A Progressive Learning Strategy for Medical Natural Language Understanding

Zhe Yang,Yi Huang*, Mengfei Guo, Yaqin Chen, Xiaoting Wu, Junlan Feng and Chao Deng
JIUTIAN Team, China Mobile Research Institute
{yangzhe,huangyi,guomengfei, chenyaqin,wuxiaoting,fengjunlan,dengchao}@chinamobile.com

Abstract

Medical natural language understanding (NLU) tasks seek to extract clinically relevant information—such as diagnostic intent, symptomatic manifestations, laboratory findings, and therapeutic regimens—from medical dialogues or textual data. Regrettably, the paucity of annotated medical datasets often impedes the development of robustly trained models across diverse tasks. A promising approach involves decomposing neural networks into modular skill components, thereby facilitating the transfer of acquired knowledge from trained tasks to novel ones. Nevertheless, in multi-task learning frameworks, the indiscriminate aggregation of skill modules into a unified architecture may result in suboptimal skill refinement. To address this limitation, we introduce a progressive learning paradigm wherein each task is constrained to leverage only the network structures of tasks preceding it in a predefined difficulty hierarchy, thereby maximizing knowledge assimilation from less complex subtasks. For empirical validation, we select four pivotal medical NLP tasks: Single Sentence Intention Classification (SSIC), Sentence Pair Relationship Judgment (SPRJ), Named Entity Recognition (NER), and Classifying Positive and Negative Clinical Findings (CPNCF). Experimental results demonstrate that our proposed strategy yields consistent performance enhancements on the CPNCF task across multiple datasets.

1 Introduction

In recent years, remarkable advancements have been achieved in the exploration of language models employing multi-task learning (MTL) paradigms. This innovative approach focuses on either pre-training or fine-tuning shared knowledge representations across diverse learning tasks, thereby significantly enhancing both the efficacy of training data utilization and overall task performance(Pilault et al., 2021; Zhang et al., 2023). In the framework of MTL, each distinct task compels the model to acquire specific facets of knowledge that exhibit partial intersection yet maintain unique characteristics. The approach not only optimizes computational resources but also facilitates accelerated knowledge transfer through synergistic learning mechanisms. For traditional MTL methods, which generally learn multiple tasks by sharing representations (e.g., **M**ulti-**T**ask **D**eep **N**eural **N**etworks (MT-DNN) (Liu et al., 2019)) or allocating representations according to learning parameters (e.g., **M**ulti-gate **M**ixture-**o**f-**E**xperts (MMoE) (Ma et al., 2018)), acquiring optimal task-specific parameters proves challenging when confronted with imbalanced data distribution, particularly scenarios where vast datasets are available for rudimentary tasks while only limited data exists for intricate ones. Consequently, it exhibits negligible transferability when applied to novel tasks. Skillnet (Zhang et al., 2022a), as an advanced MTL method, designs a model structure of task disassembly based on skills. By splitting the task to be trained into multiple skills, the model can quickly reuse the previous skills and capabilities in the process of participating in the new task training, so as to improve the training efficiency.

Nevertheless, annotated data in medical domain is usually limited and scattered across tasks (Wu et al., 2020; Fries et al., 2021). This is particularly critical in the case of CPNCF, which involves the extraction of medical entities—specifically, symptom recognition—and the subsequent determination of their presence or absence in patients. As a foundational component of numerous clinical applications, i.e., patient report generation, CPNCF's inherent complexity raises concerns regarding model convergence and comprehensive learning. Furthermore, existing methodologies fail to account for the disparate learning difficulties across tasks, resulting in lackluster progress for simpler tasks and suboptimal convergence rates for more complex ones. This

*Corresponding authors

discrepancy stems from the necessity for holistic decision-making, integrating multiple knowledge domains, which ultimately impedes both model efficiency and performance (Guo et al., 2018; Liang and Zhang, 2020).

Consequently, we propose a progressive MTL framework that establishes hierarchical linkages between primary tasks and their subordinate challenging variants, facilitating unidirectional knowledge transfer. This architecture enables foundational knowledge acquired from simpler tasks to systematically scaffold the learning of more complex counterparts. Specifically, we select elementary tasks - SSIC, SPRJ, and NER - as foundational building blocks to support the advanced CPNCF task. The training sequence is meticulously organized according to **task complexity**, which we quantitatively assess through the requisite information volume for task execution. Architecturally, we implement these task interdependencies through a **residual network framework**, which enforces a strict information flow paradigm from elementary to advanced task modules.

Our contributions can be summed up in the following three points:

• We introduce two novel complexity-driven methodologies that systematically evaluate and rank task difficulty based on the requisite information during the learning process. This framework serves as the foundational paradigm and critical initial phase for our progressive MTL architecture.

• We propose a progressive MTL framework that systematically transfers knowledge from elementary tasks to facilitate the learning of more complex counterparts via the residual network.

• We conduct extensive experiments on multiple benchmark medical datasets, empirically validating the efficacy of the proposed methodology.

2 Related Work

Many existing works focus on designing multi-task fusion methods (Yu et al., 2024; Chen et al., 2024), including cross fusion of text representations, fusion design of classification layers, etc. These fusion architectures can be divided into two types: parallel design and hierarchical design.

In parallel design, e.g., MT-DNN, most of the model structure is shared among tasks, and only the task-related output part, i.e., the top classification layer, is unique to each other. To more remotely distinguish the differences among tasks, some works have designed unique representation layer sharing mechanisms, i.e., with delicate task-shared and task-specific parts(Sun et al., 2020; Liu et al., 2021; Chai et al., 2022). Skill-net is a typical model for this type of parallel design. It decomposes the essential competencies for NLU into discrete cognitive skills, including lexical semantic comprehension and textual sentiment analysis. These foundational skills exhibit cross-task transferability while maintaining task-specific specialization. To more precisely model the intricate interplay between shared linguistic features and individual task requirements, contemporary approaches have developed dedicated memory architectures with adaptive gating mechanisms (Ma et al., 2018; Dankers et al., 2019). For instance, MMoE shares experts structure among tasks and establish distinguished gate networks to activate them. In order to make trade-off between knowledge availability and sparsity for MoE, Zhao et al. transfer partial knowledge from unselected experts to the selected based on hypernetworks.

Dissimilarly, the hierarchical structure fuses features at different depths. For example, some works integrate word features of token granularity tasks (e.g., simile extraction) as additional information into that of sentence granularity tasks (e.g., simile classification) to assist the improvement of the latter (Liu et al., 2018). Additionally, some works employ the output of upstream tasks as additional input for the downstream to build a pipeline hierarchical structure (Huang et al., 2022; Nguyen et al., 2022). Zhou et al. solve the problem of entity reference by constructing a three-level structure.

Despite the aforementioned structural designs, existing models invariably overlook the critical role of task difficulty in MTL. Intuitively, human cognition tends to deconstruct intricate tasks into simpler sub-tasks, progressively mastering the overarching skill through incremental learning. In this paradigm, acquiring proficiency in elementary tasks inherently furnishes auxiliary knowledge that facilitates the comprehension of more complex objectives. This phenomenon is particularly salient in the medical domain, where rudimentary tasks often constitute integral components of higher-order challenges. For instance, CPNCF can be operationalized through the synergistic integration of NER and contextual relationship analysis. Consequently, explicitly modeling the hierarchical interdependencies among tasks can significantly augment MTL efficacy and enhance overall performance.

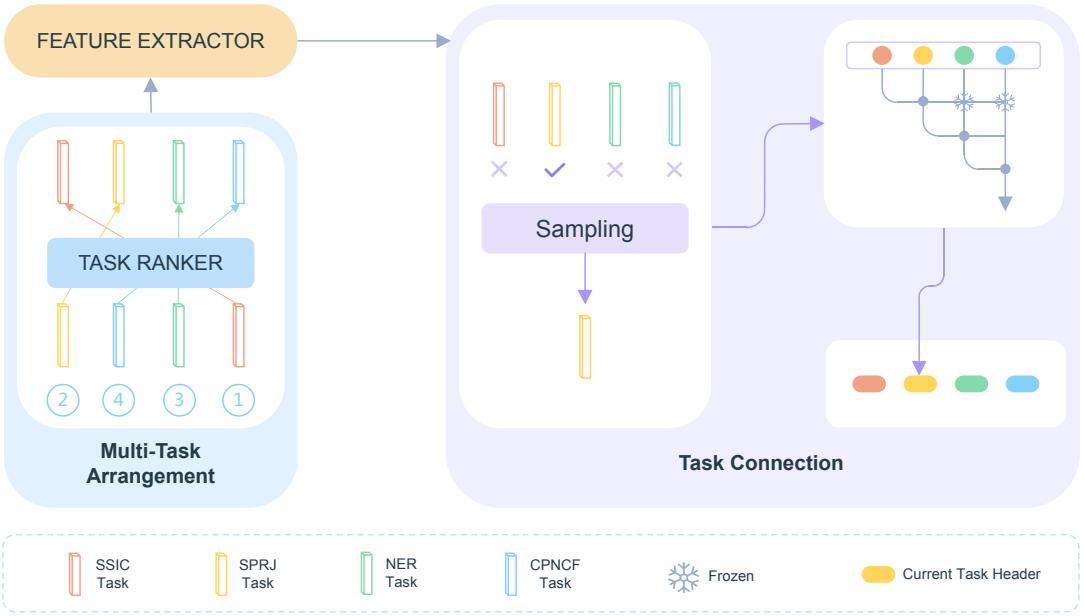

Figure 1: The overall framework of our model. We firstly rank the four tasks to the order of 'SSIC->SPRJ->NER->CPNCF' according to their ease of understanding, and a feature extractor, e.g., BERT (Devlin et al., 2019) is employed for instances of the tasks. A progressive learning architecture is established for multi-task learning where routes of the easier tasks are added to harder ones for feature augmentation. During training or inferring, only routes of the current task are activated.

3 Proposed Method

This section specifies the proposed model for MTL, as is shown in Figure 1. We initially curate a set of elementary NLU tasks in the medical domain to serve as auxiliary learning components for the CPNCF task. These selected tasks possess either well-established annotation protocols or substantial labeled datasets, facilitating robust model training. Then we implement a residual connectivity paradigm that systematically links each task's network pathway to its more complex counterparts. During training, we employ selective activation, whereby only the network modules corresponding to the current target task and its subordinate tasks are engaged.

3.1 Multi-tasks Arrangement

We choose several typical medical NLU tasks, namely, SSIC, SPRJ and NER for CPNCF accomplishment. For any tasks $T_{rd} = \{t_i | i \leq len(T_{rd})\}$, We rank them as:

$$T_{order} = \{t_j | t_j \in T_{rd} \wedge d(t_j) < d(t_{j+1})\}, \quad (1)$$

where $d(t_j)$ is a gauge to estimate the difficulty of the task t_j. The measurement works with the information the task requires. In general, we consider task A is more difficult than task B when A requires extra information to implement. In fact, ν-usable information (Ethayarajh et al., 2022) is a possible solution to this quantification but with complex implementations, hence we propose two strategies to solve this ranking problem intuitively.

• **Coarse-grained tasks are simpler than fine-grained tasks:** Coarse-grained tasks, exemplified by sentence-level comprehension, typically operate through aggregation mechanisms that pool finer-grained features (Hashimoto et al., 2017; Kruengkrai et al., 2020), e.g., token-level representations, without requiring exhaustive analysis of individual token units. Notably, these higher-level tasks simultaneously provide valuable contextual information that enhances performance on their fine-grained counterparts. This granularity dichotomy is clearly manifested in our task selection: SSIC and SPRJ operate at sentence-level granularity, whereas NER and CPNCF require precise token-level analysis. Hence:

$$d(t_s) < d(t_t), \quad (2)$$

where $t_s \in \{t_{ssic}, t_{sprj}\}$ and $t_t \in \{t_{ner}, t_{cpncf}\}$.

• **Up-stream tasks are simpler than down-stream tasks:** To illustrate, relation extraction task inherently depends on entity pairs identified through NER, establishing a clear hierarchical de-

pendency. Similarly, task-oriented dialogue system exhibits greater complexity than basic intent detection, as response generation necessitates intention-aware filtering mechanisms (Qin et al., 2019; Weld et al., 2022). Conventional approaches typically employ pipeline architectures to sequentially address these interdependent tasks. When explicit task dependencies are ambiguous, the relationship can be inferred through analysis of implicit informational prerequisites. With respect to SSIC and SPRJ, the latter necessitates comprehensive sentence-level semantic comprehension to accurately infer inter-sentential relationships. As for NER and CPNCF, considering the clinical presentation "cough, expectoration with chest and back pain, without hot flashes, night sweats, fatigue, and weight loss", NER simply extracts discrete symptom entities (e.g., "cough", "back pain", "hot flashes"), while CPNCF performs advanced clinical phenotyping by determining assertion status and incorporating contextual analysis from both the host sentence and adjacent discourse (e.g., "hot flashes-negated findings"). This demonstrates CPNCF's sophisticated requirement for contextual interpretation and clinical reasoning beyond NER.

Ultimately, we will get a task rank as:

$$d(t_{ssic}) < d(t_{sprj}) < d(t_{ner}) < d(t_{cpncf}) \quad (3)$$

3.2 Tasks Connection

We employ BERT as the backbone for our model architecture. BERT is composed of multiple multi-head attention layers, being connected by feed-forward neural networks (FFNN). Similar to many other works, we keep the attention layer unchanged and only modify the linking FFNN for task specific.

Concretely, we architecturally extend each FFNN module into a task-specific FFNN cluster architecture, wherein each pathway in the cluster is dedicated to a particular task. As illustrated in Figure 2, we denote the aforementioned tasks of ordered in Equation 1 as t_1, t_2, t_3, t_4 respectively, and develop the single FFNN into a four-pathway cluster. Information flow in the cluster of k_{th} layer can be expressed as:

$$\begin{aligned} h_k(t_1) &= \mathcal{F}_{k1}(t_1), \\ h_k(t_2) &= W_{21} * h_k(t_1) + \mathcal{F}_{k2}(t_2), \\ &\ldots, \\ h_k(t_n) &= \sum_{1 \le i \le n-1} W_{ni} * h_k(t_i) + \mathcal{F}_{kn}(t_n), \end{aligned} \quad (4)$$

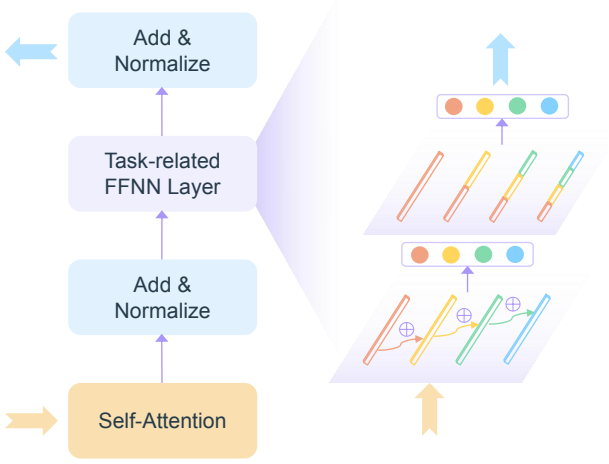

Figure 2: Similar to Switch-Transformers (Fedus et al., 2022), we implement task-specific modifications to FFNN architecture. As illustrated in the right panel of the figure, the chromatic progression from red to blue represents four distinct tasks of ascending complexity, with each color-coded bar corresponding to a dedicated FFNN pathway.

where $\mathcal{F}_{ki}(t_i)$ means the corresponding FFNN pathway of task t_i in layer k, W_{ij} is a trainable weight to measure the contribution from t_j to t_i. We can also set each W_{ij} as constant 1 for simplicity. From the formula, each task will derive its final representation iteratively for further processing.

During the training or inferring process, we will only activate the output of the current task, e.g., t_m, and other outputs will be frozen.

$$h_k = \sum_{1 \le i \le n} \mathbb{I}(i = m) * h_k(t_i), \quad (5)$$

where $\mathbb{I}$ is an indicator function to measure if current path belongs to the processing task.

Following the BERT-based feature extraction, we implement dedicated classifier heads for each downstream task to produce task-specific predictions. Specifically, for **SSIC** task, which involves categorical assignment of clinical utterances (e.g., classifying sentences into domains such as medical advice), we employ a *softmax-activated* output layer operating on the contextualized "[CLS]" token representation to perform multi-label classification. We consume *sigmoid* function for task **SPRJ** to judge the relationship between a couple of medical texts:

$$o(t) = \begin{cases} \textbf{softmax}(h([\textbf{CLS}]) & \textbf{SSIC} \\ \textbf{sigmoid}(h([\textbf{CLS}]) & \textbf{SPRJ} \end{cases} \quad (6)$$

where $h(.)$ is the last hidden states as the representation.

NER constitutes a sequence labeling task that requires assigning a categorical tag to each token in the input sequence. Specifically, we employ the BIO annotation scheme, where: B-T denotes the beginning token of an entity of type T, I-T represents an internal token within an entity of type T, and O indicates tokens outside any entity boundaries. The **CPNCF** task involves identifying potential clinical findings (e.g., symptoms or signs) within clinical text and determining their assertion status (positive or negative). Following the NER paradigm, we employ the BIO labeling scheme for token-level annotation. Crucially, accurate assertion classification necessitates comprehensive contextual understanding. To facilitate this, our model architecture incorporates enriched contextual information by surrounding the target text with special context markers ("[C]"). However, these contextual markers are excluded from loss computation during model training to maintain focus on the primary clinical findings.

$$o(t)=\begin{cases} \{\mathbf{softmax}(h(x))\}(x \in \mathcal{X}) & \mathbf{NER} \\ \{\mathbf{softmax}(h(x))\}(x \in \mathcal{X}-\mathcal{C}) & \mathbf{CPNCF} \end{cases} \tag{7}$$

where $\mathcal{X}$ denotes tokens of the medical sentence. $\mathcal{C}$ represents the context tokens in CPNCF.

4 Experiments

In this section, we present a comprehensive experimental evaluation of our proposed model across two distinct series of medical NLU datasets. Initially, we delineate the experimental configuration, encompassing the benchmark datasets, comparative baseline methodologies, and performance evaluation metrics. Subsequently, we demonstrate the empirical results for each task and conduct rigorous comparative analyses with baseline methods. Finally, we perform systematic ablation studies to critically assess the impact of both the intermediate task formulation and the sequencing of task orders within our framework.

4.1 Experimental Setup

4.1.1 Datasets

MTL on mixture source datasets: We curate publicly available datasets sourced from CBLUE (Zhang et al., 2022b), meticulously selected to align with their respective task objectives, namely:

• **KUAKE-QIC** dataset for task **SSIC**, comprises curated selections extracted from an extensive corpus of medical conversations. Its primary objective is to assess the medical-related intent of the inquirer based on their queries.

• **CHIP-STS** dataset is systematically structured to generate question pairs, which serve as samples for task **SPRJ**. A positive sample is defined as a question pair that conveys the same underlying medical concept.

• **CMeEE** dataset extracts biomedical entities from authoritative sources. These **NER** classifications encompass medical equipment, clinical procedures, diagnostic examinations, among other relevant biomedical concepts.

• **CHIP-MDCFNPC** dataset processes pipeline involves two key stages: initial alignment and SOAP (Subjective, Objective, Assessment, Plan) classification, followed by NER and assertion labeling (positive/negative) on the Subjective and Assessment component. We employ its NER and assertion labels to conduct task **CPNCF**.

MTL on the same source dataset: The public **IMCS21** dataset (Chen et al., 2022) comprises authentic online doctor-patient dialogues that have undergone rigorous multi-level manual annotation, encompassing named entities, dialogue intentions, symptom labels, and medical reports. Aligned with the model tasks outlined in Section 3.1, we specifically utilize dialogue intentions, named entities, and symptom labels for tasks **SPRJ**, **NER**, and **CPNCF** respectively. Additionally, we formulate task **SPRJ** based on dialogue content analysis. Positive samples for the SPRJ task consist of semantically coherent adjacent question-answer pairs, while negative samples are generated through random substitution of either questions or answers within these pairs.

4.1.2 Baselines for comparison

We compare the proposed model with the vanilla fine-tuning and several typical MTL methods:

	s1	s2	s3	s4	s5	s6	s7
SSIC	✓			✓			✓
SPRJ	✓		✓			✓	✓
NER		✓					✓
CPNCF		✓	✓		✓		✓

Table 1: Activated skills in skill-net baseline.

Model	SSIC	SPRJ	NER	CPNCF	Average Result
Mixture Dataset					
Vanilla fine-tuning	81.3	87.8	53.7	50.7	68.4
MT-DNN (Liu et al., 2019)	79.9	86.6	53.6	60.0	70.0
MMoE (Ma et al., 2018)	79.1	87.2	68.8	83.7	79.7
Skill-net (Zhang et al., 2022a)	79.4	87.4	66.4	68.3	75.4
Our Model	80.5	86.3	54.5	**84.7**	76.5
IMCS21 Dataset					
Vanilla fine-tuning	85.6	89.4	65.0	41.1	70.3
MT-DNN (Liu et al., 2019)	85.3	89.5	86.0	45.1	76.5
MMoE (Ma et al., 2018)	84.5	87.7	92.4	73.7	84.6
Skill-net (Zhang et al., 2022a)	85.0	89.4	81.5	59.4	76.8
Our Model	85.1	89.3	82.4	**89.2**	86.5

Table 2: Evaluation results (%) on two types of datasets. The purpose is to improve the performance of the complex task of **CPNCF** via multi-task learning.

Vanilla: Each task undergoes specialized fine-tuning utilizing its own dedicated BERT architecture, with complete parameter isolation maintained across all tasks.

MT-DNN (Liu et al., 2019): We employ a shared BERT architecture as a universal feature extractor across all tasks, while exclusively adapting the classifier layers (consistent with the approach described in Section 3.2) to create task-specific output modules.

MMoE (Ma et al., 2018): constitutes a MTL architecture employing the **mixture of experts** paradigm (Shazeer et al., 2017). While the conventional MMoE model incorporates a shared bottom representation layer across tasks, our comparative experiment introduces a structural modification wherein we replace the FFNN in each BERT layer with MMoE components. This architectural adaptation ensures a comparable degree of structural parity with our proposed method, thereby maintaining experimental fairness in the comparative analysis.

Skill-net (Zhang et al., 2022a): We introduce targeted modifications to the Skill-net architecture to accommodate our specific task requirements, particularly since the CPNCF task is not included in the original framework. Accordingly, as demonstrated in Table 1, we implement seven specialized skills: s1 (sequence-level semantic comprehension), s2 (token-level semantic interpretation), s3 (cross-segment interaction analysis), s4 (sentiment analysis), s5 (natural language question processing), s6 (medical domain text understanding), and s7 (generic linguistic processing).

4.1.3 Evaluation Metrics

Given that all tasks fundamentally constitute classification problems, we employ the Micro-F1 score as our primary evaluation metric to ensure consistent and comprehensive performance assessment. Furthermore, all experimental procedures are executed utilizing a single *Tesla V100S-PCIE-32GB* GPU to maintain computational consistency across trials.

4.2 Evaluation Results

It can be seen from Table 2 that our model is superior to existing algorithms in many aspects:

First, on tasks like SSIC and SPRJ, our model has comparable effects to **Vanilla fine-tuning** method which demonstrates that our progressive framework effectively maintains performance on simpler tasks without incurring performance degradation, despite the incorporation of more complex task objectives. Meantime, compared with the **MMoE** and **Skill-net** architecture, our model has a slight improvement on simple tasks.

However, for more complex task such as CPNCF - which requires contextual assessment of entity polarities (positive/negative effects) within center sentences - the Vanilla model demonstrates significantly diminished performance efficacy, with F1 score being only 50.7/41.1% for two types of datasets. Our proposed model leverages residual architecture to effectively integrate knowledge acquired from simpler tasks, demonstrating significant performance gains of 84.7% and 89.2% respectively. While Skill-net does incorporate certain competencies derived from elementary task

training, the interdependencies among these skills remain markedly less defined than those enabled by our residual architecture and progressive learning paradigm. It follows logically, that our framework demonstrates superior performance in handling complex tasks. When contrasted with the conventional MT-DNN architecture, several inherent limitations become apparent. While this design offers simplicity, it suffers from two critical shortcomings: First, the complete overlap of multi-task representation layers introduces instability during optimization, manifesting as erratic training dynamics. Second, the absence of explicit task relationship modeling results in largely isolated learning processes, thereby impeding cross-task knowledge transfer and synergistic learning. Consequently, the experimental results demonstrate that our model achieves a performance advantage exceeding 20% on task CPNCF and 10% on average over MT-DNN. As for **MMoE** method, our model has a rivaling score on average results and a higher result on CPNCF task.

We also conduct experiments on low resource settings, where we sample k (=8/16) training items for each label of all the tasks (we also make a subset of 1000+ samples of each task for evaluation). Results are displayed in Table 3.

Our method surpasses other models for CPNCF task with average 7.95% and 3% on Mixture and IMCS21 datasets.

Experimental results on the IMCS21 dataset (8-shot setting) indicate that our approach exhibits a modest performance gap compared to MMoE, while preserving comparable precision levels on auxiliary tasks.

4.3 Ablation Study

We implement two distinct ablation studies to systematically evaluate our framework: **1.** elimination of intermediately complex tasks to assess their impact on model performance, and **2.** randomized task sequence permutation to examine the task ordering impact.

4.3.1 Intermediate Tasks

We analyze the impact of intermediate tasks on the proposed model, specifically including three sets of trials: deleting task of **SPRJ**, deleting task of **NER**, and deleting both of them.

It can be seen from Table 4 that the complete multi-task model beats models with missing intermediate tasks. Concretely, on the IMCS21 dataset, the model is 0.5% better than the intermediate task missing models by average, and the result attains 7.1% for mixture source datasets. Moreover, when both the intermediate tasks are removed, the performance reaches the worst among all the task-delete settings. Intuitively, both NER and SPRJ constitute essential competencies for successfully executing the CPNCF task. Consequently, these two capabilities are expected to significantly influence the final performance on CPNCF.

In general, model performance tends to scale positively with parameter capacity. Our architecture demonstrably exceeds the parameter count of the original BERT model. To validate our progressive learning paradigm, we implement a quad-FFNN cluster architecture per BERT layer - analogous to the configuration detailed in Section 3.2 - exclusively dedicated to CPNCF task acquisition.

As illustrated in Figure 3, we observe that increasing the parameter size consistently enhances model performance, as evidenced by the progression from the "single-ffnn" baseline to the "4-ffnn" configuration. More notably, our proposed learning strategy yields further performance gains beyond mere parameter scaling. This substantiates that

k	Model	SSIC	SPRJ	NER	CPNCF
Mixture Dataset					
8	Vanilla	68.2	57.4	56.7	28.1
	MT-DNN	67.3	62.1	59.6	49.2
	MMoE	55.0	54.6	46.9	56.4
	Skill	68.7	64.0	59.6	49.4
	ours	66.1	63.1	56.6	**56.9**
16	Single	72.5	56.2	57.4	54.2
	MT-DNN	72.1	52.6	60.1	53.1
	MMoE	54.8	53.3	56.1	47.6
	Skill	71.8	62.2	62.2	54.8
	ours	70.9	61.1	58.6	**57.2**
IMCS21 Dataset					
8	Vanilla	68.6	60.1	65.9	54.9
	MT-DNN	63.4	75.6	59.9	55.6
	MMoE	52.0	54.6	51.9	**66.7**
	Skill	68.4	68.7	66.7	61.0
	ours	63.6	53.1	60.5	61.6
16	Single	78.3	75.0	70.1	55.2
	MT-DNN	76.8	73.2	60.6	58.2
	MMoE	59.7	55.4	59.6	61.3
	Skill	76.8	73.5	69.7	61.9
	ours	73.9	71.8	69.8	**63.1**

Table 3: Evaluation result (%) in low resource setting. k means number of training items for each label.

Model	Mixture Dataset				IMCS21 Dataset			
	SSIC	SPRJ	NER	CPNCF	SSIC	SPRJ	NER	CPNCF
Our Model	80.5	86.3	54.5	84.7	85.1	89.3	82.4	89.2
w/o. SPRJ TASK	79.5	_	48.9	77.2	85.5	_	81.1	90.2
w/o. NER TASK	78.8	84.6	_	78.6	85.0	89.8	_	88.2
w/o. SPRJ & NER TASK	78.6	_	_	77.0	85.7	_	_	87.8

Table 4: Intermediate tasks effect (%) on two types of datasets.

Model	Mixture Dataset					IMCS21 Dataset				
	SSIC	SPRJ	NER	CPNCF	Aver.	SSIC	SPRJ	NER	CPNCF	Aver.
Our Model	80.5	86.3	54.5	84.7	76.5	85.1	89.3	82.4	89.2	86.5
1→3→2→4	80.0	85.2	63.4	79.8	77.1	84.6	73.8	88.6	89.1	84.0
3→4→1→2	34.6	50.3	70.5	88.8	61.1	84.2	49.8	80.9	85	75.0
2→1→4→3	78.4	86.2	47.3	82.1	73.5	84.3	89.3	73.5	86.1	83.3
4→3→2→1	74.7	82.2	40.9	30.1	57.0	84.0	88.2	81.7	41.3	73.8

Table 5: Order of tasks effect (%) on two types of datasets.

the performance improvement in the CPNCF task is not solely attributable to the expanded parameter space. Instead, the progressive MTL paradigm plays a pivotal role in driving these advancements, underscoring its efficacy in optimizing model performance.

4.3.2 Orders of Tasks

We randomly change the order of tasks to evaluate the effect of our progressive learning strategy. To be detailed, we design four out-of-order modes comprehensively: swapping the order of intermediate tasks ($1 \rightarrow 3 \rightarrow 2 \rightarrow 4$), exchanging the order of simple and complex tasks ($3 \rightarrow 4 \rightarrow 1 \rightarrow 2$), internal swapping for both simple and complex tasks ($2 \rightarrow 1 \rightarrow 4 \rightarrow 3$), and arranging tasks in reverse order ($4 \rightarrow 3 \rightarrow 2 \rightarrow 1$).

As evidenced by the experimental results presented in Table 5, we demonstrate that the progressive learning strategy constitutes a critical factor in enhancing MTL performance. For mixture source datasets, the model with tasks in order outperforms those out of order for 9.3% by average. The data is 7.5% on IMCS21 dataset. Notably, when the multi-task sequence is inverted, thereby disrupting the knowledge transfer, we observe a significant degradation in CPNCF performance. In fact, this configuration demonstrates inferior efficacy compared to the vanilla method, as the interference from conflicting task objectives outweighs any potential benefits of MTL.

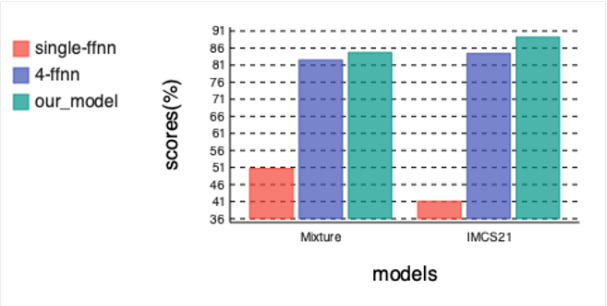

Figure 3: Ablation study on parameters size testing.

5 Conclusion

In the medical domain, CPNCF serves as a fundamental component for numerous downstream tasks, yet the paucity of annotated data consistently hinders optimal performance when employing conventional training paradigms. To address this critical limitation, we introduce an innovative yet computationally efficient learning framework that strategically leverages auxiliary simple tasks to generate rich supervisory signals for the target objective. Our methodology incorporates residual architectures to adaptively modify the feedforward neural networks within the basic Transformer structure, thereby facilitating hierarchical knowledge transfer from elementary to complex tasks. Extensive empirical evaluations conducted on comprehensive medical datasets demonstrate the superior efficacy of our progressive learning approach.

Limitations

We propose a novel multi-task learning strategy that facilitates efficient learning of complex tasks by incorporating knowledge learned from simple tasks. Throughout the framework, ranking tasks according to complexity is critical, and the two ranking principles we introduced are adapted to medical text understanding tasks. However, in different scenarios, ranking strategies may be various, and more principle support is required. Therefore, we will explore more general ranking rules in the future in order to extend our progressive learning strategy to a wider range of natural language understanding tasks.

References

Heyan Chai, Siyu Tang, Jinhao Cui, Ye Ding, Binxing Fang, and Qing Liao. 2022. Improving multi-task stance detection with multi-task interaction network. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2990–3000, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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

Wei Chen, Zhiwei Li, Hongyi Fang, Qianyuan Yao, Cheng Zhong, Jianye Hao, Qi Zhang, Xuanjing Huang, Jiajie Peng, and Zhongyu Wei. 2022. A benchmark for automatic medical consultation system: frameworks, tasks and datasets. *Bioinformatics*, 39(1). Btac817.

Verna Dankers, Marek Rei, Martha Lewis, and Ekaterina Shutova. 2019. Modelling the interplay of metaphor and emotion through multitask learning. 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 2218–2229, 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.

Kawin Ethayarajh, Yejin Choi, and Swabha Swayamdipta. 2022. Understanding dataset difficulty with $\mathcal{V}$-usable information. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 5988–6008. PMLR.

William Fedus, Barret Zoph, and Noam Shazeer. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *The Journal of Machine Learning Research*, 23(1):5232–5270.

Jason Alan Fries, Ethan H. Steinberg, Saelig Khattar, S. Fleming, José D. Posada, Alison Callahan, and Nigam Haresh Shah. 2021. Ontology-driven weak supervision for clinical entity classification in electronic health records. *Nature Communications*, 12.

Michelle Guo, Albert Haque, De-An Huang, Serena Yeung, and Li Fei-Fei. 2018. Dynamic task prioritization for multitask learning. In *Proceedings of the European conference on computer vision (ECCV)*, pages 270–287.

Kazuma Hashimoto, Caiming Xiong, Yoshimasa Tsuruoka, and Richard Socher. 2017. A joint many-task model: Growing a neural network for multiple NLP tasks. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1923–1933, Copenhagen, Denmark. Association for Computational Linguistics.

Ziming Huang, Zhuoxuan Jiang, Ke Wang, Juntao Li, Shanshan Feng, and Xian-Ling Mao. 2022. Gated mechanism enhanced multi-task learning for dialog routing. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3064–3073, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Canasai Kruengkrai, Thien Hai Nguyen, Sharifah Mahani Aljunied, and Lidong Bing. 2020. Improving low-resource named entity recognition using joint sentence and token labeling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5898–5905, Online. Association for Computational Linguistics.

Sicong Liang and Yu Zhang. 2020. A simple general approach to balance task difficulty in multi-task learning. *ArXiv*, abs/2002.04792.

Liyang Liu, Yi Li, Zhanghui Kuang, J Xue, Yimin Chen, Wenming Yang, Qingmin Liao, and Wayne Zhang. 2021. Towards impartial multi-task learning. iclr.

Lizhen Liu, Xiao Hu, Wei Song, Ruiji Fu, Ting Liu, and Guoping Hu. 2018. Neural multitask learning for simile recognition. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1543–1553, Brussels, Belgium. 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. 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.

Minh Van Nguyen, Bonan Min, Franck Dernoncourt, and Thien Nguyen. 2022. Learning cross-task dependencies for joint extraction of entities, events, event arguments, and relations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9349–9360, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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*.

Libo Qin, Wanxiang Che, Yangming Li, Haoyang Wen, and Ting Liu. 2019. A stack-propagation framework with token-level intent detection for spoken language understanding. 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 2078–2087, Hong Kong, China. Association for Computational Linguistics.

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*.

Ximeng Sun, Rameswar Panda, Rogerio Feris, and Kate Saenko. 2020. Adashare: Learning what to share for efficient deep multi-task learning. *Advances in Neural Information Processing Systems*, 33.

Henry Weld, Xiaoqi Huang, Siqu Long, Josiah Poon, and Soyeon Caren Han. 2022. A survey of joint intent detection and slot filling models in natural language understanding. *ACM Comput. Surv.*, 55(8).

Stephen Wu, Kirk Roberts, Surabhi Datta, Jingcheng Du, Zongcheng Ji, Yuqi Si, Sarvesh Soni, Qiong Wang, Qiang Wei, Yang Xiang, and 1 others. 2020. Deep learning in clinical natural language processing: a methodical review. *Journal of the American Medical Informatics Association*, 27(3):457–470.

Jun Yu, Yutong Dai, Xiaokang Liu, Jin Huang, Yishan Shen, Ke Zhang, Rong Zhou, Eashan Adhikarla, Wenxuan Ye, Yixin Liu, Zhaoming Kong, Kai Zhang, Yilong Yin, Vinod Namboodiri, Brian D. Davison, Jason H. Moore, and Yong Chen. 2024. Unleashing the power of multi-task learning: A comprehensive survey spanning traditional, deep, and pretrained foundation model eras. *Preprint*, arXiv:2404.18961.

Fan Zhang, Duyu Tang, Yong Dai, Cong Zhou, Shuangzhi Wu, and Shuming Shi. 2022a. Skillnet-nlu: A sparsely activated model for general-purpose natural language understanding. *Preprint*, arXiv:2203.03312.

Ningyu Zhang, Mosha Chen, Zhen Bi, Xiaozhuan Liang, Lei Li, Xin Shang, Kangping Yin, Chuanqi Tan, Jian Xu, Fei Huang, and 1 others. 2022b. Cblue: A chinese biomedical language understanding evaluation benchmark. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7888–7915.

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, Zihan Qiu, Huijia Wu, Zili Wang, Zhaofeng He, and Jie Fu. 2024. Hypermoe: Towards better mixture of experts via transferring among experts. *Preprint*, arXiv:2402.12656.

Baohang Zhou, Xiangrui Cai, Ying Zhang, and Xiaojie Yuan. 2021. An end-to-end progressive multi-task learning framework for medical named entity recognition and normalization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6214–6224, Online. Association for Computational Linguistics.

A Implementation details

We make implementation of the proposed model with Pytorch. The size of our model is *328,962,816* in parameters. We set the learning rate as *5e-5*, and the training batch size as *16*.