*} x W_i + b_i)}{\sum_{1 \leq i \leq N} \alpha_{ij}^{*}} \\ &\approx \frac{\sum_{i \in S_j} (x W_i + b_i)}{|S_j|}, \\ \textbf{s.t. } S_j &= \{i | \alpha_{ij}^{'} = 1\}, \end{aligned} \tag{8}$$

where S_j is the picked skills set for task j. Moreover, the proposed framework maintains sparsity by activating only a limited subset of skills for each task. Therefore, numbers for skills and tasks in Equation 2 should satisfy the combination theory:

$$2^N - 1 \geq n. \tag{9}$$

2.3 Multi-task heads

NLU tasks are fundamentally divided into two granularity levels: sentence-level and token-level. To accommodate this dichotomy, we design task-specific heads for each category, which are appended downstream of the BERT encoder(with modification as showcased in Equation 8):

$$o(x) = \begin{cases} \sigma_s(h([\mathbf{CLS}]) & \textbf{sentence–level} \\ \sigma_s(h(t_k)) | t_k \in x\} & \textbf{token–level} \end{cases} \tag{10}$$

where **"[CLS]"** is the special token in BERT tokenizer. $h(.)$ means the modified BERT encoder. $\sigma_s(.)$ represents the *softmax* operation. Referring to Equation 4, the final loss function of our model is signified as:

$$loss = \sum (\mathbf{CE}(o(x), y)) + \beta l_{orth}, \tag{11}$$

where **CE** measures the cross-entropy value for the NLU tasks, and β is the ratio for orthogonality.

3 Experiments

We conduct experiments on several NLU task datasets to evaluate the proposed framework. In Section 3.1, the experiment settings, i.e., datasets and baselines, are displayed thoroughly. The main results and analyses are discussed in Section 3.2.

3.1 Experiment setting

- **Datasets:** **TNEWS** [1] is a news text classification dataset which comprised 15 types of labels. **ChnSentiCorp** (Tan, 2020) is a binary sentiment classification dataset for online-shopping comments. **AFQMC** (Xu and et al., 2020) is designed to predict whether two pieces of sentences exhibit semantic similarity. **OCNLI** dataset (Hu et al., 2020) aspires to map the logical relationship between a sentence pair into 3 categories. **OntoNotesEE** (Weischedel et al., 2013) is a named entity recognition dataset with 18 types, e.g., person, organization and location, etc. **CMRC2018** (Cui et al., 2019) is a machine reading comprehension dataset, which requires to extract a passage span for the given question.

[1] https://github.com/aceimnorstuvwxz/toutiao-text-classfication-dataset

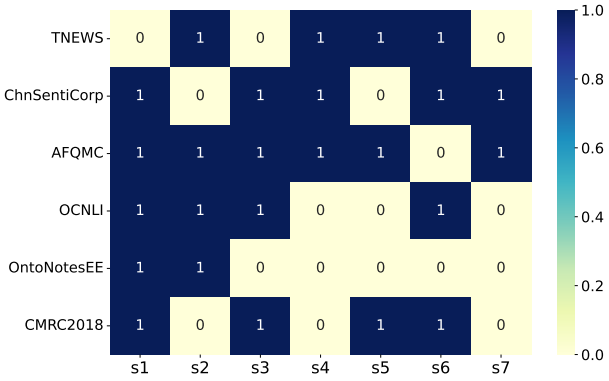

Figure 2: Skills activation among tasks, i.e., α_{ij}^{*}. Unlike MMoE, the number of activated skills varies dynamically across tasks

- **Baselines for comparison: Single task fine-tuning** (to train the tasks separately), **MT-DNN**(Liu et al., 2019), **MMoE**, **Skill-Net**. As for the last two methods, the FFN routines' number are all set to 7 which is the exact as our model. Furthermore, for Skill-Net, we retain the skill setting for tasks in Table 3.

3.2 Results

We employ MacBERT (Cui et al., 2020) as the backbone to conduct experiments and report F1 score for the evaluation results. As is displayed in Table. 1, our model exceeds 1.64% F1 value (with computing the mean difference between the results of our method and those of all other methods in the final column, i.e., **Average Result**) by average compared to other methods. Specifically, it surpasses MMoE as well as Skill-net on both TNEWS and ChnSentiCorp datasets. Mentioning OCNLI task, it also shows superiority in comparison with the baselines that exhibits an advantage with average 0.52% improvement. It is crucial to highlight that our method dominates in token-level tasks, i.e., **OntoNoteEE** and **CMRC2018**, that attains 7.0% and 2.62% ascendancy respectively among the baselines. With detailed comparison to **MMoE** and **Skill-net**, referring to MMoE, our method shows great advancement with average 8.27% (the average performance superiority over MMoE on the OntoNotesEE and CMRC2018 tasks), and 15.4% for OntoNotesEE. As for Skill-net, it still dominates with average 1.07%. Notably, it is unnecessary to pre-design a task-skill mapping manually, enabling to save labor costs in Skill-net.

We evaluate the skills activation states after the training phase in terms of α_{ij}^{*} mentioned in Equation 6, as is demonstrated in Figure 2 .

Model	TNEWS	ChnSentiCorp	AFQMC
Ours	**51.29**	**90.08**	67.10
$\beta = 0$	50.39	89.40	**67.47**

OCNLI	OntoNotesEE	CMRC2018	Average
69.02	**60.32**	**60.56**	**66.40**
67.93	59.01	58.85	65.51

Table 2: Evaluations on embedding orthogonality (%).

Additionally, we verify the skills embedding constraint states with calculating the value of $|E_s E_s^T|$ in Equation 4. In the left part of the Figure 3, embedding similarity matrix among the skill pairs unfolds as a diagonal pattern which illustrates an orthogonality in skills embedding space, hence it alleviates the embedding mode collapse. Nevertheless, the right part exhibits skill overlapping phenomenon with several off-diagonal elements being large in values, i.e., the similarity score of 17 between skill 6 and skill 7. Referring to Table. 2, the F1 score on average result descent by 0.89% with $\beta = 0$ that suggests a significance for the embedding constraint, however, it still exceeds MMoE to much degree, i.e., 2.17% for the average result and 13.93% for the harder task, i.e., OntoNotesEE. Thus the framework design of our method contributes more to the final results, and the orthogonality further enhances the model ability.

4 Conclusion

We propose a novel sparse MTL framework for NLU that extends the conventional FFN architecture in transformer encoders to a clustered FFN ensemble, enabling dynamic network composition tailored to specific tasks. In contrast to Skill-Net, our method eliminates heuristic task-skill mapping by instead learning optimal associations through embedding similarity metrics. To optimize skill representation learning, we introduce an embedding orthogonality constraint that minimizes the inner product between any pair of skill embeddings. Furthermore, we implement a differentiable thresholding mechanism, thereby maintaining complete differentiability throughout the network for effective end-to-end training. Comprehensive empirical evaluation across NLU datasets demonstrates the superior performance and robustness of our proposed framework.

Limitations

In this paper, we propose a sparsified multi-task NLU framework that addresses the limitations of Skill-Net method (such as manually designed task-skill mappings) and the load-balancing issues in MMoE. Our approach has demonstrated effectiveness across multiple NLU tasks. In the future, we will extend the validation of this algorithm to NLG tasks and evaluate its performance on popular LLMs.

Acknowledgments

This work is supported by China Mobile Strategic Project (R26110S3, R24113J4).

References

Andrew Brock, Jeff Donahue, and Karen Simonyan. 2019. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*.

Vinod Kumar Chauhan, Jiandong Zhou, Ping Lu, Soheila Molaei, and David A. Clifton. 2024. A brief review of hypernetworks in deep learning. *Artificial Intelligence Review*, 57(9).

Shijie Chen, Yu Zhang, and Qiang Yang. 2024. Multi-task learning in natural language processing: An overview. *ACM Comput. Surv.*, 56(12).

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for Chinese natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 657–668, Online. Association for Computational Linguistics.

Yiming Cui, Ting Liu, Wanxiang Che, Li Xiao, Zhipeng Chen, Wentao Ma, Shijin Wang, and Guoping Hu. 2019. A span-extraction dataset for Chinese machine reading comprehension. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5883–5889, Hong Kong, China. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Hai Hu, Kyle Richardson, Liang Xu, Lu Li, Sandra Kübler, and Lawrence Moss. 2020. OCNLI: Original Chinese Natural Language Inference. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3512–3526, Online. Association for Computational Linguistics.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4487–4496, Florence, Italy. Association for Computational Linguistics.

Jiaqi Ma, Zhe Zhao, Xinyang Yi, Jilin Chen, Lichan Hong, and Ed H. Chi. 2018. Modeling task relationships in multi-task learning with multi-gate mixture-of-experts. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '18, page 1930–1939, New York, NY, USA. Association for Computing Machinery.

Weicheng Ma, Renze Lou, Kai Zhang, Lili Wang, and Soroush Vosoughi. 2021. GradTS: A gradient-based automatic auxiliary task selection method based on transformer networks. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5621–5632, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jonas Pfeiffer, Sebastian Ruder, Ivan Vulić, and Edoardo Ponti. 2023. Modular deep learning. *Transactions on Machine Learning Research*. Survey Certification.

Jonathan Pilault, Amine El hattami, and Christopher Pal. 2021. Conditionally adaptive multi-task learning: Improving transfer learning in NLP using fewer parameters & less data. In *International Conference on Learning Representations*.

Noam Shazeer, *Azalia Mirhoseini, *Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*.

Songbo Tan. 2020. Chnsenticorp.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA. Curran Associates Inc.

Ralph Weischedel, Martha Palmer, Mitchell Marcus, Eduard Hovy, Sameer Pradhan, Lance Ramshaw, Nianwen Xue, Ann Taylor, Jeff Kaufman, Michelle Franchini, and 1 others. 2013. Ontonotes release 5.0 ldc2013t19. *Linguistic Data Consortium, Philadelphia, PA*, 23(170):20.

Liang Xu and et al. 2020. CLUE: A Chinese language understanding evaluation benchmark. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4762–4772, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Fan Zhang, Duyu Tang, Yong Dai, Cong Zhou, Shuangzhi Wu, and Shuming Shi. 2022. Skillnet-nlu: A sparsely activated model for general-purpose natural language understanding.

Zhihan Zhang, Wenhao Yu, Mengxia Yu, Zhichun Guo, and Meng Jiang. 2023. A survey of multi-task learning in natural language processing: Regarding task relatedness and training methods. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 943–956, Dubrovnik, Croatia. Association for Computational Linguistics.

Hao Zhao, Jie Fu, and Zhaofeng He. 2023. Prototype-based HyperAdapter for sample-efficient multi-task tuning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4603–4615, Singapore. Association for Computational Linguistics.

A Results on $E_s E_s^T$

Evaluation results on $E_s E_s^T$ constraint, as is displayed in Figure 3.

B Related Work

Multi-Task Deep Neural Networks (MT-DNN) (Liu et al., 2019) is a representative hard sharing model which shares the bottom layers, i.e., lexicon and transformer (Vaswani et al., 2017) encoders, across multiple tasks, whilst designs different classification layers for task-specific. The whole encoder sharing design is efficient among kindred tasks, yet demonstrates boundedness for tasks at various granularity (Ma et al., 2021), i.e., sentence-grained and token-grained tasks which conduct attention mechanism at different levels and precipitate sub-optimal encoder parameters.

Referring to soft sharing that divides representation layers into task-shared and task-specific features, Multi-gate Mixture-of-Experts (MMoE) method (Ma et al., 2018) exhibits popularity to distinguish differences among tasks. The authors combine the shared bottom structure and the MoE strategy which trains a corresponding gate network for each task and selects expert routines via the gate values. However, the gate values are usually larger than zero that makes the framework dense and time-consuming for inference. "Sparsity-gated MoE" method (Shazeer et al., 2017) attaches sparsity and noise components into original gate network that utilizes "keep-top-k" operation for experts selection. The operation establishes parameters update for experts at different training stages which may result in expert collapse. Moreover, the top-k setting will limit the model expression for various tasks. Skill-net (Zhang et al., 2022) proposes a sparse multi-task learning framework that allocates several skill routines, i.e., the expert routines in the MoE method, manually for a task. Whereas, it suffers from the experiential bias for static skill index selection.

The task type information is also essential for deriving task-specific structures (Pfeiffer et al., 2023). In paper (Zhao et al., 2023), the authors define an instance-dense retriever to map the instance encoding into the task type space. After that, the type embedding is utilized to excite the HyperNetwork (Chauhan et al., 2024) to generate parameters for task-related adapter. Nonetheless, the method trains the type learner and the task processing module separately, thus cannot ensure an optimal pa-

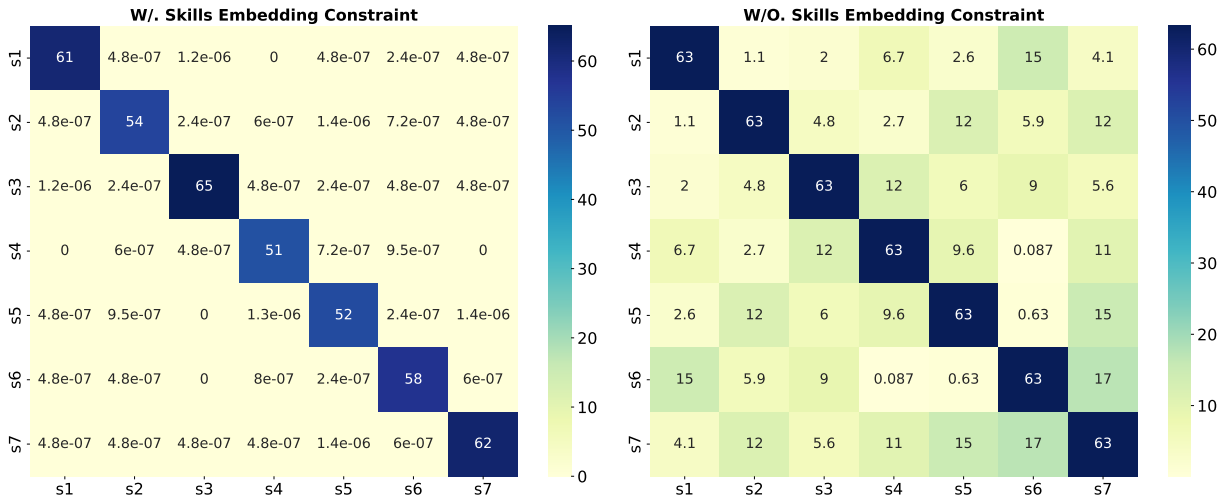

Figure 3: Skills embedding constraint visualization, i.e., $|E_s E_s^t|$.

rameters for linkage between the type embedding and the NLU models.

C Training Settings

• **Hyper-parameters:** Following the datasets and previous methods, numbers for tasks and skills, i.e., n and N in (2), are **6** and **7** separately. The embedding dimension d is set to **64**. With respect to the differentiable approximation in (6), the similarity threshold s_t and the coefficient α are set to **0.35** and $\mathbf{10^9}$. We assign value of $\mathbf{10^{-3}}$ to skill's embedding constraint ratio, i.e., β, in (11). In terms of the system, all the experiments are conducted on a single *Tesla V100S-PCIE-32GB* GPU.

• **Skill-net setting:** Skill activation for skill-net is shown in Table 3.

	s1	s2	s3	s4	s5	s6	s7
TNEWS	✓						✓
ChnSentiCorp	✓			✓			✓
AFQMC	✓		✓			✓	✓
OCNLI	✓		✓				✓
OntoNotesEE		✓					✓
CMRC2018		✓	✓		✓		✓

Table 3: Activated skills in skill-net baseline.

D LLM Usage Clarification

Throughout the paper, the use of LLMs is solely restricted to the polishing of textual elements, such as lexical or phrasal substitutions, and does not extend beyond this scope.