SMedBERT: A Knowledge-Enhanced Pre-trained Language Model with Structured Semantics for Medical Text Mining

Taolin Zhang[1,2,3] [*], **Zerui Cai**[4][*] **Chengyu Wang**[2], **Minghui Qiu**[2]
Bite Yang[5], **Xiaofeng He**[4,6][†]

[1] School of Software Engineering, East China Normal University [2] Alibaba Group
[3] Shanghai Key Laboratory of Trsustworthy Computing
[4] School of Computer Science and Technology, East China Normal University [5] DXY
[6] Shanghai Research Institute for Intelligent Autonomous Systems

zhangtl0519@gmail.com, zrcai_flow@126.com, yangbt@dxy.cn
{chengyu.wcy, minghui.qmh}@alibaba-inc.com, hexf@cs.ecnu.edu.cn

Abstract

Recently, the performance of Pre-trained Language Models (PLMs) has been significantly improved by injecting knowledge facts to enhance their abilities of language understanding. For medical domains, the background knowledge sources are especially useful, due to the massive medical terms and their complicated relations are difficult to understand in text. In this work, we introduce SMedBERT, a medical PLM trained on large-scale medical corpora, incorporating deep structured semantics knowledge from neighbours of linked-entity. In SMedBERT, the mention-neighbour hybrid attention is proposed to learn heterogeneous-entity information, which infuses the semantic representations of entity types into the homogeneous neighbouring entity structure. Apart from knowledge integration as external features, we propose to employ the neighbors of linked-entities in the knowledge graph as additional global contexts of text mentions, allowing them to communicate via shared neighbors, thus enrich their semantic representations. Experiments demonstrate that SMedBERT significantly outperforms strong baselines in various knowledge-intensive Chinese medical tasks. It also improves the performance of other tasks such as question answering, question matching and natural language inference.[1]

1 Introduction

Pre-trained Language Models (PLMs) learn effective context representations with self-supervised tasks, spotlighting in various NLP tasks (Wang et al., 2019a; Nan et al., 2020; Liu et al., 2020a). In addition, Knowledge-Enhanced PLMs (KEPLMs) (Zhang et al., 2019;
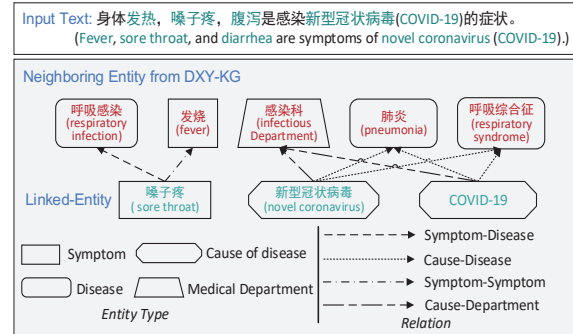

Figure 1: Example of neighboring entity information in medical text. (Best viewed in color)

Liu et al., 2020b; Wang et al., 2019b) further benefit language understanding by grounding these PLMs with high-quality, human-curated knowledge facts, which are difficult to learn from raw texts.

In the literatures, a majority of KEPLMs (Zhang et al., 2020a; Hayashi et al., 2020; Sun et al., 2020) inject information of entities corresponding to mention-spans from Knowledge Graphs (KGs) into contextual representations. However, those KEPLMs only utilize linked-entity in the KGs as auxiliary information, which pay little attention to the neighboring structured semantics information of the entity linked with text mentions. In the medical context, there exist complicated domain knowledge such as relations and medical facts among medical terms (Rotmensch et al., 2017; Li et al., 2020), which are difficult to model using previous approaches. To address this issue, we consider leveraging structured semantics knowledge in medical KGs from the two aspects. (1) Rich semantic information from neighboring structures of linked-entities, such as entity types and relations, are highly useful for medical text understanding. As in Figure 1, "新型冠状病毒" (novel coronavirus) can be the

---

[*]T. Zhang and Z. Cai contributed equally to this work.
[†] Corresponding author.
[1]The code and pre-trained models will be available at https://github.com/MatNLP/SMedBERT.

cause of many diseases, such as "肺炎" (pneumonia) and "呼吸综合征" (respiratory syndrome).[2] (2) Additionally, we leverage neighbors of linked-entity as global "contexts" to complement plain-text contexts used in (Mikolov et al., 2013a; Pennington et al., 2014). The structure knowledge contained in neighbouring entities can act as the "knowledge bridge" between mention-spans, facilitating the interaction of different mention representations. Hence, PLMs can learn better representations for rare medical terms.

In this paper, we introduce SMedBERT, a KE-PLM pre-trained over large-scale medical corpora and medical KGs. To the best of our knowledge, SMedBERT is the first PLM with structured semantics knowledge injected in the medical domain. Specifically, the contributions of SMedBERT mainly include two modules:

Mention-neighbor Hybrid Attention: We fuse the embeddings of the node and type of linked-entity neighbors into contextual target mention representations. The type-level and node-level attentions help to learn the importance of entity types and the neighbors of linked-entity, respectively, in order to reduce the knowledge noise injected into the model. The type-level attention transforms the homogeneous node-level attention into a heterogeneous learning process of neighboring entities.

Mention-neighbor Context Modeling: We propose two novel self-supervised learning tasks for promoting interaction between mention-span and corresponding global context, namely masked neighbor modeling and masked mention modeling. The former enriches the representations of "context" neighboring entities based on the well trained "target word" mention-span, while the latter focuses on gathering those information back from neighboring entities to the masked target like low-frequency mention-span which is poorly represented (Turian et al., 2010).

In the experiments, we compare SMedBERT against various strong baselines, including mainstream KEPLMs pre-trained over our medical resources. The underlying medical NLP tasks include: named entity recognition, relation extraction, question answering, question matching and natural language inference. The results show that SMedBERT consistently outperforms all the base-

lines on these tasks.

2 Related Work

PLMs in the Open Domain. PLMs have gained much attention recently, proving successful for boosting the performance of various NLP tasks (Qiu et al., 2020). Early works on PLMs focus on feature-based approaches to transform words into distributed representations (Collobert and Weston, 2008; Mikolov et al., 2013b; Pennington et al., 2014; Peters et al., 2018). BERT (Devlin et al., 2019) (as well as its robustly optimized version RoBERTa (Liu et al., 2019b)) employs bidirectional transformer encoders (Vaswani et al., 2017) and self-supervised tasks to generate context-aware token representations. Further improvement of performances mostly based on the following three types of techniques, including self-supervised tasks (Joshi et al., 2020), transformer encoder architectures (Yang et al., 2019) and multi-task learning (Liu et al., 2019a).

Knowledge-Enhanced PLMs. As existing BERT-like models only learn knowledge from plain corpora, various works have investigated how to incorporate knowledge facts to enhance the language understanding abilities of PLMs. KEPLMs are mainly divided into the following three types. (1) Knowledge-enhanced by Entity Embedding: ERNIE-THU (Zhang et al., 2019) and Know-BERT (Peters et al., 2019) inject linked-entity as heterogeneous features learned by KG embedding algorithms such as TransE (Bordes et al., 2013). (2) Knowledge-enhanced by Entity Description: E-BERT (Zhang et al., 2020a) and KE-PLER (Wang et al., 2019b) add extra description text of entities to enhance semantic representation. (3) Knowledge-enhanced by Triplet Sentence: K-BERT (Liu et al., 2020b) and CoLAKE (Sun et al., 2020) convert triplets into sentences and insert them into the training corpora without pre-trained embedding. Previous studies on KG embedding (Nguyen et al., 2016; Schlichtkrull et al., 2018) have shown that utilizing the surrounding facts of entity can obtain more informative embedding, which is the focus of our work.

PLMs in the Medical Domain. PLMs in the medical domain can be generally divided into three categories. (1) BioBERT (Lee et al., 2020), BlueBERT (Peng et al., 2019), SCIBERT (Beltagy et al., 2019) and ClinicalBert (Huang et al., 2019) apply continual learning on

---

[2] Although we focus on Chinese medical PLMs here. The proposed method can be easily adapted to other languages, which is beyond the scope of this work.
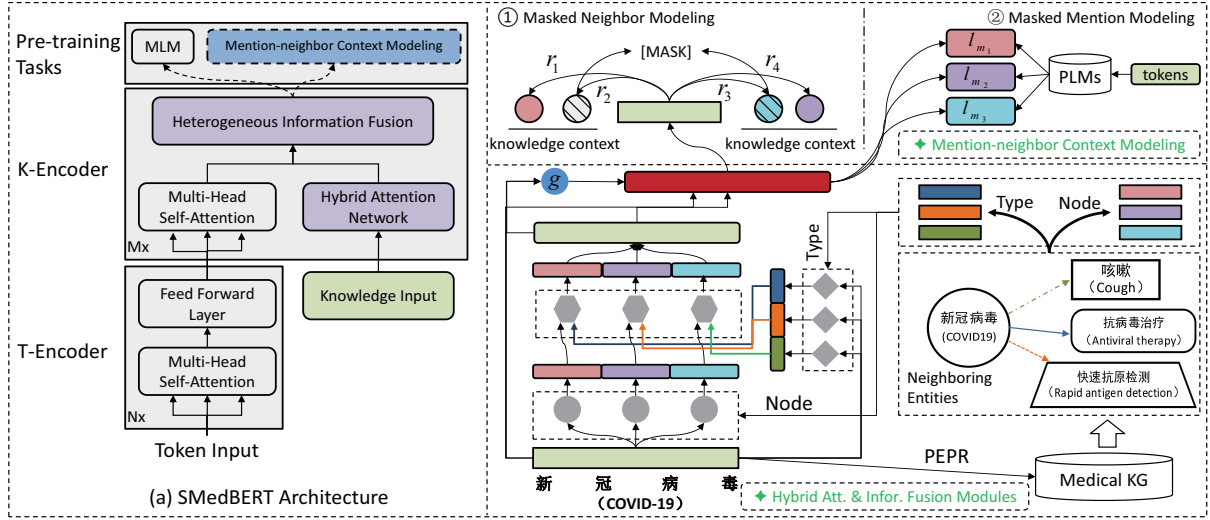
Figure 2: Model overview of SMedBERT. The left part is our model architecture and the right part is the details of our model including hybrid attention network and mention-neighbor context modeling pre-training tasks.

medical domain texts, such as PubMed abstracts, PMC full-text articles and MIMIC-III clinical notes. (2) PubMedBERT (Gu et al., 2020) learns weights from scratch using PubMed data to obtain an in-domain vocabulary, alleviating the out-of-vocabulary (OOV) problem. This training paradigm needs the support of large-scale domain data and resources. (3) Some other PLMs use domain self-supervised tasks for pre-training. For example, MC-BERT (Zhang et al., 2020b) masks Chinese medical entities and phrases to learn complex structures and concepts. DiseaseBERT (He et al., 2020) leverages the medical terms and its category as the labels to pre-train the model. In this paper, we utilize both domain corpora and neighboring entity triplets of mentions to enhance the learning of medical language representations.

3 The SMedBERT Model

3.1 Notations and Model Overview

In the PLM, we denote the hidden feature of each token $\{w_1, ..., w_N\}$ as $\{h_1, h_2, ..., h_N\}$ where N is the maximum input sequence length and the total number of pre-training samples as M. Let E be the set of mention-span e_m in the training corpora. Furthermore, the medical KG consists of the entities set $\mathcal{E}$ and the relations set $\mathcal{R}$. The triplet set is $S = \{(h, r, t) \mid h \in \mathcal{E}, r \in \mathcal{R}, t \in \mathcal{E}\}$, where h is the head entity with relation r to the tail entity t. The embeddings of entities and relations trained on KG by TransR (Lin et al., 2015) are represented as Γ_{ent} and Γ_{rel}, respectively. The

neighboring entity set recalled from KG by e_m is denoted as $\mathcal{N}_{e_m} = \{e_m^1, e_m^2, ..., e_m^K\}$ where K is the threshold of our PEPR algorithm. We denote the number of entities in the KG as Z. The dimensions of the hidden representation in PLM and the KG embeddings are d_1 and d_2, respectively.

The main architecture of the our model is shown in Figure 2. SMedBERT mainly includes three components: (1) Top-K entity sorting determine which K neighbour entities to use for each mention. (2) Mention-neighbor hybrid attention aims to infuse the structured semantics knowledge into encoder layers, which includes type attention, node attention and gated position infusion module. (3) Mention-neighbor context modeling includes masked neighbor modeling and masked mention modeling aims to promote mentions to leverage and interact with neighbour entities.

3.2 Top-K Entity Sorting

Previous research shows that simple neighboring entity expansion may induce knowledge noises during PLM training (Wang et al., 2019a). In order to recall the most important neighboring entity set from the KG for each mention, we extend the Personalized PageRank (PPR) (Page et al., 1999) algorithm to filter out trivial entities. [3] Recall that the iterative process in PPR is $V_i = (1 - \alpha)A \cdot V_{i-1} + \alpha P$ where A is the normalized adjacency matrix, α is the damping factor, P is uniformly distributed jump probability vector, and V is the

---

[3]We name our algorithm to be Personalized Entity PageRank, abbreviated as PEPR.

iterative score vector for each entity.

PEPR specifically focuses on learning the weight for the target mention span in each iteration. It assigns the span e_m a higher jump probability 1 in P with the remaining as $\frac{1}{Z}$. It also uses the entity frequency to initialize the score vector V:

$$V_{e_m} = \begin{cases} \frac{t_{e_m}}{T} & e_m \in E \\ \frac{1}{M} & e_m \notin E \end{cases} \quad (1)$$

where T is the sum of frequencies of all entities. t_{e_m} is the frequency of e_m in the corpora. After sorting, we select the top-K entity set $\mathcal{N}_{e_m}$.

3.3 Mention-neighbor Hybrid Attention

Besides the embeddings of neighboring entities, SMedBERT integrates the type information of medical entities to further enhance semantic representations of mention-span.

3.3.1 Neighboring Entity Type Attention

Different types of neighboring entities may have different impacts. Given a specific mention-span e_m, we compute the neighboring entity type attention. Concretely, we calculate hidden representation of each entity type τ as $h_\tau = \sum_{e_m^i \in E_m^\tau} h_{e_m^i}$. E_m^τ are neighboring entities of e_m with the same type τ and $h_{e_m^i} = \Gamma_{ent}\left(e_m^i\right) \in \mathbb{R}^{d_2}$.

$$h'_{e_m} = \mathcal{LN}\left(\sigma\left(f_{sp}\left(h_i, \ldots, h_j\right) W_{be}\right)\right) \quad (2)$$

where f_{sp} is the self-attentive pooling (Lin et al., 2017) to generate the mention-span representation $h_{e_m} \in \mathbb{R}^{d_1}$ and the $(h_i, h_{i+1}, \ldots, h_j)$ is the hidden representation of tokens $(w_i, w_{i+1}, \ldots, w_j)$ in mention-span e_m trained by PLMs. $h'_{e_m} \in \mathbb{R}^{d_2}$ is obtained by $\sigma(\cdot)$ non-linear activation function GELU (Hendrycks and Gimpel, 2016) and the learnable projection matrix $W_{be} \in \mathbb{R}^{d_1 \times d_2}$. $\mathcal{LN}$ is the LayerNorm function (Ba et al., 2016). Then, we calculate the each type attention weight using the type representation $h_\tau \in \mathbb{R}^{d_2}$ and the transformed mention-span representation h'_{e_m}:

$$\alpha'_\tau = \tanh\left(h'_{e_m} W_t + h_\tau W_{t'}\right) W_a \quad (3)$$

where $W_t \in \mathbb{R}^{d_2 \times d_2}$, $W_{t'} \in \mathbb{R}^{d_2 \times d_2}$ and $W_a \in \mathbb{R}^{d_2 \times 1}$. Finally, the neighboring entity type attention weights α_τ are obtained by normalizing the attention score α'_τ among all entity types $\mathcal{T}$.

3.3.2 Neighboring Entity Node Attention

Apart from entity type information, different neighboring entities also have different influences.

Specifically, we devise the neighboring entity node attention to capture the different semantic influences from neighboring entities to the target mention span and reduce the effect of noises. We calculate the entity node attention using the mention-span representation h'_{e_m} and neighboring entities representation $h_{e_m^i}$ with entity type τ as:

$$\beta'_{e_m e_m^i} = \frac{\left(h'_{e_m} W_q\right)\left(h_{e_m^i} W_k\right)^T}{\sqrt{d_2}} \alpha_\tau \quad (4)$$

$$\beta_{e_m e_m^i} = \frac{\exp\left(\beta'_{e_m e_m^i}\right)}{\sum_{e_m^i \in \mathcal{N}_{e_m}} \exp\left(\beta'_{e_m e_m^i}\right)} \quad (5)$$

where $W_q \in \mathbb{R}^{d_2 \times d_2}$ and $W_k \in \mathbb{R}^{d_2 \times d_2}$ are the attention weight matrices.

The representations of all neighboring entities in $\mathcal{N}_{e_m}$ are aggregated to $\bar{h}'_{e_m} \in \mathbb{R}^{d_2}$:

$$\widehat{h}'_{e_m} = \sum_{e_m^i \in \mathcal{N}_{e_m}} \beta_{e_m e_m^i}\left(h_{e_m^i} W_v + b_v\right) \quad (6)$$

$$\bar{h}'_{e_m} = \mathcal{LN}\left(\widehat{h}'_{e_m} + \left(\sigma\left(\widehat{h}'_{e_m} W_{l1} + b_{l1}\right) W_{l2}\right)\right) \quad (7)$$

where $W_v \in \mathbb{R}^{d_2 \times d_2}$, $W_{l1} \in \mathbb{R}^{d_2 \times 4d_2}$, $W_{l2} \in \mathbb{R}^{4d_2 \times d_2}$. $b_v \in \mathbb{R}^{d_2}$ and $b_{l1} \in \mathbb{R}^{4d_2}$ are the bias vectors. $\bar{h}'_{e_m}$ is the mention-neighbor representation from hybrid attention module.

3.3.3 Gated Position Infusion

Knowledge-injected representations may divert the texts from its original meanings. We further reduce knowledge noises via gated position infusion:

$$h'_{e_{mf}} = \sigma\left(\left[\bar{h}'_{e_m} \| h'_{e_m}\right] W_{mf} + b_{mf}\right) \quad (8)$$

$$\widetilde{h}'_{e_{mf}} = \mathcal{LN}(h'_{e_{mf}} W_{bp} + b_{bp}) \quad (9)$$

where $W_{mf} \in \mathbb{R}^{2d_2 \times 2d_2}$, $W_{bp} \in \mathbb{R}^{2d_2 \times d_1}$, $b_{mf} \in \mathbb{R}^{2d_2}$, $b_{bp} \in \mathbb{R}^{d_1}$. $h'_{e_{mf}} \in \mathbb{R}^{2d_2}$ is the span-level infusion representation. "$\|$" means concatenation operation. $\widetilde{h}'_{e_{mf}} \in \mathbb{R}^{d_1}$ is the final knowledge-injected representation for mention e_m. We generate the output token representation h_{if} by [4]:

$$g_i = \tanh\left(\left(\left[h_i \| \widetilde{h}'_{e_{mf}}\right]\right) W_{ug} + b_{ug}\right) \quad (10)$$

$$h_{if} = \sigma\left(\left(\left[h_i \| g_i * \widetilde{h}'_{e_{mf}}\right]\right) W_{ex} + b_{ex}\right) + h_i \quad (11)$$

where $W_{ug}, W_{ex} \in \mathbb{R}^{2d_1 \times d_1}$. $b_{ug}, b_{ex} \in \mathbb{R}^{d_1}$. "$*$" means element-wise multiplication.

---

[4]We find that restricting the knowledge infusion position to tokens is helpful to improve performance.

3.4 Mention-neighbor Context Modeling

To fully exploit the structured semantics knowledge in KG, we further introduce two novel self-supervised pre-training tasks, namely Masked Neighbor Modeling (MNeM) and Masked Mention Modeling (MMeM).

3.4.1 Masked Neighbor Modeling

Formally, let r be the relation between the mention-span e_m and a neighboring entity e_m^i:

$$h_{mf} = \mathcal{LN}\left(\sigma\left(f_{sp}\left(h_{if}, \ldots, h_{jf}\right) W_{sa}\right)\right) \quad (12)$$

where h_{mf} is the mention-span hidden features based on the tokens hidden representation $\left(h_{if}, h_{(i+1)f}, \ldots, h_{jf}\right)$. $h_r = \Gamma_{rel}\left(r\right) \in \mathbb{R}^{d_2}$ is the relation r representation and $W_{sa} \in \mathbb{R}^{d_1 \times d_2}$ is a learnable projection matrix. The goal of MNeM is leveraging the structured semantics in surrounding entities while reserving the knowledge of relations between entities. Considering the object functions of skip-gram with negative sampling (SGNS) (Mikolov et al., 2013a) and score function of TransR (Lin et al., 2015):

$$\mathcal{L}_S = \log f_s(w, c) + k \cdot \mathbb{E}_{c_n \sim P_D}[\log f_s(w, -c_n)] \quad (13)$$

$$f_{tr}(h, r, t) = \| hM_r + r - tM_r \| \quad (14)$$

where the w in $\mathcal{L}_S$ is the target word of context c. f_s is the compatibility function measuring how well the target word is fitted into the context. Inspired by SGNS, following the general energy-based framework (LeCun et al., 2006), we treat mention-spans in corpora as "target words", and neighbors of corresponding entities in KG as "contexts" to provide additional global contexts. We employ the Sampled-Softmax (Jean et al., 2015) as the criterion $\mathcal{L}_{MNeM}$ for the mention-span e_m:

$$\sum_{\mathcal{N}_{e_m}} \log \frac{\exp(f_s(\theta))}{\exp(f_s(\theta)) + K \cdot \mathbb{E}_{e_n \sim Q(e_n)}[\exp(f_s(\theta'))]} \quad (15)$$

where θ denotes the triplet (e_m, r, e_m^i), $e_m^i \in \mathcal{N}_{e_m}$. θ' is the negative triplets (e_m, r, e_n), and e_n is negative entity sampled with $Q(e_m^i)$ detailed in Appendix B. To keep the knowledge of relations between entities, we define the compatibility function as:

$$f_s\left(e_m, r, e_m^i\right) = \frac{h_{mf}M_r + h_r}{||h_{mf}M_r + h_r||} \cdot \frac{(h_{e_m^i}M_r)^T}{||h_{e_m^i}M_r||}\mu \quad (16)$$

where μ is a scale factor. Assuming the norms of both $h_{mf}M_r + h_r$ and $h_{e_m^i}M_r$ are 1, we have:

$$f_s\left(e_m, r, e_m^i\right) = \mu \iff f_{tr}(h_{mf}, h_r, h_{e_m^i}) = 0 \quad (17)$$

which indicates the proposed f_s is equivalence with f_{tr}. Because $| h_{e_n} M_r |$ needs to be calculated for each e_n, the computation of the score function f_s is costly. Hence, we transform part of the formula f_s as follows:

$$(h_{mf}M_r + h_r) \cdot (h_{e_n}M_r)^T =$$
$$\left[\begin{array}{cc} h_{mf} & 1 \end{array}\right] \left[\begin{array}{c} M_r \\ h_r \end{array}\right] \left[\begin{array}{c} M_r \\ h_r \end{array}\right]^T \left[\begin{array}{cc} h_{e_n} & 0 \end{array}\right]^T$$
$$= \left[\begin{array}{cc} h_{mf} & 1 \end{array}\right] M_{P_r} \left[\begin{array}{cc} h_{e_n} & 0 \end{array}\right]^T \quad (18)$$

In this way, we eliminate computation of transforming each h_{e_n}. Finally, to compensate the offset introduced by the negative sampling function $Q(e_m^i)$ (Jean et al., 2015), we complement $f_s(e_m, r, e_m^i)$ as:

$$\frac{\left[\begin{array}{cc} h_{mf} & 1 \end{array}\right] M_{P_r}}{\| \left[\begin{array}{cc} h_{mf} & 1 \end{array}\right] M_{P_r} \|} \cdot \frac{\left[\begin{array}{cc} h_{e_m^i} & 0 \end{array}\right]}{\| h_{e_m^i} \|}\mu - \mu \log Q(e_m^i) \quad (19)$$

3.4.2 Masked Mention Modeling

In contrast to MNeM, MMeM transfers the semantic information in neighboring entities back to the masked mention e_m.

$$\mathcal{Y}_m = \mathcal{LN}\left(\sigma\left(f_{sp}\left(h_{ip}, \ldots, h_{jp}\right) W_{sa}\right)\right) \quad (20)$$

where $\mathcal{Y}_m$ is the ground-truth representation of e_m and $h_{ip} = \Gamma_p(w_i) \in \mathbb{R}^{d_2}$. Γ_p is the pre-trained embedding of BERT in our medical corpora. The mention-span representation obtained by our model is h_{mf}. For a sample s, the loss of MMeM $\mathcal{L}_{MMeM}$ is calculated via Mean-Squared Error:

$$\mathcal{L}_{MMeM} = \sum_{m_i}^{\mathcal{M}_s} \| h_{m_i f} - \mathcal{Y}_{m_i} \|^2 \quad (21)$$

where $\mathcal{M}_s$ is the set of mentions of sample s.

3.5 Training Objective

In SMedBERT, the training objectives mainly consist of three parts, including the self-supervised loss proposed in previous works and the mention-neighbor context modeling loss proposed in our work. Our model can be applied to medical text pre-training directly in different languages as long

as high-quality medical KGs can be obtained. The total loss is as follows:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{EX}} + \lambda_1 \mathcal{L}_{\text{MNeM}} + \lambda_2 \mathcal{L}_{\text{MMeM}} \quad (22)$$

where $\mathcal{L}_{\text{EX}}$ is the sum of sentence-order prediction (SOP) (Lan et al., 2020) and masked language modeling. λ_1 and λ_2 are the hyperparameters.

4 Experiments

4.1 Data Source

Pre-training Data. The pre-training corpora after pre-processing contains 5,937,695 text segments with 3,028,224,412 tokens (4.9 GB). The KGs embedding trained by TransR (Lin et al., 2015) on two trusted data sources, including the Symptom-In-Chinese from OpenKG[5] and DXY-KG [6] containing 139,572 and 152,508 entities, respectively. The number of triplets in the two KGs are 1,007,818 and 3,764,711. The pre-training corpora and the KGs are further described in Appendix A.1.

Task Data. We use four large-scale datasets in ChineseBLUE (Zhang et al., 2020b) to evaluate our model, which are benchmark of Chinese medical NLP tasks. Additionally, we test models on four datasets from real application scenarios provided by DXY company [7] and CHIP [8], i.e., Named Entity Recognition (DXY-NER), Relation Extraction (DXY-RE, CHIP-RE) and Question Answer (WebMedQA (He et al., 2019)). For other information of the downstream datasets, we refer readers to Appendix A.2.

4.2 Baselines

In this work, we compare SMedBERT with general PLMs, domain-specific PLMs and KEPLMs with knowledge embedding injected, pre-trained on our Chinese medical corpora:

General PLMs: We use three Chinese BERT-style models, namely BERT-base (Devlin et al., 2019), BERT-wwm (Cui et al., 2019) and RoBERTa (Liu et al., 2019b). All the weights are initialized from (Cui et al., 2020).

Domain-specific PLMs: As very few PLMs in the Chinese medical domain are available, we consider the following models. MC-BERT (Zhang et al., 2020b) is pre-trained over a Chinese

Model	D1	D2	D3
SGNS-char-med	27.21%	27.16%	21.72%
SGNS-word-med	24.64%	24.95%	20.37%
GLOVE-char-med	27.24%	27.12%	21.91%
GLOVE-word-med	24.41%	23.89%	20.56%
BERT-open	29.79%	29.41%	21.83%
BERT-wwm-open	29.75%	29.55%	21.97%
RoBERTa-open	30.84%	30.56%	21.98%
MC-BERT	30.63%	30.34%	22.65%
BioBERT-zh	30.84%	30.69%	22.71%
ERNIE-med	30.97%	30.78%	22.99%
KnowBERT-med	30.95%	30.77%	23.07%
SMedBERT	**31.81%**	**32.14%**	**24.08%**

Table 1: Results of unsupervised semantic similarity task. "med" refers to models continually pre-trained on medical corpora, and "open" means open-domain corpora. "char' and "word" refer to the token granularity of input samples.

medical corpora via masking different granularity tokens. We also pre-train BERT using our corpora, denoted as BioBERT-zh.

KEPLMs: We employ two SOTA KEPLMs continually pre-trained on our medical corpora as our baseline models, including ERNIE-THU (Zhang et al., 2019) and KnowBERT (Peters et al., 2019). For a fair comparison, KEPLMs use other additional resources rather than the KG embedding are excluded (See Section 2), and all the baseline KEPLMs are injected by the same KG embedding.

The detailed parameter settings and training procedure are in Appendix B.

4.3 Intrinsic Evaluation

To evaluate the semantic representation ability of SMedBERT, we design an unsupervised semantic similarity task. Specifically, we extract all entities pairs with equivalence relations in KGs as positive pairs. For each positive pair, we use one of the entity as query entity while the other as positive candidate, which is used to sample other entities as negative candidates. We denote this dataset as **D1**. Besides, the entities in the same positive pair often have many neighbours in common. We select positive pairs with large proportions of common neighbours as **D2**. Additionally, to verify the ability of SMedBERT of enhancing the low-frequency mention representation, we extract all positive pairs that with at least one low-frequency mention as **D3**. There are totally 359,358, 272,320 and 41,583 samples for **D1, D2,**

Model	Named Entity Recognition					Relation Extraction			
	cMedQANER		DXY-NER		Average	CHIP-RE	DXY-RE		Average
	Dev	Test	Dev	Test	Test	Test	Dev	Test	Test
BERT-open	80.69%	83.12%	79.12%	79.03%	81.08%	85.86%	94.18%	94.13%	90.00%
BERT-wwm-open	80.52%	83.07%	79.48%	79.29%	81.18%	86.01%	94.35%	94.38%	90.20%
RoBERT-open	80.92%	83.29%	79.27%	79.33%	81.31%	86.19%	94.64%	94.66%	90.43%
BioBERT-zh	80.72%	83.38%	79.52%	79.45%	81.42%	86.12%	94.54%	94.64%	90.38%
MC-BERT	81.02%	83.46%	79.79%	79.59%	81.53%	86.09%	94.74%	94.73%	90.41%
KnowBERT-med	81.29%	83.75%	80.86%	80.44%	82.10%	86.27%	95.05%	94.97%	90.62%
ERNIE-med	81.22%	83.87%	80.82%	80.87%	82.37%	86.25%	94.98%	94.91%	90.58%
SMedBERT	**82.23%**	**84.75%**	**83.06%**	**82.94%**	**83.85%**	**86.95%**	**95.73%**	**95.89%**	**91.42%**

Table 2: Performance of Named Entity Recognition (NER) and Relation Extraction (RE) tasks in terms of F1. The Development data of CHIP-RE is unreleased in public dataset.

Model	Question Answering					Question Matching		Natural Lang. Infer.	
	cMedQA		WebMedQA		Average	cMedQQ		cMedNLI	
	Dev	Test	Dev	Test	Test	Dev	Test	Dev	Test
BERT-open	72.99%	73.82%	77.20%	79.72%	76.77%	86.74%	86.72%	95.52%	95.66%
BERT-wwm-open	72.03%	72.96%	77.06%	79.68%	76.32%	86.98%	86.82%	95.53%	95.78%
RoBERT-open	72.22%	73.18%	77.18%	79.57%	76.38%	87.24%	86.97%	95.87%	96.11%
BioBERT-zh	74.32%	75.12%	78.04%	80.45%	77.79%	87.30%	87.06%	95.89%	96.04%
MC-BERT	74.40%	74.46%	77.85%	80.54%	77.50%	87.17%	87.01%	95.81%	96.06%
KnowBERT-med	74.38%	75.25%	78.20%	80.67%	77.96%	87.25%	87.14%	95.96%	96.03%
ERNIE-med	74.37%	75.22%	77.93%	80.56%	77.89%	87.34%	87.20%	96.02%	96.25%
SMedBERT	**75.06%**	**76.04%**	**79.26%**	**81.68%**	**78.86%**	**88.13%**	**88.09%**	**96.64%**	**96.88%**

Table 3: Performance of Question Answering (QA), Question Matching (QM) and Natural Language Inference (NLI) tasks. The metric of the QA task is Acc@1 and those of QM and NLI are F1.

D3 respectively. We describe the details of collecting data and embedding words in Appendix C. In this experiments, we compare SMedBERT with three types of models: classical word embedding methods (**SGNS** (Mikolov et al., 2013a), **GLOVE** (Pennington et al., 2014)), PLMs and KEPLMs. We compute the similarity between the representation of query entities and all the other entities, retrieving the most similar one. The evaluation metric is top-1 accuracy (Acc@1).

Experiment results are shown in Table 1. From the results, we observe that: (1) SMedBERT greatly outperforms all baselines especially on the dataset **D2 (+1.36%)**, where most positive pairs have many shared neighbours, demonstrating that ability of SMedBERT to utilize semantic information from the global context. (2) In dataset **D3**, SMedBERT improve the performance significantly **(+1.01%)**, indicating our model is effective to enhance the representation of low-frequency mentions.

4.4 Results of Downstream Tasks

We first evaluate our model in NER and RE tasks that are closely related to entities in the input texts. Table 2 shows the performances on medical NER and RE tasks. In NER and RE tasks, we can observe from the results: (1) Compared with PLMs trained in open-domain corpora, KEPLMs with medical corpora and knowledge facts achieve better results. (2) The performance of SMedBERT is greatly improved compared with the strongest baseline in two NER datasets **(+0.88%, +2.07%)**, and **(+0.68%, +0.92%)** on RE tasks. We also evaluate SMedBERT on QA, QM and NLI tasks and the performance is shown in Table 3. We can observe that SMedBERT improve the performance consistently on these datasets **(+0.90% on QA, +0.89% on QM and +0.63% on NLI)**. In general, it can be seen from Table 2 and Table 3 that injecting the domain knowledge especially the structured semantics knowledge can improve the result greatly.
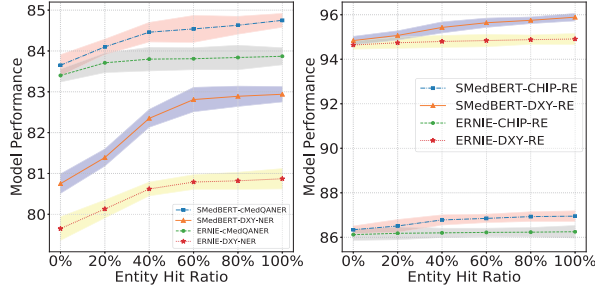
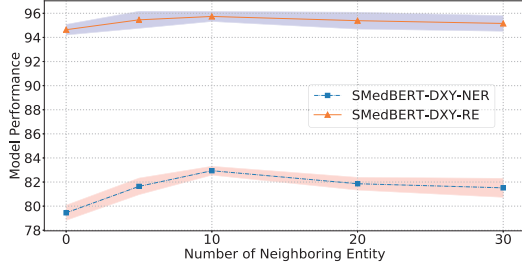Figure 3: Entity hit ratio results of SMedBERT and ERNIE in NER and RE tasks.

Figure 4: The influence of different K values in results.

4.5 Influence of Entity Hit Ratio

In this experiment, we explore the model performance in NER and RE tasks with different entity hit ratios, which control the proportions of knowledge-enhanced mention-spans in the samples. The average number of mention-spans in samples is about 40. Figure 3 illustrates the performance of SMedBERT and ERNIE-med (Zhang et al., 2019). From the result, we can observe that: (1) The performance improves significantly at the beginning and then keeps stable as the hit ratio increases, proving the heterogeneous knowledge is beneficial to improve the ability of language understanding and indicating too much knowledge facts are unhelpful to further improve model performance due to the knowledge noise (Liu et al., 2020b). (2) Compared with previous approaches, our SMedBERT model improves performance greatly and more stable.

4.6 Influence of Neighboring Entity Number

We further evaluate the model performance under different K over the test set of DXY-NER and DXY-RE. Figure 4 shows the the model result with $K = \{5, 10, 20, 30\}$. In our settings, the SMedBERT can achieve the best performance in different tasks around $K = 10$. The results of SMedBERT show that the model performance increasing first and then decreasing with the increasing

Model	D5	D6	D7	D8
SMedBERT	**84.75%**	**82.94%**	**86.95%**	**95.89%**
ERNIE-med	83.87%	80.87%	86.25%	94.91%
- Type Att.	84.25%	81.99%	86.61%	95.29%
- Hybrid Att.	83.71%	80.85%	86.46%	95.20%
- Know. Loss	84.31%	82.12%	86.50%	95.43%

Table 4: Ablation study of SMedBERT on four datasets (testing set). Due to the space limitation, we use the abbreviations "D5", "D6", "D7", and "D8" to represent the cMedQANER, DXY-NER, CHIP-RE, and DXY-RE datasets respectively.

of K. This phenomenon also indicates the knowledge noise problem that injecting too much knowledge of neighboring entities may hurt the performance.

4.7 Ablation Study

In Table 4, we choose three important model components for our ablation study and report the test set performance on four datasets of NER and RE tasks that are closely related to entities. Specifically, the three model components are neighboring entity type attention, the whole hybrid attention module, and mention-neighbor context modeling respectively, which includes two masked language model loss $\mathcal{L}_{\text{MNeM}}$ and $\mathcal{L}_{\text{MMeM}}$.

From the result, we can observe that: (1) Without any of the three mechanisms, our model performance can also perform competitively with the strong baseline ERNIE-med (Zhang et al., 2019). (2) Note that after removing the hybrid attention module, the performance of our model has the greatest decline, which indicates that injecting rich heterogeneous knowledge of neighboring entities is effective.

5 Conclusion

In this work, we address medical text mining tasks with the structured semantics KEPLM proposed named SMedBERT. Accordingly, we inject entity type semantic information of neighboring entities into node attention mechanism via heterogeneous feature learning process. Moreover, we treat the neighboring entity structures as additional global contexts to predict the masked candidate entities based on mention-spans and vice versa. The experimental results show the significant improvement of our model on various medical NLP tasks and the intrinsic evaluation. There are two research directions that can be further explored: (1) Injecting

deeper knowledge by using "farther neighboring" entities as contexts; (2) Further enhancing Chinese medical long-tail entity semantic representation.

Acknowledgements

References

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. In *EMNLP*, pages 3613–3618.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *NIPS*, pages 2787–2795.

Ronan Collobert and Jason Weston. 2008. A unified architecture for natural language processing: deep neural networks with multitask learning. In *ICML*, pages 160–167.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for chinese natural language processing. In *EMNLP*, pages 657–668.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for chinese BERT. *CoRR*, abs/1906.08101.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186.

Yu Gu, Robert Tinn, Hao Cheng, Michael Lucas, Naoto Usuyama, Xiaodong Liu, Tristan Naumann, Jianfeng Gao, and Hoifung Poon. 2020. Domain-specific language model pretraining for biomedical natural language processing. *CoRR*, abs/2007.15779.

Hiroaki Hayashi, Zecong Hu, Chenyan Xiong, and Graham Neubig. 2020. Latent relation language models. In *AAAI*, pages 7911–7918.

Junqing He, Mingming Fu, and Manshu Tu. 2019. Applying deep matching networks to chinese medical question answering: a study and a dataset. *BMC Medical Informatics Decis. Mak.*, 19-S(2):91–100.

Yun He, Ziwei Zhu, Yin Zhang, Qin Chen, and James Caverlee. 2020. Infusing disease knowledge into BERT for health question answering. In *EMNLP*, pages 4604–4614.

Dan Hendrycks and Kevin Gimpel. 2016. Gaussian error linear units (gelus). *arXiv:1606.08415*.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *CoRR*, abs/1904.05342.

Paul Jaccard. 1912. The distribution of the flora in the alpine zone. *New Phydvtologist*, 11(2):37–50.

Sébastien Jean, KyungHyun Cho, Roland Memisevic, and Yoshua Bengio. 2015. On using very large target vocabulary for neural machine translation. In *ACL*, pages 1–10.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Trans. Assoc. Comput. Linguistics*, 8:64–77.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *ICLR*.

Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, and F Huang. 2006. A tutorial on energy-based learning. *Predicting structured data*, 1(0).

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinform.*, 36(4):1234–1240.

Linfeng Li, Peng Wang, Jun Yan, Yao Wang, Simin Li, Jinpeng Jiang, Zhe Sun, Buzhou Tang, Tsung-Hui Chang, Shenghui Wang, and Yuting Liu. 2020. Real-world data medical knowledge graph: construction and applications. *Artif. Intell. Medicine*, 103:101817.

Yankai Lin, Zhiyuan Liu, Maosong Sun, Yang Liu, and Xuan Zhu. 2015. Learning entity and relation embeddings for knowledge graph completion. In *AAAI*, pages 2181–2187.

Zhouhan Lin, Minwei Feng, Cícero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *ICLR*.

Dayiheng Liu, Yeyun Gong, Jie Fu, Yu Yan, Jiusheng Chen, Daxin Jiang, Jiancheng Lv, and Nan Duan. 2020a. Rikinet: Reading wikipedia pages for natural question answering. In *ACL*, pages 6762–6771.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020b. K-BERT: enabling language representation with knowledge graph. In *AAAI*, pages 2901–2908.

Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019a. Multi-task deep neural networks for natural language understanding. In *ACL*, pages 4487–4496.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013a. Efficient estimation of word representations in vector space. In *ICLR*.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013b. Distributed representations of words and phrases and their compositionality. In *NIPS*, pages 3111–3119.

Guoshun Nan, Zhijiang Guo, Ivan Sekulic, and Wei Lu. 2020. Reasoning with latent structure refinement for document-level. In *ACL*, pages 1546–1557.

Dat Quoc Nguyen, Kairit Sirts, Lizhen Qu, and Mark Johnson. 2016. Neighborhood mixture model for knowledge base completion. In *CoNLL*, pages 40–50.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical Report 1999-66, Stanford InfoLab.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing. In *BioNLP*, pages 58–65.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

Matthew E. Peters, Mark Neumann, Robert L. Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *EMNLP*, pages 43–54.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL*, pages 2227–2237.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *CoRR*, abs/2003.08271.

Maya Rotmensch, Yoni Halpern, Abdulhakim Tlimat, Steven Horng, and David Sontag. 2017. Learning a health knowledge graph from electronic medical records. *Scientific reports*, 7(1):1–11.

Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *ESWC*, pages 593–607.

Tianxiang Sun, Yunfan Shao, Xipeng Qiu, Qipeng Guo, Yaru Hu, Xuanjing Huang, and Zheng Zhang. 2020. Colake: Contextualized language and knowledge embedding. In *COLING*, pages 3660–3670.

Joseph P. Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised. In *ACL*, pages 384–394.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NIPS*, pages 5998–6008.

Xiaoyan Wang, Pavan Kapanipathi, Ryan Musa, Mo Yu, Kartik Talamadupula, Ibrahim Abdelaziz, Maria Chang, Achille Fokoue, Bassem Makni, Nicholas Mattei, and Michael Witbrock. 2019a. Improving natural language inference using external knowledge in the. In *AAAI*, pages 7208–7215.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2019b. KEPLER: A unified model for knowledge embedding and pre-trained. *CoRR*, abs/1911.06136.

William E Winkler. 1990. String comparator metrics and enhanced decision rules in the fellegi-s.

Liang Xu, Xuanwei Zhang, and Qianqian Dong. 2020. Cluecorpus2020: A large-scale chinese corpus for pre-training langua. *CoRR*, abs/2003.01355.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understand. In *NIPS*, pages 5754–5764.

Denghui Zhang, Zixuan Yuan, Yanchi Liu, Zuohui Fu, Fuzhen Zhuang, Pengyang Wang, Haifeng Chen, and Hui Xiong. 2020a. E-BERT: A phrase and product knowledge enhanced language model. *CoRR*, abs/2009.02835.

Ningyu Zhang, Qianghuai Jia, Kangping Yin, Liang Dong, Feng Gao, and Nengwei Hua. 2020b. Conceptualized representation learning for chinese biomedical text m. *CoRR*, abs/2008.10813.

Sheng Zhang, Xin Zhang, Hui Wang, Jiajun Cheng, Pei Li, and Zhaoyun Ding. 2017. Chinese medical question answer matching using end-to-end characte. *Applied Sciences*, 7(8):767.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: enhanced language representation with informative entities. In *ACL*, pages 1441–1451.

A Data Source

A.1 Pre-training Data

A.1.1 Training Corpora

The pre-training corpora is crawled from DXY BBS (Bulletin Board System) [9], which is a very popular Chinese social network for doctors, medical institutions, life scientists, and medical practitioners. The BBS has more than 30 channels, which contains 18 forums and 130 fine-grained groups, covering most of the medical domains. For our pre-training purpose, we crawl texts from channels about clinical medicine, pharmacology, public health and consulting. For text pre-processing, we mainly follow the methods of (Xu et al., 2020). Additionally, (1) we remove all URLs, HTML tags, e-mail addresses, and all tokens except characters, digits, and punctuation (2) all documents shorter than 256 are discard, while documents longer than 512 are cut into shorter text segments.

A.1.2 Knowledge Graph

The DXY knowledge graph is construed by extracting structured text from DXY website[10], which includes information of diseases, drugs and hospitals edited by certified medical experts, thus the quality of the KG is guaranteed. The KG is mainly disease-centered, including totally 3,764,711 triples, 152.508 unique entities, and 44 types of relations. The details of Symptom-In-Chinese from OpenKG is available [11]. We finally get 26 types of entities, 274,163 unique entities, 56 types of relations, and 4,390,726 triples after the fusion of the two KGs.

A.2 Task Data

We choose the four large-scale datasets in ChineseBlue tasks (Zhang et al., 2020b) while others are ignored due to the limitation of datasets size, which are cMedQANER, cMedQQ, cMedQNLI and cMedQA. WebMedQA (He et al., 2019) is a real-world Chinese medical question answering dataset and CHIP-RE dataset are collected from online health consultancy websites. Note that since both the WebMedQA and cMedQA datasets are very large while we have many baselines to be compared, we randomly sample the official training set, development set and test set respectively

[9] https://www.dxy.cn/bbs/newweb/pc/home
[10] https://portal.dxy.cn/
[11] http://openkg.cn/dataset/symptom-in-chinese

to form their corresponding smaller version for experiments. DXY-NER and DXY-RE are datasets from real medical application scenarios provided by a prestigious Chinese medical company. The DXY-NER contains 22 unique entity types and 56 relation types in the DXY-RE. These two datasets are collected from the medical forum of DXY and books in the medical domain. Annotators are selected from junior and senior students with clinical medical background. In the process of quality control, the two datasets are annotated twice by different groups of annotators. An expert with medical background performs quality check manually again when annotated results are inconsistent, whereas perform sampling quality check when results are consistent. Table 5 shows the datasets size of our experiments.

B Model Settings and Training Details

Hyper-parameters. d_1=768, d_2=200, K=10, μ=10, λ_1=2, λ_2=4.

Model Details. We align the all mention-spans to the entity in KG by exact match for comparison purpose with ENIRE-THU (Zhang et al., 2019). The negative sampling function is defined as $Q(e_m^i) = \frac{t_{e_m^i}}{C_{e_m^i}}$, where $C_{e_m^i}$ is the sum of frequency of all mentions with the same type of e_m^i. The Mention-neighbor Hybrid Attention module is inserted after the tenth transformer encoder layer to compare with KnowBERT (Peters et al., 2019), while we perform the Mention-neighbor Context Modeling based on the output of BERT encoder. We use all the base-version PLMs in the experiments. The size of SMedBERT is 474MB while 393MB of that are components of BERT, and the added 81MB is mostly of the KG embedding. Results are presented in average with 5 random runs with different random seeds and the same hyper-parameters.

Training Procedure. We strictly follow the originally pre-training process and parameter setting of other KEPLMs. We only adapt their publicly available code from English to Chinese and use the knowledge embedding trained on our medical KG. To have a fair comparison, the pre-training processing of SMedBERT is mostly set based on ENIRE-THU (Zhang et al., 2019) without layer-special learning rates in KnowBERT (Peters et al., 2019). We only pre-train SMedBERT on the collected medical data for 1 epoch. In pre-training

The Dataset Size in Our Experiments					
Dataset	Train	Dev	Test	Task	Metric
cMedQANER (Zhang et al., 2020b)	1,673	175	215	NER	F1
cMedQQ (Zhang et al., 2020b)	16,071	1,793	1,935	QM	F1
cMedQNLI (Zhang et al., 2020b)	80,950	9,065	9,969	NLI	F1
cMedQA (Zhang et al., 2017)	186,771	46,600	46,600	QA	Acc@1
WebMedQA (He et al., 2019)	252,850	31,605	31,655	QA	Acc@1
CHIP-RE *	43,649	-	10,622	RE	F1
DXY-NER	34,224	8,576	8,592	NER	F1
DXY-RE	141,696	35,456	35,794	RE	F1

* CHIP-RE dataset is released in CHIP 2020. (http://cips-chip.org.cn/2020/eval2)

Table 5: The statistical data and metric of eight datasets used in our SMedBERT model.

process, the learning rate is set to $5e^{-5}$ and batch size is 512 with the max sequence length is 512. For fine-tuning, we find the following ranges of possible values work well, i.e., batch size is {8,16}, learning rate (AdamW) is {$2e^{-5}$, $4e^{-5}$, $6e^{-5}$} and the number of epochs is {2,3,4}. Pre-training SMedBERT takes about 36 hours per epoch on 2 NVIDIA GeForce RTX 3090 GPUs.

C   Data and Embedding of Unsupervised Semantic Similarity

Since the KGs used in this paper is a directed graph, we first transform the directed "等价关系" (equivalence relations) pairs to undirected pairs and discard the duplicated pairs. For each positive pairs, we use head and tail as query respectively and sample the negative candidates based on the other. Specifically, we randomly select 19 negative entities with the same type and has a Jaro-Winkle similarity (Winkler, 1990) bigger 0.6 with the ground-truth entity. We select from all samples in **Dataset-1** with positive pairs that the neighbours sets of head and tail entity have Jaccard Index (Jaccard, 1912) no less than 0.75 and at least 3 common element to construct the **Dataset-2**. For **Dataset-3**, we count the frequency of all entity mentions in pre-training corpora, and treat mentions with frequency no more than 200 as low-frequency mentions.

Classic Word Representation Embedding: We train the character-level and word-level embedding using SGNS (Mikolov et al., 2013a) and GLOVE (Pennington et al., 2014) model respectively on our medical corpora with open-source toolkits[12]. We average the character embedding for all tokens in the mention to get the character-level representation. However, since some mentions are very rare in the corpora for word-level representation, we use the character-level representation as their word-level representation.

BERT-like Representation Embedding: We extract the token hidden features of the last layer and average the representations of the input tokens except [CLS] and [SEP] tag, to get a vector for each entity.

Similarity Measure: We try using the inverse of L2-distance and cosine similarity as measurement, and we find that cosine similarity always perform better. Hence, we report all experiment results under the cosine similarity metric.

[12]SGNS: https://github.com/JuGyang/word2vec-SGNS. Glove: https://github.com/stanfordnlp/GloVe