Improving Biomedical Pretrained Language Models with Knowledge

Zheng Yuan[1*] **Yijia Liu**[2] **Chuanqi Tan**[2†] **Songfang Huang**[2] **Fei Huang**[2]
[1]Tsinghua University [2]Alibaba Group
yuanz17@mails.tsinghua.edu.cn
{yanshan.lyj,chuanqi.tcq,songfang.hsf,f.huang}@alibaba-inc.com

Abstract

Pretrained language models have shown success in many natural language processing tasks. Many works explore incorporating knowledge into language models. In the biomedical domain, experts have taken decades of effort on building large-scale knowledge bases. For example, the Unified Medical Language System (UMLS) contains millions of entities with their synonyms and defines hundreds of relations among entities. Leveraging this knowledge can benefit a variety of downstream tasks such as named entity recognition and relation extraction. To this end, we propose KeBioLM, a biomedical pretrained language model that explicitly leverages knowledge from the UMLS knowledge bases. Specifically, we extract entities from PubMed abstracts and link them to UMLS. We then train a knowledge-aware language model that firstly applies a text-only encoding layer to learn entity representation and applies a text-entity fusion encoding to aggregate entity representation. Besides, we add two training objectives as entity detection and entity linking. Experiments on the named entity recognition and relation extraction from the BLURB benchmark demonstrate the effectiveness of our approach. Further analysis on a collected probing dataset shows that our model has better ability to model medical knowledge.

1 Introduction

Large-scale pretrained language models (PLMs) are proved to be effective in many natural language processing (NLP) tasks (Peters et al., 2018; Devlin et al., 2019). However, there are still many works that explore multiple strategies to improve the PLMs. Firstly, in specialized domains (i.e biomedical domain), many works demonstrate that using in-domain text (i.e. PubMed and MIMIC for biomedical domain) can further improve downstream tasks over general-domain PLMs (Lee et al., 2020; Peng et al., 2019; Gu et al., 2020; Shin et al., 2020; Lewis et al., 2020; Beltagy et al., 2019; Alsentzer et al., 2019). Secondly, unlike training language models (LMs) with unlabeled text, many works explore training the model with structural knowledge (i.e. triplets and facts) for better language understanding (Zhang et al., 2019; Peters et al., 2019; Févry et al., 2020; Wang et al., 2019). In this work, we propose to combine the above two strategies for a better **K**nowledge **e**nhanced **Bio**medical pretrained **L**anguage **M**odel (KeBioLM).

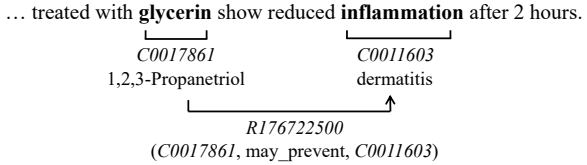

Figure 1: An example of the biomedical sentence. Two entities "glycerin" and "inflammation" are linked to *C0017861* (1,2,3-Propanetriol) and *C0011603* (dermatitis) respectively with a relation triplet (*C0017861*, *may_prevent*, *C0011603*) in UMLS.

As an applied discipline that needs a lot of facts and evidence, the biomedical and clinical fields have accumulated data and knowledge from a very early age (Ashburner et al., 2000; Stearns et al., 2001). One of the most representative work is Unified Medical Language System (UMLS) (Bodenreider, 2004) that contains more than 4M entities with their synonyms and defines over 900 kinds of relations. Figure 1 shows an example. There are two entities "glycerin" and "inflammation" that can be linked to *C0017861* (1,2,3-Propanetriol) and *C0011603* (dermatitis) respectively with a *may_prevent* relation in UMLS. As the most important facts in biomedical text, entities and relations can provide information for better text understanding (Xu et al., 2018; Yuan et al., 2020).

To this end, we propose to improve biomedical PLMs with explicit knowledge modeling. Firstly,

[*] Work done at Alibaba DAMO Academy.
[†] Corresponding author.

Proceedings of the BioNLP 2021 workshop, pages 180–190

we process the PubMed text to link entities to the knowledge base. We apply an entity recognition and linking tool ScispaCy (Neumann et al., 2019) to annotate 660M entities in 3.5M documents. Secondly, we implement a knowledge enhanced language model based on Févry et al. (2020), which performs a text-only encoding and a text-entity fusion encoding. Text-only encoding is responsible for bridging text and entities. Text-entity fusion encoding fuses information from tokens and knowledge from entities. Finally, two objectives as entity extraction and linking are added to learn better entity representations. To be noticed, we initialize the entity embeddings with TransE (Bordes et al., 2013), which leverages not only entity but also relation information of the knowledge graph.

We conduct experiments on the named entity recognition (NER) and relation extraction (RE) tasks in the BLURB benchmark dataset. Results show that our KeBioLM outperforms the previous work with average scores of 87.1 and 81.2 on 5 NER datasets and 3 RE datasets respectively. Furthermore, our KeBioLM also achieves better performance in a probing task that requires models to fill the masked entity in UMLS triplets.

We summary our contributions as follows[1]:

- We propose KeBioLM, a biomedical pretrained language model that explicitly incorporates knowledge from UMLS.
- We conduct experiments on 5 NER datasets and 3 RE datasets. Results demonstrate that our KeBioLM achieves the best performance on both NER and RE tasks.
- We collect a cloze-style probing dataset from UMLS relation triplets. The probing results show that our KeBioLM absorbs more knowledge than other biomedical PLMs.

2 Related Work

2.1 Biomedical PLMs

Models like ELMo (Peters et al., 2018) and BERT (Devlin et al., 2019) show the effectiveness of the paradigm of first pre-training an LM on the unlabeled text then fine-tuning the model on the downstream NLP tasks. However, direct application of the LMs pre-trained on the encyclopedia and web text usually fails on the biomedical domain, because of the distinctive terminologies and idioms.

The gap between general and biomedical domains inspires the researchers to propose LMs specially tailored for the biomedical domain. BioBERT (Lee et al., 2020) is the most widely used biomedical PLM which is trained on PubMed abstracts and PMC articles. It outperforms vanilla BERT in named entity recognition, relation extraction, and question answering tasks. Jin et al. (2019) train BioELMo with PubMed abstracts, and find features extracted by BioELMo contain entity-type and relational information. Different training corpora have been used for enhancing performance of sub-domain tasks. ClinicalBERT (Alsentzer et al., 2019), BlueBERT (Peng et al., 2019) and bio-lm (Lewis et al., 2020) utilize clinical notes MIMIC to improve clinical-related downstream tasks. SciBERT (Beltagy et al., 2019) uses papers from the biomedical and computer science domain as training corpora with a new vocabulary. KeBioLM is trained on PubMed abstracts to adapt to PubMed-related downstream tasks.

To understand the factors in pretraining biomedical LMs, Gu et al. (2020) study pretraining techniques systematically and propose PubMedBERT pretrained from scratch with an in-domain vocabulary. Lewis et al. (2020) also find using an in-domain vocabulary enhances the downstream performances. This inspires us to utilize the in-domain vocabulary for KeBioLM.

2.2 Knowledge-enhanced LMs

LMs like ELMo and BERT are trained to predict correlation between tokens, ignoring the meanings behind them. To capture both the textual and conceptual information, several knowledge-enhanced PLMs are proposed.

Entities are used for bridging tokens and knowledge graphs. Zhang et al. (2019) align tokens and entities within sentences, and aggregate token and entity representations via two multi-head self-attentions. KnowBert (Peters et al., 2019) and Entity as Experts (EAE) (Févry et al., 2020) use the entity linker to perform entity disambiguation for candidate entity spans and enhance token representations using entity embeddings. Inspired by entity-enhanced PLMs, we follow the model of EAE to inject biomedical knowledge into KeBioLM by performing entity detection and linking.

Relation triplets provide intrinsic knowledge be-

[1]Our codes and model can be found at https://github.com/GanjinZero/KeBioLM.

tween entity pairs. KEPLER (Wang et al., 2019) learns the knowledge embeddings through relation triplets while pretraining. K-BERT (Liu et al., 2020) converts input sentences into sentence trees by relation triplets to infuse knowledge.

In the biomedical domain, He et al. (2020) inject disease knowledge to existing PLMs by predicting diseases names and aspects on Wikipedia passages. Michalopoulos et al. (2020) use UMLS synonyms to supervise masked language modeling. We propose KeBioLM to infuse various kinds of biomedical knowledge from UMLS including but not limited to diseases.

3 Approach

In this paper, we assume to access an entity set $\mathcal{E} = \{e_1, ..., e_t\}$. For a sentence $\mathbf{x} = \{x_1, ..., x_n\}$, we assume some spans $m = (x_i, ..., x_j)$ can be grounded to one or more entities in $\mathcal{E}$. We further assume the disjuncture of these spans. In this paper, we use UMLS to set the entity set.

3.1 Model Architecture

To explicitly model both the textual and conceptual information, we follow Févry et al. (2020) and use a multi-layer self-attention network to encode both the text and entities. The model can be viewed as building the links between text and entities in the lower layers and fusing the text and entity representation in the upper layers. The overall architecture is shown in Figure 2. To be more specific, we set the PubMedBERT (Gu et al., 2020) as our backbone. We split the layers of the backbone into two groups, performing a text-only encoding and text-entity fusion encoding respectively.

Text-only encoding. For the first group, which is closer to the input, we extract the final hidden states and perform a token-wise classification to identify if the token is at the beginning, inside, or outside of a *mention* (i.e., the BIO scheme). The probabilities of the B/I/O label $\{l_i\}$ are written as:

$$\mathbf{h}_1, ..., \mathbf{h}_n = \text{Transformers}^0(x_1, ..., x_n) \quad (1)$$

$$p(l_i \mid \mathbf{x}) = \text{softmax}(\mathbf{W}_l \mathbf{h}_i + \mathbf{b}_l) \quad (2)$$

After identifying the mention boundary, we maintain a function $\mathcal{M}(i) \rightarrow \mathcal{E} \cup \{\text{NIL}\}$, which returns the entity of the i-th token belongs.[2] We collect the mentions with a sentence $\mathbf{x}$. For a mention $m = (s, t)$, where s and t represents the starting and ending indexes of m, we encode it as the concatenation of hidden states of the boundary tokens $\mathbf{h}_m = [\mathbf{h}_s; \mathbf{h}_t]$.

For an entity $e_j \in \mathcal{E}$ in the KG, we denote its entity embedding as $\mathbf{e}_j$. For a mention m, we search the k nearest entities of its projected representation $\mathbf{h}'_m = \mathbf{W}_m \mathbf{h}_m + \mathbf{b}_m$ in the entity embedding space, obtaining a set of entities $\mathcal{E}'$. The normalized similarity between $\mathbf{h}'_m$ and $\mathbf{e}_j$ is calculated as

$$a_j = \frac{\exp(\mathbf{h}'_m \cdot \mathbf{e}_j)}{\sum_{e_k \in \mathcal{E}'} \exp(\mathbf{h}'_m \cdot \mathbf{e}_k)} \quad (3)$$

The additional entity representation $\mathbf{e}'_m$ of m is calculated as a weighted sum of the embeddings $\mathbf{e}'_m = \sum_{e_j \in \mathcal{E}'} a_j \cdot \mathbf{e}_j$.

Text-entity fusion encoding. After getting the mentions and entities, we fuse the entity embeddings with the text embedding by summation. For the i-th token, the entity-enhanced embedding is calculated as:

$$\mathbf{h}^*_i = \begin{cases} \mathbf{h}_i + (\mathbf{W}_e \mathbf{e}'_m + \mathbf{b}_e), & \exists m, \mathcal{M}(i) = m, \\ \mathbf{h}_i, & \text{otherwise.} \end{cases} \quad (4)$$

$\mathcal{M}(i) = m$ represents the i-th token belong to entity e_m. The sequence of $\mathbf{h}^*_1, ..., \mathbf{h}^*_n$ is then fed into the second group of transformer layers to generate text-entity representations. The final hidden states $\mathbf{h}^f_i$ are calculated as:

$$\mathbf{h}^f_1, ..., \mathbf{h}^f_n = \text{Transformers}^1(\mathbf{h}^*_1, ..., \mathbf{h}^*_n) \quad (5)$$

3.2 Pretraining Tasks

We have three pretraining tasks for KeBioLM. Masked language modeling is a cloze-style task for predicting masked tokens. Since the entities are the main focus of our model, we add two tasks as entity detection and linking respectively following Févry et al. (2020). Finally, we jointly minimize the following loss:

$$\mathcal{L} = \mathcal{L}_{MLM} + \mathcal{L}_{ED} + \mathcal{L}_{EL} \quad (6)$$

Masked Language Modeling Like BERT and other LMs, we predict the masked tokens $\{x_i\}$ in inputs using the final hidden representations $\{\mathbf{h}^f_i\}$. The loss $\mathcal{L}_{MLM}$ is calculated based on the cross-entropy of masked and predicted tokens:

$$p_M(x_i \mid \mathbf{x}) = \text{softmax}(\mathbf{W}_m \mathbf{h}^f_i + \mathbf{b}_m) \quad (7)$$

$$\mathcal{L}_{MLM} = \sum -\log p_M(x_i \mid \mathbf{x}) \quad (8)$$

[2] NIL is returned when there is no entity being matched.

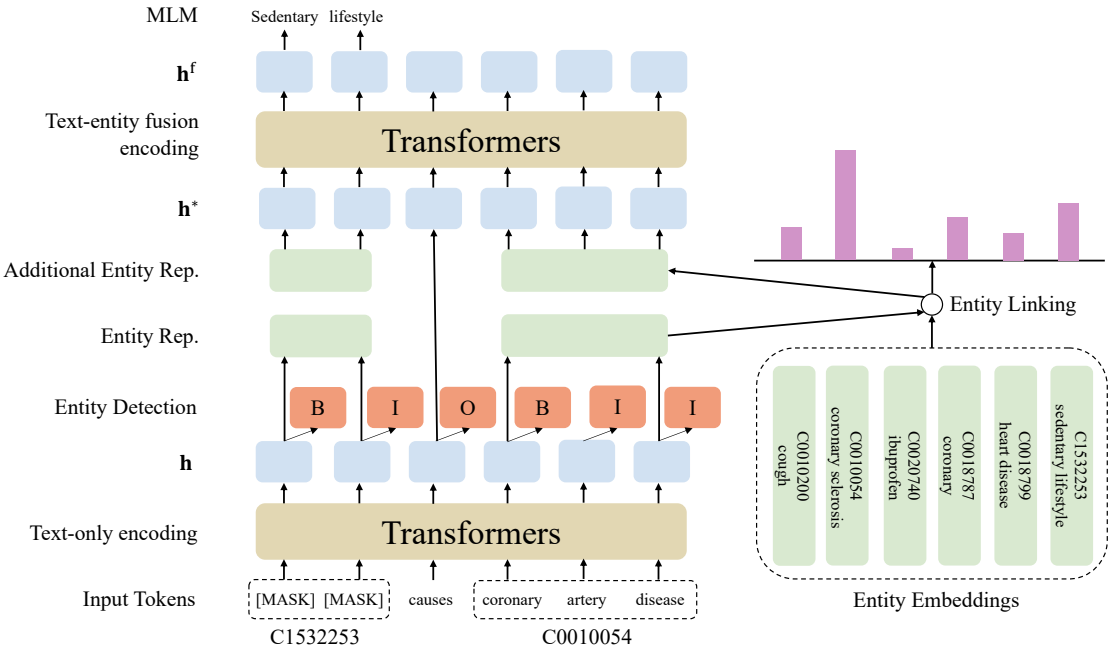

Figure 2: The overall architecture of KeBioLM.

Whole word masking is successful in training masked language models (Devlin et al., 2019; Cui et al., 2019). In the biomedical domain, entities are the semantic units of texts. Therefore, we extend this technique to whole entity masking. We mask all tokens within a word or entity span. KeBioLM replaces 12% of tokens to *[MASK]* and 1.5% tokens to random tokens. This is more difficult for models to recover tokens, which leads to learning better entity representations.

Entity Detection Entity detection is an important task in biomedical NLP to link the tokens to entities. Thus, We add an entity detection loss by calculating the cross-entropy for BIO labels:

$$\mathcal{L}_{ED} = \sum_{i=1}^{n} -\log p(l_i \mid \mathbf{x}) \tag{9}$$

Entity Linking One medical entity in different names linking to the same index permits the model to learn better text-entity representations. To link mention $\{m\}$ in texts with entities $\{e\}$ in entity set $\mathcal{E}$, we calculate the cross-entropy loss using similarities between $\{\mathbf{h}'_m\}$ and entities in $\mathcal{E}$:

$$\mathcal{L}_{EL} = \sum -\log \frac{\exp(\mathbf{h}'_m \cdot \mathbf{e})}{\sum_{e_j \in \mathcal{E}} \exp(\mathbf{h}'_m \cdot \mathbf{e}_j)} \tag{10}$$

3.3 Data Creation

Given a sentence S from PubMed content, we need to recognize entities and link them to the UMLS knowledge base. We use ScispaCy (Neumann et al., 2019), a robust biomedical NER and entity linking model, to annotate the sentence. Unlike previous work (Vashishth et al., 2020) that only retains recognized entities in a subset of Medical Subject Headings (MeSH) (Lipscomb, 2000), we relax the restriction to annotate all entities to UMLS 2020 AA release [3] whose linking scores are higher than a threshold of 0.85.

4 Experiments

In this section, we first introduce the pretraining details of KeBioLM. Then we introduce the BLURB datasets for evaluating our approach. Finally, we introduce a probing dataset based on UMLS triplets for evaluating knowledge modeling.

4.1 Pretraining Details

We use ScispaCy to acquire 477K CUIs and 660M entities among 3.5M PubMed documents[4] from PubMedDS dataset (Vashishth et al., 2020) as training corpora.

We initialize entity embeddings by TransE (Bordes et al., 2013) which learns embeddings from relation triplets. Relation triplets come from UMLS

[3] https://www.nlm.nih.gov/research/umls/licensedcontent/umlsarchives04.html#2020AA

[4] The count of documents in PubMedDS is based on https://arxiv.org/pdf/2005.00460v1.pdf.

	#Train	#Dev	#Test	#Ments	#Ments (UMLS)	#Ments (KeBioLM)
BC5chem	5,203	5,347	5,385	15,935	10,373	8,993
BC5dis	4,182	4,244	4,424	12,850	8,846	3,878
NCBI	5,137	787	960	6,884	1,985	1,091
BC2GM	15,197	3,061	6,325	24,583	2,808	2,423
JNLPBA	46,750	4,551	8,662	59,963	6,099	5,233
ChemProt	18,035	11,268	15,745	39,022	13,106	10,772
DDI	25,296	2,496	5,716	15,738	10,429	9,212
GAD	4,261	535	534	-	-	-

Table 1: The training instances (mentions for NER tasks and sentences with two entities for RE tasks) and the mention counts of NER and RE datasets preprocessed in BLURB benchmark respectively. The mention counts overlapping with UMLS 2020 AA release and KeBioLM are also listed. For the GAD dataset, annotated mentions do not appear in the BLURB preprocessed version.

2020 AA release. We train TransE with the L2-norm distance function and set embedding dim to 100. Adam (Kingma and Ba, 2014) is used as the optimizer with a learning rate of 1e-3, batch size of 2048, and train epoch of 30. Entity embeddings add 45.5M parameters to KeBioLM.

The parameters of transformers in KeBioLM are initialized from the checkpoint of PubMedBERT. We also use the vocabulary from PubMedBERT. AdamW (Loshchilov and Hutter, 2017) is used as the optimizer for KeBioLM with 10,000 steps warmup and linear decay. We use an 8-layer transformer for text-only encoding and a 4-layer transformer for text-entity fusion encoding. We set the learning rate to 5e-5, batch size to 512, max sequence length to 512, and training epochs to 2. For each input sequence, we limit the max entities count to 50 and the excessive entities will be truncated. To generate entity representation $\mathbf{e}'_m$, the most $k = 100$ similar entities are used. We train our model with 8 NVIDIA 16GB V100 GPUs.

4.2 Datasets

In this section, we evaluate KeBioLM on NER tasks and RE tasks of the BLURB benchmark[5] (Gu et al., 2020). For all tasks, we use the preprocessed version from BLURB. We measure the NER and RE datasets in terms of F1-score. Table 1 shows the counts of training instances in BLURB datasets (i.e., annotated mentions for NER datasets and sentences with two mentions for RE datasets). We also report the count of annotated mentions overlapping with the UMLS 2020 release and KeBioLM in each dataset. The percentage of mentions overlapping with KeBioLM ranges from 8.7% (NCBI-disease) to 58.5% (DDI) which indicates that KeBioLM learns entity knowledge related to downstream tasks.

4.2.1 Named Entity Recognition

BC5-chem & BC5-disease (Li et al., 2016) contain 1500 PubMed abstracts for extracting chemical and disease entities respectively.

NCBI-disease (Doğan et al., 2014) includes 793 PubMed abstracts to detect disease entities.

BC2GM (Smith et al., 2008) contains 20K PubMed sentences to extract gene entities.

JNLPBA (Collier and Kim, 2004) includes 2,000 PubMed abstracts to identify molecular biology-related entities. We ignore entity types in JNLPBA following Gu et al. (2020).

4.2.2 Relation Extraction

ChemProt (Krallinger et al., 2017) classifies the relation between chemicals and proteins within sentences from PubMed abstracts. Sentences are classified into 6 classes including a negative class.

DDI (Herrero-Zazo et al., 2013) is a RE dataset with sentence-level drug-drug relation on PubMed abstracts. There are four classes for relation: advice, effect, mechanism, and false.

GAD (Bravo et al., 2015) is a gene-disease relation binary classification dataset collected from PubMed sentences.

4.3 Fine-tuning Details

NER We follow Gu et al. (2020) to formulate NER tasks as sequential labeling tasks with the

[5] https://microsoft.github.io/BLURB/

	Bio-BERT	Sci-BERT	Clinical-BERT	Blue-BERT	disease-BERT	bio-lm†	PubMed-BERT	KeBio-LM
BC5chem	92.9	92.5	90.8	91.2	-	92.9	**93.3**	**93.3**$_{\pm 0.2}$
BC5dis	84.7	84.5	83.0	83.7	**86.5**	83.8	85.6	86.1$_{\pm 0.3}$*
NCBI	**89.1**	88.1	88.3	88.0	87.1	87.7	87.8	**89.1**$_{\pm 0.3}$*
BC2GM	83.8	83.4	81.7	81.9	-	87.0	84.5	**85.1**$_{\pm 1.6}$
JNLPBA	79.4	79.5	78.6	78.7	-	80.6	80.1	**82.0**$_{\pm 0.2}$*
NER	86.0	85.6	84.5	84.7	-	86.4	86.3	**87.1**$_{\pm 0.3}$*
ChemProt	76.1	75.2	72.0	71.5	-	75.4	77.2	**77.5**$_{\pm 0.3}$*
DDI	80.9	81.1	78.2	77.8	-	81.0	**82.4**	81.9$_{\pm 0.8}$
GAD	80.9	80.9	78.4	77.2	-	82.2	82.3	**84.3**$_{\pm 1.0}$*
RE	79.3	79.1	76.2	75.5	-	79.5	80.6	**81.2**$_{\pm 0.5}$*

Table 2: F1-scores on NER and RE tasks in BLURB benchmark. Standard deviations of KeBioLM are reported across five runs. Results of diseaseBERT-biobert and bio-lm come from their corresponded papers. Others are copied from BLURB. * indicates that $p \leq 0.05$ of one-sample t-test which compares whether the mean performance of KeBioLM is better than PubMedBERT. † Bio-lm applies different metrics with BLURB (micro F1 v.s. macro F1). Thus, we just list its results but do not directly compare with them.

BIO tagging scheme and ignore the entity types in NER datasets. We classify labels of tokens by a linear layer on top of the hidden representations.

RE We replace the entity mentions in RE datasets with entity indicators like @DISEASE$ or @GENE$ to avoid models classifying relations by memorizing entity names. We add these entity indicators into the vocabulary of LMs. We concatenate the representation of two concerned entities and feed it into a linear layer for relation classification.

Parameters We adopt AdamW as the optimizer with a 10% steps linear warmup and a linear decay. We search the hyperparameters of learning rate among 1e-5, 3e-5, and 5e-5. We fine-tune the model for 60 epochs. We evaluate the model at the end of each epoch and choose the best model according to the evaluation score on the development set. We set batch size as 16 when fine-tuning. The maximal input lengths are 512 for all NER datasets. We truncate ChemProt and DDI to 256 tokens, and GAD to 128 tokens. To perform a fair comparison, we fine-tune our model with 5 different seeds and report the average score.

4.4 Results

We compare KeBioLM with following base-size biomedical PLMs on the above-mentioned datasets: BioBERT (Lee et al., 2020), SciBERT (Beltagy et al., 2019), ClinicalBERT (Alsentzer et al., 2019), BlueBERT (Peng et al., 2019), bio-lm (Lewis et al., 2020), diseaseBERT (He et al., 2020), and PubMedBERT (Gu et al., 2020) [6].

Table 2 shows the main results on NER and RE datasets of the BLURB benchmark. In addition, we report the average scores for NER and RE tasks respectively. KeBioLM achieves state-of-the-art performance for NER and RE tasks. Compared with the strong baseline BioBERT, KeBioLM shows stable improvements in NER and RE datasets (+1.1 in NER, +1.9 in RE). Compared with our baseline model PubMedBERT, KeBioLM performs significantly better in BC5dis, NCBI, JNLPBA, ChemProt, and GAD ($p \leq 0.05$ based on one-sample t-test) and achieves better average scores (+0.8 in NER, +0.6 in RE). DiseaseBERT is a model carefully designed for predicting disease names and aspects, which leads to better performance in the BC5dis dataset (+0.4). They only report the promising results in disease-related tasks, however, our model obtains consistent promising performances across all kinds of biomedical tasks. In the BC2GM dataset, KeBioLM outperforms our baseline model PubMedBERT and other PLMs except for bio-lm, and the standard deviation of the BC2GM task is evidently larger than other tasks. Another exception is the DDI dataset, we observe a slight performance degradation compared to PubMedBERT (-0.5). The average performances demonstrate that fusing entity knowledge into the LM boosts the performances across the board.

[6] We use BioBERT v1.1, SciBERT-scivocab-uncased, BioClinicalBERT, BlueBERT-pubmed-mimic, bio-lm(RoBERTa-base-PM-M3-Voc), diseaseBERT-biobert and PubMedBERT-abstract versions for comparison.

	KeBio-LM	-wem	+rand	+frz
BC5chem	**93.3**	92.8	92.8	92.3
BC5dis	**86.1**	85.9	85.5	85.5
NCBI	**89.1**	88.4	88.8	88.3
BC2GM	85.1	84.5	84.5	**85.7**
JNLPBA	**82.0**	81.5	81.9	81.8
NER	**87.1**	86.6	86.7	86.7
ChemProt	**77.5**	77.3	76.3	76.8
DDI	**81.9**	80.6	81.4	80.7
GAD	**84.3**	83.1	82.3	82.8
RE	**81.2**	80.3	80.0	80.1

Table 3: Ablation studies for KeBioLM architecture on the BLURB benchmark. We use -wem, +rand and +frz to represent pretraining setting (a), (b) and (c), respectively.

	$l_0 = 8$ $l_1 = 4$	$l_0 = 4$ $l_1 = 8$	$l_0 = 12$ $l_1 = 0$
BC5chem	**93.3**	93.1	93.2
BC5dis	**86.1**	85.7	86.0
NCBI	**89.1**	88.5	88.4
BC2GM	85.1	84.8	**86.8**
JNLPBA	**82.0**	81.7	78.8
NER	**87.1**	86.8	86.6
ChemProt	77.5	**77.7**	77.6
DDI	**81.9**	81.0	80.1
GAD	**84.3**	82.9	83.2
RE	**81.2**	80.5	80.3

Table 4: Ablation studies for transformer layers count in KeBioLM on the BLURB benchmark.

4.5 Ablation Test

We conduct ablation tests to validate the effectiveness of each part in KeBioLM. We pretrain the model with the following settings and reuse the same parameters described above: (a) Remove whole entity masking and retain whole word masking while pretraining (-wem); (b) Initialize entity embeddings randomly (+rand); (c) Initialize entity embeddings by TransE and freeze the entity embeddings while pretraining (+frz).

In Table 3, we observe the following results. Firstly, comparing KeBioLM with setting (a) shows that whole entity masking boosting the performances consistently in all datasets (+0.5 in NER, +0.9 in RE). Secondly, comparing KeBioLM with setting (b) indicates initializing the entity embeddings randomly degrades performances in NER tasks and RE tasks (-0.4 in NER, -1.2 in RE). Entity embeddings initialized by TransE utilize relation knowledge in UMLS and enhance the results. Thirdly, freezing the entity embeddings in setting (c) reduces the performances on all datasets compared to KeBioLM except BC2GM (-0.4 in NER, -1.1 in RE). This indicates that updating entity embedding while pretraining helps KeBioLM to have better text-entity representations, and this leads to better downstream performances.

To evaluate how the count of transformer layers affects our model, we pretrain KeBioLM with the different number of layers. For the convenience of notation, denote l_0 is the layer count of text-only encoding and l_1 is the layer count of text-entity fusion encoding. We have the following settings: (i) $l_0 = 8, l_1 = 4$ (our base model), (ii)$l_0 = 4, l_1 = 8$, (iii)$l_0 = 12, l_1 = 0$ (without the second group of transformer layers, $\{\mathbf{h}_i\}$ are used for token representations). Results are shown in Table 4. Our base model (i) has better performance than setting (ii) (+0.3 in NER, +0.7 in RE). Training setting (iii) is equal to a traditional BERT model with additional entity extraction and entity linking tasks. The comparison with (i) and (iii) indicates that text-entity representations have better performances than text-only representations (+0.5 in NER, +0.9 in RE) in the same amount of parameters.

4.6 UMLS Knowledge Probing

We establish a probing dataset based on UMLS triplets to evaluate how LMs understand medical knowledge via pretraining.

4.6.1 Probing Dataset

UMLS triplets are stored in the form of (s, r, o) where s and o are CUIs in UMLS and r is a relation type. We generate two queries for one triplet based on names of CUIs and relation type:

- Q_1: *[CLS]* s r *[MASK] [SEP]*
- Q_2: *[CLS] [MASK]* r o *[SEP]*

For example, we sample a triplet and terms of corresponded entities (*C0048038*:apraclonidine, *may_prevent*, *C0028840*:ocular hypertension). We remove the underscores of relation names and generate two queries (we omit *[CLS]* and *[SEP]*):

- Q_1: apraclonidine may prevent *[MASK]*.
- Q_2: *[MASK]* may prevent ocular hypertension.

#Queries	#Relations	#Avg. CUIs
143,771	922	2.39

Table 5: The number of generated UMLS relation probing dataset.

For relation names end with "of", "as" , and "by", we add "is" in front of relation names. For instance, *translation_of* is converted to *is translation of*, *classified_as* is converted to *is classified as*, and *used_by* is converted to *is used by*. Commonly, different relation triplets can generate same query since triplets may overlap $(s, r, -)$ or $(-, r, o)$ with each other. We deduplicate all repeat queries and randomly choose at most 200 queries from all relation types in UMLS. After deduplication, one query can have multiple CUIs as answers. For example:

- Q: *[MASK]* may treat essential tremor.
- A_1: *C0282321*: propranolol hydrochloride
- A_2: *C0033497*: propranolol

We summarize our generated UMLS relation probing dataset in Table 5. Unlike LAMA (Petroni et al., 2019) and X-FACTR (Jiang et al., 2020) that contain less than 50 kinds of relation, our probing task is a more difficult task requiring a model to decode entities over 900 kinds of relations.

4.6.2 Multi [MASK] Decoding

To probe PLMs using generated queries, we require models to recover the masked tokens. Since biomedical entities are usually formed by multiple words and each word can be tokenized into several wordpieces (Wu et al., 2016), models have to recover multiple *[MASK]* tokens. We limit the max length of one entity is 10 for decoding.

We decode the multi *[MASK]* tokens using the confidence-based method described in Jiang et al. (2020). We also implement a beam search for decoding. Unlike beam search in machine translation that decodes tokens from left to right, we decode tokens in an arbitrary order. For each step, we calculate the probabilities of all undecoded masked tokens based on original input and decoded tokens. We predict only one token within undecoded tokens with the top $B = 5$ accumulated log probabilities. Decoding will be accomplished after count of *[MASK]* times iterations and we keep the best $B = 5$ decoding results. We skip the refinement stage since it is time-consuming and does not significantly improve the results.

	Type 1	Type 2	Overall
SciBERT	13.92	1.01	2.75
ClinicalBERT	4.19	0.33	0.79
BlueBERT	4.67	0.39	1.02
KeBioLM	**14.01**	**1.48**	**3.26**

Table 6: Results of the probing test in terms of Recall@5.

4.6.3 Evaluation Metric

Since multiple correct CUIs exist for one query, we consider a model answering the query correctly if any decoded tokens in any *[MASK]* length hit any of the correct CUIs. We evaluate the probing results by the relation-level macro-recall@5.

4.6.4 Probing Results

We classify probing queries into two types based on their difficulties. Type 1: **answers within queries** (24,260 queries); Type 2: **answers not in queries** (119,511 queries). Here are examples of Type 1 (Q_1 and A_1) and Type 2 (Q_2 and A_2) queries:

- Q_1: *[MASK]* has form tacrolimus monohydrate.
- A_1: *C0085149*: tacrolimus
- Q_2: cosyntropin may diagnose *[MASK]*.
- A_2: *C0001614*: adrenal cortex disease

Table 6 summarizes the probing results of different PLMs according to query types. Checkpoints of BioBERT and PubMedBERT miss a cls/predictions layer and cannot perform the probe directly. Compared to other PLMs, KeBioLM achieves the best scores in both two types and obviously outperforms BlueBERT and ClincalBERT with a large margin, which indicates that KeBioLM learns more medical knowledge.

Table 7 lists some probing examples. SciBERT can decode medical entities for *[MASK]* tokens which may be unrelated. KeBioLM decodes relation correctly and is aware of the synonyms of hepatic. KeBioLM states that *Vaccination may prevent tetanus* which is a correct but not precise statement.

5 Conclusions

In this paper, we propose to improve biomedical pretrained language models with knowledge. We

Query & Answer CUI	SciBERT	KeBioLM
omalizumab may treat *[MASK]* *C0004096*: asthma	migraine the disease	**asthma** severe allergic asthma
phentolamine may diagnose *[MASK]* *C0031511*: phaeochromocytoma	depression the serotonin syndrome	**pheochromocytoma** renovascular hypertension
[MASK] is noun form of hepatic *C0023884*: liver	it the form of hepatic	**liver** hepatic only
[MASK] may prevent tetanus *C0305062*: tetanus toxoid	it bcg vaccination	vaccination prophylactic tetanus vaccination

Table 7: Probing examples of UMLS relation triplets. Queries and answer CUIs are listed. We only list one correct CUI for each query. For each model, one *[MASK]* token decoding result and an example of multi *[MASK]* decoding result are displayed. Bold text represents a term of the answer CUI.

propose KeBioLM which applies text-only encoding and text-entity fusion encoding and has two additional entity-related pretraining tasks: entity detection and entity linking. Extensive experiments have shown that KeBioLM outperforms other PLMs on NER and RE datasets of the BLURB benchmark. We further probe biomedical PLMs by querying UMLS relation triplets, which indicates KeBioLM absorbs more biomedical knowledge than others. In this work, we only leverage the relation information in TransE to initialize the entity embeddings. We will further investigate how to directly incorporate the relation information into LMs in the future.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful comments and suggestions. This work is supported by Alibaba Group through Alibaba Research Intern Program.

References

Emily Alsentzer, John Murphy, William Boag, Wei-Hung Weng, Di Jindi, Tristan Naumann, and Matthew McDermott. 2019. Publicly available clinical BERT embeddings. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 72–78, Minneapolis, Minnesota, USA. Association for Computational Linguistics.

Michael Ashburner, Catherine A Ball, Judith A Blake, David Botstein, Heather Butler, J Michael Cherry, Allan P Davis, Kara Dolinski, Selina S Dwight, Janan T Eppig, et al. 2000. Gene ontology: tool for the unification of biology. *Nature genetics*, 25(1):25–29.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. 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 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Olivier Bodenreider. 2004. The unified medical language system (umls): integrating biomedical terminology. *Nucleic acids research*, 32(suppl_1):D267–D270.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Neural Information Processing Systems (NIPS)*, pages 1–9.

Àlex Bravo, Janet Piñero, Núria Queralt-Rosinach, Michael Rautschka, and Laura I Furlong. 2015. Extraction of relations between genes and diseases from text and large-scale data analysis: implications for translational research. *BMC bioinformatics*, 16(1):1–17.

Nigel Collier and Jin-Dong Kim. 2004. Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Joint Workshop on Natural Language Processing in Biomedicine and its Applications (NLPBA/BioNLP)*, pages 73–78, Geneva, Switzerland. COLING.

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. *arXiv preprint arXiv:1906.08101*.

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.

Rezarta Islamaj Doğan, Robert Leaman, and Zhiyong Lu. 2014. Ncbi disease corpus: a resource for dis-

ease name recognition and concept normalization. *Journal of biomedical informatics*, 47:1–10.

Thibault Févry, Livio Baldini Soares, Nicholas FitzGerald, Eunsol Choi, and Tom Kwiatkowski. 2020. Entities as experts: Sparse memory access with entity supervision. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4937–4951, Online. Association for Computational Linguistics.

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. *arXiv preprint arXiv:2007.15779*.

Yun He, Ziwei Zhu, Yin Zhang, Qin Chen, and James Caverlee. 2020. Infusing Disease Knowledge into BERT for Health Question Answering, Medical Inference and Disease Name Recognition. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4604–4614, Online. Association for Computational Linguistics.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, 46(5):914–920.

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020. X-FACTR: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5943–5959, Online. Association for Computational Linguistics.

Qiao Jin, Bhuwan Dhingra, William Cohen, and Xinghua Lu. 2019. Probing biomedical embeddings from language models. In *Proceedings of the 3rd Workshop on Evaluating Vector Space Representations for NLP*, pages 82–89.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Martin Krallinger, Obdulia Rabal, Saber A Akhondi, Martın Pérez Pérez, Jesús Santamaría, Gael Pérez Rodríguez, et al. 2017. Overview of the biocreative vi chemical-protein interaction track. In *Proceedings of the sixth BioCreative challenge evaluation workshop*, volume 1, pages 141–146.

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. *Bioinformatics*, 36(4):1234–1240.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online. Association for Computational Linguistics.

Jiao Li, Yueping Sun, Robin J Johnson, Daniela Sciaky, Chih-Hsuan Wei, Robert Leaman, Allan Peter Davis, Carolyn J Mattingly, Thomas C Wiegers, and Zhiyong Lu. 2016. Biocreative v cdr task corpus: a resource for chemical disease relation extraction. *Database*, 2016.

Carolyn E Lipscomb. 2000. Medical subject headings (mesh). *Bulletin of the Medical Library Association*, 88(3):265.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-bert: Enabling language representation with knowledge graph. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(03):2901–2908.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alex Wong. 2020. Umlsbert: Clinical domain knowledge augmentation of contextual embeddings using the unified medical language system metathesaurus.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. ScispaCy: Fast and robust models for biomedical natural language processing. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 319–327, Florence, Italy. Association for Computational Linguistics.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Matthew E. Peters, Mark Neumann, Robert Logan, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. 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 43–54, Hong Kong, China. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Hoo-Chang Shin, Yang Zhang, Evelina Bakhturina, Raul Puri, Mostofa Patwary, Mohammad Shoeybi, and Raghav Mani. 2020. BioMegatron: Larger biomedical domain language model. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4700–4706, Online. Association for Computational Linguistics.

Larry Smith, Lorraine K Tanabe, Rie Johnson nee Ando, Cheng-Ju Kuo, I-Fang Chung, Chun-Nan Hsu, Yu-Shi Lin, Roman Klinger, Christoph M Friedrich, Kuzman Ganchev, et al. 2008. Overview of biocreative ii gene mention recognition. *Genome biology*, 9(2):1–19.

Michael Q Stearns, Colin Price, Kent A Spackman, and Amy Y Wang. 2001. Snomed clinical terms: overview of the development process and project status. In *Proceedings of the AMIA Symposium*, page 662. American Medical Informatics Association.

Shikhar Vashishth, Rishabh Joshi, Ritam Dutt, Denis Newman-Griffis, and Carolyn Rose. 2020. Medtype: Improving medical entity linking with semantic type prediction. *arXiv preprint arXiv:2005.00460*.

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2019. Kepler: A unified model for knowledge embedding and pre-trained language representation. *arXiv preprint arXiv:1911.06136*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

B. Xu, X. Shi, Z. Zhao, and W. Zheng. 2018. Leveraging biomedical resources in bi-lstm for drug-drug interaction extraction. *IEEE Access*, 6:33432–33439.

Zheng Yuan, Zhengyun Zhao, and Sheng Yu. 2020. Coder: Knowledge infused cross-lingual medical term embedding for term normalization. *arXiv preprint arXiv:2011.02947*.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.