Graph-Enriched Biomedical Entity Representation Transformer

Andrey Sakhovskiy$^{1,3[0000-0003-2762-2910]}$, Natalia Semenova$^{1,2[0000-0003-4189-5739]}$, Artur Kadurin$^{2[0000-0003-1482-9365]}$, and Elena Tutubalina$^{2,3[0000-0001-7936-0284]}$

[1] Sber AI, Moscow, Russia
[2] Artificial Intelligence Research Institute, Moscow, Russia
[3] Kazan (Volga Region) Federal University, Kazan, Russia
{andrey.sakhovskiy,tutubalinaev}@gmail.com

Abstract. Infusing external domain-specific knowledge about diverse biomedical concepts and relationships into language models (LMs) advances their ability to handle specialised in-domain tasks like medical concept normalization (MCN). However, existing biomedical LMs are primarily trained with contrastive learning using synonymous concept names from a terminology (e.g., UMLS) as positive anchors, while accurate aggregation of the features of graph nodes and neighbors remains a challenge. In this paper, we present Graph-Enriched Biomedical Entity Representation Transformer (GEBERT) which captures graph structural data from the UMLS via graph neural networks and contrastive learning. In GEBERT, we enrich the entity representations by introducing an additional graph-based node-level contrastive objective. To enable mutual knowledge sharing among the textual and the structural modalities, we minimize the contrastive objective between a concept's node representation and its textual embedding obtained via LM. We explore several state-of-the-art convolutional graph architectures, namely GraphSAGE and GAT, to learn relational information from local node neighborhood. After task-specific supervision, GEBERT achieves state-of-the-art results on five MCN datasets in English.

Keywords: Natural language processing · Biomedical entity representations · Knowledge representation · Graph neural network · Entity linking

1 Introduction

Biomedical entity representation finds application in numerous biomedical tasks, such as knowledge discovery, information extraction, and search [5,31,15,9,21,29]. Nonetheless, identifying specific biomedical concepts like diseases, symptoms, and drugs in free-form text can be problematic because their names, abbreviations, and spelling inconsistencies are highly variable. Moreover, a single biomedical concept can appear in numerous nonstandard forms. This challenge can be addressed by medical concept normalization (MCN; also called medical concept linking) which is the task where entity mentions are mapped against a large

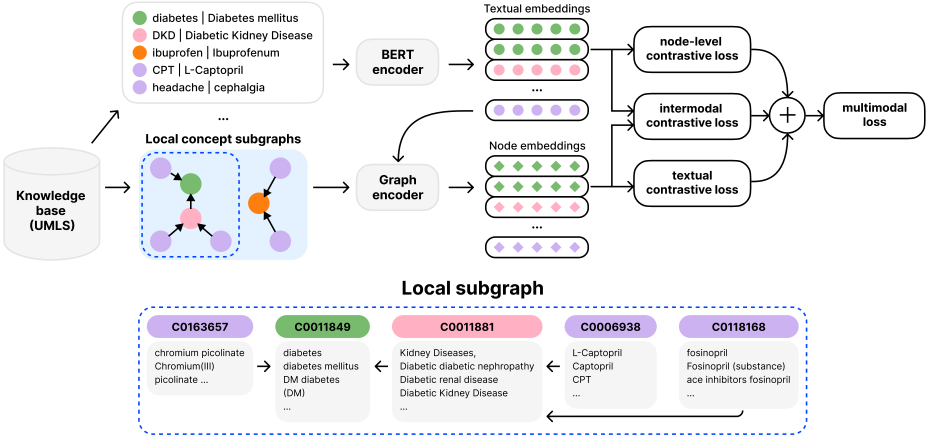

Fig. 1. GEBERT model's architecture overview. Our model consists of two encoders for text and graph data. Graph encoder uses textual embeddings from BERT as an additional input. The loss function is a weighted sum of three terms: textual and node-level contrastive losses, and intermodal contrastive loss to match representations between different encoders.

set of medical concept names and their concept unique identifiers (CUIs) from a knowledge base (KB). In addition to a high variation of mentions, the biomedical domain is characterized by extensive KBs such as the Unified Medical Language System (UMLS) [3].

Early models for MCN [26,19] commonly used classification type losses that are often trained on narrow benchmarks and lead to significant performance degradation on other domains and structurally different texts. Modern approaches usually employ similarity between *embeddings* (distributed representations) of entity mentions and concepts constructed by language models (LMs) and a BERT [7]-like ranking architecture [38,30,32]. However, the problem of learning meaningful and robust entity representations still poses a challenge for LMs.

Biomedical knowledge has been injected into neural networks by metric learning and contrastive learning [27,24,22,17,37]. Naturally, knowledge from KBs is typically represented as triples (head, relation, tail); head and tail terms of the same and different concepts serve as positive and negative pairs (e.g, *diabetic nephropathy* is a synonym to *diabetic kidney disease* and differs from *diabetes mellitus*, as shown in Fig. 1). In addition to representation learning with textual triples [27,22,17], [37] proposed to use term-relation-term similarity inspired by semantic matching methods like TransE [4] and DistMult [36]. However, these structural approaches are unable efficiently use textual node features.

In this paper, we present **G**raph-**E**nriched **B**iomedical **E**ntity **R**epresentation **T**ransformer (GEBERT), which uses contrastive learning and graph neural networks to capture graph structural data from a KB. As shown in Fig. 1, the GEBERT architecture consists of three losses: (i) a textual contrastive loss that learns on synonymous concept names; (ii) a node-level contrastive loss that learns to produce concept embeddings that are independent of the surface form choice; (iii) an intermodal contrastive loss that allows the information exchange between the textual and graph encoder. The source code and pre-trained models are freely available[4].

2 Related Work

MCN is typically formulated as a classification or ranking problem with a wide range of features, including syntactic, morphological parsing, dictionaries of medical concepts and their synonyms, and distances between formal concept names and raw entity mentions in terms of sparse/dense representations [1,33,6].

Classification approaches [26,19] are typically trained on labeled datasets with mentions linked to a small set of target concepts, while existing biomedical KBs, such as the UMLS, have millions of concepts. Ranking models use training pairs of positive and negative terms from a dictionary to determine how similar entity mention and concept names are. [24] trained a triplet network to rank the candidate concept names based on their similarity with a disease mention. Convolutional and pooling layers based on *word* embeddings were chosen as the encoder. [30] proposed the BioSyn model, which maximizes the likelihood that all synonym representations are present in the top 20 candidates. As a similarity function, BioSyn combines the sparse and dense scores with a scalar weight. To encode the morphological information of given strings, sparse scores are computed on character-level TF-IDF representations. Dense scores are defined by the similarity between `CLS` tokens of a single vector of input in BioBERT [14]. [22] proposed a DILBERT model which optimizes the relative similarity of mentions and concept names from a terminology via triplet loss. Different negative sampling strategies were applied to DILBERT models including random sampling and re-sampling using concept names from concepts' parents (parent-child or broader-narrower relationships). However, both DILBERT and BioSyn were trained on a dataset in English with a narrow subsample of concepts from a specific terminology.

There are few attempts to inject external domain-specific knowledge (e.g., UMLS) into pre-trained language models (LMs) in order to learn entity representations [27,20,17,18,37]. [27] presented an encoding framework with context, concept, and synonym-based objectives. Synonym-based objective enforces similar representations between synonymous names, while concept-based objective pulls the name's representations closer to its concept's centroid. This model was trained on 29 million PubMed abstracts annotated with UMLS concepts of diseases and chemicals. However, ranking on these embeddings shows worse results than models with dictionaries and features on three sets in English. UmlsBERT [20], a bert-like LM, integrates the domain knowledge from UMLS during the pre-training process via a novel knowledge augmentation strategy. Recently, a self-alignment pretraining (SAP) [17] procedure for learning on synonymous

[4] `https://github.com/Andoree/GEBERT`

term pairs from the UMLS has been proposed. The authors of the procedure released a BERT-based SapBERT model that is pre-trained on English synonyms from UMLS. SapBERT pre-trained on UMLS outperformed on MCN task several domain-specific LMs such as BioBERT [14], SciBERT [2], and UmlsBERT [20].

The SAP procedure makes no use of the UMLS graph's structure that describes the relations between concepts. To address the limitation, a relation-aware language model named CODER was proposed [37]. The authors infused the relational knowledge from the UMLS graph into the original SAP procedure by introducing a relational loss in addition to synonym-based contrastive loss. The main difference compared to SapBERT is that CODER simultaneously learns from synonyms and related concepts.

3 Background and architecture

Let V denote a set of all concepts present in a knowledge base. Knowledge graphs, such as UMLS, usually store relational information in the form of relation triplets (h, r, t) where h and t are concepts from V and r is a relation type. In this work, we omit the relation types and view the UMLS graph as an oriented unlabelled graph $G = G(V, \mathcal{E})$, where $\mathcal{E}$ is the set of oriented edges with relation types dropped. For each concept $c \in V$, UMLS presents a set of k synonymous terms $S_c = \{s_1^c, s_2^c, \ldots, s_k^c\}$. For each term from S_c, the UMLS stores the label of the language it came from. Let s denote an arbitrary textual term which, in other words, is a concept name. The goal of the biomedical entity linking task is to predict a concept $c \in V$ that s belongs to.

3.1 Self-alignment pretraining

A reasonable and straightforward way to learn an informative representation space of biomedical entities is to represent textual knowledge from KG in the form of positive and negative term pairs and optimize some contrastive learning loss function.

In this work, we adopt the self-alignment pretraining (SAP) procedure [17]. To enrich the training procedure with harder negative samples, SAP employs online hard mining for valid triplets [23,10]. During SAP, the model is encouraged to produce similar representations for all terms that represent the same concept (share the same CUI). At each pretraining step, we sample a batch B that consists of N positive samples $(c, s_i^c, s_j^c) \in V \times S_c \times S_c$. Given B, SAP constructs all possible term triplets (s^p, s^a, s^n) such that $p = a$ and $n \neq a$. s^a is called an anchor term; s^p is a positive term for s^a (i.e., s^p and s^a are synonymous terms representing the same concept $a = p$); s^n is a negative term for s^a (i.e., s^n and s^a represent non-matching concepts). Each triple produces a positive pair (s^a, s^p) and a negative pair (s^a, s^n). To keep only the most informative triples, we use online hard mining for valid triplets with respect to the following constraint:

$$\|f_{enc}(s^a) - f_{enc}(s^p)\| < \|f_{enc}(s^a) - f_{enc}(s^n)\| + \lambda$$

where f_{enc} is a BERT-based textual encoder, $\|\cdot\|$ is the normalized L_2-norm, and λ is a pre-defined mining margin. Thus, the mining procedure discards all the triplets such that the distance from an anchor to its negative sample is greater than the distance to its positive sample by more than λ. Let $\mathcal{P}$ and $\mathcal{N}$ denote the sets of all positive and negative term pairs, respectively. The SAP procedure utilizes the Multi-Similarity (MS) loss [35] to learn from $\mathcal{P}$ and $\mathcal{N}$.

$$\mathcal{L}_{sap} = \frac{1}{|B|}\sum_{i=1}^{|B|}\left(\frac{1}{\alpha}\log\left(1+\sum_{n\in\mathcal{N}_i} e^{\alpha(S_{in}-\epsilon)}\right)+ \ +\frac{1}{\beta}\log\left(1+\sum_{p\in\mathcal{P}_i} e^{-\beta(S_{ip}-\epsilon)}\right)\right),$$

where α, β, and ϵ are the parameters of MS-loss. $\mathcal{P}_i$ and $\mathcal{N}_i$ are the sets of positive and negative samples for the anchor concept i.

3.2 Graph neural networks

Message passing framework A common way to learn structured knowledge from graph is to iteratively update the representation of node v by passing and aggregating messages from local node neighborhood $N(v)$ using a graph neural network. Message Passing Neural Networks (MPNN) [11] framework that describes an update of node representation $h_v^{(l)}$ at the $(l+1)$-th MPNN layer as the composition of a message function f_m and an update function f_u:

$$h_v^{(l+1)} = f_u(h_v^{(l)}, \sum_{u\in N(v)} f_m(h_v^{(l)}, h_u^{(l)}))$$

where $N(v)$ is the set of neighboring nodes of node v. As the number of neighbors can significantly vary across different nodes and result in excessive computational complexity, we use a uniformly drawn fixed-size subset of neighbors instead of the full node neighborhood as proposed by [13]. The choice of f_m and f_u functions is the key difference between various GNN models that fall under the MPNN framework. In GraphSAGE [13], a common and rather simple implementation of MPNN framework, an element-wise operator (e.g., max- or mean-pooling) is used as an f_m to aggregate the vectors of neighbor nodes $N(v)$ into a single vector. The aggregated representation is further concatenated with the original representation and passed to a linear layer W^{l+1} with a non-linear activation function σ. In this work, we use the GraphSAGE implementation with mean-pooling aggregation:

$$h_v^{(l+1)} = \sigma(W^l \cdot [h_v^{(l)} \parallel MEAN(N(v))])$$

where $MEAN$ is the mean-pooling operator, $[\cdot \parallel \cdot]$ is the concatenation of two vectors. The simplicity of GraphSAGE prevents a context-aware message passing since the mean-pooling treats all nodes from $N(v)$ with equal weights. It means that graphSAGE is not able to weigh neighborhoods with respect to their relevance to the target node.

Graph attention network (GAT) [34] addresses the limitation by introducing the self-attention over neighboring nodes and learning the aggregated neighborhood representation as the weighted sum of neighboring nodes representations. Given two node representations $h_u^{(l-1)}$ and $h_v^{(l-1)}$, the l-th GAT layer computes the relevance of node u for the target node v as the normalized attention score $\alpha_{uv}^{(l)}$:

$$e_{uv}^{(l)} = a^T \cdot LeakyReLU(W^{(l)} \cdot [h_u^{(l-1)} \parallel h_v^{(l-1)}])$$

$$\alpha_{uv}^{(l)} = \frac{exp(e_{uv}^{(l)})}{\sum_{w \in N(v)} exp(e_{wv}^{(l)})}$$

With the attention scores obtained, the aggregated neighborhood representation is computed as a weighted sum of neighboring nodes embeddings.

3.3 GEBERT

Textual loss In GEBERT, we adopt and extend the pretraining procedure described in Sec. 3.1. At each training step, we begin by sampling a batch B of random positive samples. Each positive sample is a triplet $t = (c, s_i^c, s_j^c) \in V \times S_c \times S_c$ which consists of concept (node) identifier and two synonymous concept names. For each t, we randomly sample a set of concept node's neighbors (concept's neighborhood) $N(c)$ using the graph G. Next, we produce a textual embedding for each term present in B using a textual encoder f_{enc} and calculate the textual loss $\mathcal{L}_{sap}$ using the representations of concept names from the batch B.

Node-level loss We define the batch B's subgraph $G_B = G(V_B, \mathcal{E}_B)$ as the union of concept nodes from batch B and all nodes and edges from the concept's neighborhood $N(c), c \in B$. Our goal is to enrich the embedding space of the textual encoder f_{enc} with the structural knowledge stored in G_B while keeping the embeddings of terms representing the same concept close to each other by cosine distance. As shown in Fig. 1, for each positive pair, textual encoder f_{enc} produces two textual embeddings: for the first and the second terms of the pair, respectively. These embeddings are passed to a graph encoder for node initialization.

Let $H_1 \in R^{|B| \times d}$ and $H_2 \in R^{|B| \times d}$ denote the matrices of d-dimensional textual embeddings of the first and the second terms of positive pairs from B, respectively. To obtain two graph-enriched representations g_c^1 and g_c^2 of the node (concept) c, we stack multiple MPNN layers to aggregate the structural information from the node's neighborhood $N(c)$ using the H_1 and H_2 as the initial representations of nodes V_B. Next, we collect all positive node samples (c, g_c^1, g_c^2) and pass them to the SAP procedure to obtain a node-level contrastive loss $\mathcal{L}_{node}$. Thus, the major difference between $\mathcal{L}_{sap}$ and $\mathcal{L}_{node}$ is that the latter operates on the graph-aware node (concept) embeddings rather than textual embeddings of concept terms.

Intermodal loss Let (c, g_c^1, g_c^2) and $(c, f_{enc}(s_i^c), f_{enc}(s_j^c))$ denote a term-level and node-level positive samples of concept c, respectively. We construct two intermodal positive samples $(c, g_c^1, f_{enc}(s_j^c))$ and $(c, f_{enc}(s_i^c), g_c^2)$ each containing a node-level and a term-level representation of c. To allow a mutual knowledge exchange between the textual encoder f_{enc} and a graph encoder, we collect all intermodal positive pairs and once again apply the SAP procedure to optimize the intermodal contrastive MS-loss $\mathcal{L}_{int}$, that minimizes the distance between textual and node representations of the same concept and pushes away the representations of non-matching concepts.

$$\mathcal{L}_{GEBERT} = \mathcal{L}_{sap} + \lambda_{node}\mathcal{L}_{node} + \lambda_{int}\mathcal{L}_{int}, \tag{1}$$

where λ_{node} and λ_{int} are the pre-selected weights of $\mathcal{L}_{node}$ and $\mathcal{L}_{int}$.

4 Experimental Evaluation

We initialized GEBERT with PubMedBERT[5] [12]. The model was trained on an English UMLS graph for 1 epoch with a learning rate of $2 \cdot 10^{-5}$. We set $\lambda_{node} = \lambda_{int} = 0.1$ and the maximum size of node neighborhood to 3. As a graph encoder, we use 3 consecutive layers of either GraphSAGE or GAT.

We implemented two versions of GEBERT that differ in graph encoder architecture: (i) GraphSAGE-GEBERT and (ii) GAT-GEBERT. To train our implementations of GEBERT, we use the UMLS 2020AB release which contains approximately 4.4 million concepts and 15.9 million unique concept names from 215 source vocabularies. We remove all concept names that originate from non-English source vocabularies and remove all duplicated edges. We follow the batching strategy proposed by the authors of SapBERT [17]: to ensure each batch includes a sufficient number of positive pairs, we pre-compute synonym pairs with common CUIs. If a concept produces more than 50 positive pairs, we randomly sample 50 of them.

Data To evaluate our models, we use 5 datasets: (i) NCBI [8], (ii) BC5CDR-D [16], (iii) BC5CDR-D [16], (iv) TAC2017ADR [28], (v) BC2GN [25]. Due to overlap between official train/test sets, we follow [32] and use the presented *refined* test sets. For details on preprocessing and sets, please refer to [32]. We have used the publicly available code provided by the authors at `https://github.com/insilicomedicine/Fair-Evaluation-BERT`.

The NCBI Disease Corpus [8] is a collection of 793 abstracts from PubMed, which include mentions of diseases and their corresponding concepts. [16] introduces a task for the extraction of chemical-disease relations (CDR) from 1500 PubMed abstracts, with annotations for both chemicals and diseases. BioCreative II GN (BC2GN) [25] contains human gene and gene product mentions in PubMed abstracts for gene normalization (GN). TAC 2017 ADR challenge [28] focuses on extracting adverse drug reactions (ADRs) from product labels, such as prescribing information or package inserts.

[5] `huggingface.co/BiomedNLP-PubMedBERT-base-uncased-abstract-fulltext`

Table 1. Evaluation of models on academic evaluation datasets (*refined* test sets).

Model	**NCBI**		**BC5CDR D**		**BC5CDR C**		**TAC ADR**		**BC2GN**	
	@1	@5	@1	@5	@1	@5	@1	@5	@1	@5
	Zero-shot evaluation									
enSapBERT	**71.57**	**84.31**	73.67	84.32	85.88	91.29	**82.58**	**90.93**	**87.72**	92.18
enCODER	69.12	**84.31**	73.36	**85.54**	84.24	90.82	79.15	88.41	84.47	90.96
GraphSAGE-GEBERT	70.59	82.84	73.97	84.02	**86.12**	**91.76**	81.54	90.61	86.19	93.10
GAT-GEBERT	70.59	83.33	**74.58**	85.39	85.41	**91.76**	82.12	89.90	87.31	**92.79**
	Evaluation after fine-tuning									
enSapBERT	75.49	**84.80**	74.89	84.02	86.12	**93.41**	86.20	91.26	88.83	93.30
enCODER	73.53	82.84	75.34	**85.24**	86.35	92.71	84.72	91.00	88.32	92.49
GraphSAGE-GEBERT	**76.47**	**84.80**	**75.80**	84.93	**87.53**	93.18	**86.33**	**91.97**	**88.93**	93.50
GAT-GEBERT	73.04	**84.80**	75.49	84.78	87.06	92.71	85.82	91.32	88.63	**93.60**

Table 2. Error analysis examples of mentions, predicted and golden concept names from GraphSAGE-GEBERT on TAC ADR refined test set.

Mention	**Predicted concept**	**Golden concept**
clinical deterioration	clinical worsening	general physical health deterioration
mean change in heart rate 1 2 beats per minute	mean heart rate higher by an average of 1 to 2 bpm	heart rate abnormal
increased number of lashes	increased lacrimation	growth of eyelashes
body temperature dysregulation	body temperature fluctuation	temperature regulation disorder
emerging suicidality	suicidality	suicidal intention
homicidal threats	homicidal attempt	homicidal ideation

Experimental Setup We evaluate the proposed models in two settings: (i) zero-shot evaluation and (ii) evaluation with fine-tuning.

For zero-shot evaluation, we employ a ranking approach [32] that is built on the embeddings of mentions and potential concepts. Each entity mention and concept name is first passed through a model that produces their embeddings and then through an average pooling layer that yields a fixed-sized vector. The inference task is then reduced to finding the closest concept name representation to entity mention representation in a common embedding space, where the Euclidean distance can be used as the metric. Nearest concept names are chosen as top-k concepts for entities.

For the evaluation with fine-tuning, we utilize BioSyn [30], a model that iteratively updates candidates by applying synonym marginalization. The model utilizes two distinct similarity functions designed to capture both morphological and semantic information. The sparse representations are obtained with TF-IDF and dense representations are obtained using a BERT-based model. We adopt the default BioSyn hyper-parameters [30]. For each dataset, we trained BioSyn for 20 epochs, following [32].

We evaluate the models in the IR scenario, where the goal is to find top-k concepts for every entity mention in a dictionary of concept names and their identifiers. Following previous works [27,32,30,17,18,37], we use the top-k accuracy as the evaluation metric: Acc@k = 1 if the correct UMLS concept unique identifier is retrieved at the rank $\leq k$, otherwise Acc@k = 0.

Compared Representations We compare the following representations:

- *enSapBERT*: a BERT-based metric learning framework that generates hard triplets based on the UMLS for pre-training [17]. The model is adopted from `huggingface.co/cambridgeltl/SapBERT-from-PubMedBERT-fulltext`.
- *enCODER*: a contrastive learning model inspired by semantic matching methods that uses both synonyms and relations from the UMLS [37]. We have used the model provided at `huggingface.co/GanjinZero/coder_eng`.

4.1 Results

Tab. 1 shows the Acc@1 and Acc@5 metrics for five datasets. In zero-shot evaluation, basic enSapBERT outperformed CODER and GEBERT on 3 of 5 datasets in terms of Acc@1. On disease and chemical mentions from BC5CDR, the best models are GraphSAGE-GEBERT and GAT-GEBERT with a slight improvement over enSapBERT. An interesting finding is that enCODER is the worst performing model on all five datasets in terms of Acc@1 despite the fact it inherited one of two its training objectives from enSapBERT. The situation changes after the fine-tuning: our GraphSAGE-GEBERT model becomes a leader on all five academic datasets with an insignificant improvement against enSapBERT on TAC ADR and BC2GN (0.13% and 0.1%, respectively) and a notable improvement on NCBI, BC5CDR Disease, and BC5CDR Chemical (0.98%, 0.91%, and 1.41%, respectively). On average, GraphSAGE-GEBERT outperformed enSapBERT and enCODER by 0.71% and 1.36% Acc@1, respectively. enCODER remains the worst-performing on 3 of 5 datasets. Thus, having a decent performance in zero-shot setting, our proposed GraphSAGE-GEBERT shows superior performance in the biomedical domain after in-domain fine-tuning.

Discussion and Error analysis We looked through erroneous predictions of the fine-tuned GraphSAGE-GEBERT model on the refined test set of the TAC 2017 ADR corpus. Some examples of the model's errors are presented in Tab. 2. After the error analysis, we can draw the following key observations. First, in many cases, the model predicts a concept that is in some relation (e.g., hyponymic or hypernymic) with the true concept. For example, the model marks a mention related to heart rate change as the partial case of it – a heart rate decrease. Second, as can be seen from the examples, the normalization problem with a rich vocabulary poses a great challenge by providing a plethora of distinct but semantically related concepts (such as ‘homicidal attempt‘ and ‘homicidal ideation‘). Thus, in many cases, a true concept and the wrongly predicted one are connected by some relation in the UMLS. We believe that a proper utilization of this relational knowledge is the key to the improvement of normalization quality. Presumably, neither GEBERT nor enCODER fully reveal the power of relational knowledge stored in the UMLS graph. More tricky and effective methods to encode structural knowledge from graphs into LMs are yet to be explored.

5 Conclusion

In this work, we have presented a new model called GEBERT which allows a mutual knowledge exchange between the textual encoder and a graph encoder. We pre-trained two GEBERT models with different state-of-the-art GNN encoders on an English UMLS graph which contains 4M concepts (nodes), 15M textual concept names, and 38.8M relationships (edges). The experimental results on five benchmark datasets in English demonstrate that after task-specific fine-tuning GEBERT outperforms existing state-of-the-art concept normalization models. We consider the following two directions for future work. First, we plan to adopt the proposed model for multilingual pre-training. Second, we plan to infuse relation types at the node neighborhood stage.

Acknowledgements The work has been supported by the Russian Science Foundation grant # 23-11-00358.

References

1. Aronson, A.R.: Effective mapping of biomedical text to the UMLS Metathesaurus: the MetaMap program. In: Proceedings of the AMIA Symposium. p. 17 (2001)
2. Beltagy, I., Lo, K., Cohan, A.: 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). pp. 3615–3620 (2019)
3. Bodenreider, O.: The unified medical language system (umls): integrating biomedical terminology. Nucleic acids research **32**(suppl_1), D267–D270 (2004)
4. Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., Yakhnenko, O.: Translating embeddings for modeling multi-relational data. Advances in neural information processing systems **26** (2013)
5. Chen, H., Chen, W., Liu, C., Zhang, L., Su, J., Zhou, X.: Relational network for knowledge discovery through heterogeneous biomedical and clinical features. Scientific Reports **6**(1), 29915 (2016)
6. Dermouche, M., Looten, V., Flicoteaux, R., Chevret, S., Velcin, J., Taright, N.: ECSTRA-INSERM@ CLEF eHealth2016-task 2: ICD10 code extraction from death certificates. CLEF (2016)
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: 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. pp. 4171–4186 (2019)

8. Doğan, R.I., Leaman, R., Lu, Z.: Ncbi disease corpus: a resource for disease name recognition and concept normalization. Journal of biomedical informatics **47** (2014)
9. Fiorini, N., Canese, K., Starchenko, G., Kireev, E., Kim, W., Miller, V., Osipov, M., Kholodov, M., Ismagilov, R., Mohan, S., et al.: Best match: new relevance search for pubmed. PLoS biology **16**(8), e2005343 (2018)
10. Gillick, D., Kulkarni, S., Lansing, L., Presta, A., Baldridge, J., Ie, E., Garcia-Olano, D.: Learning dense representations for entity retrieval. In: Proceedings of the 23rd Conference on Computational Natural Language Learning. pp. 528–537 (2019)
11. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: International conference on machine learning. pp. 1263–1272. PMLR (2017)
12. Gu, Y., Tinn, R., Cheng, H., Lucas, M., Usuyama, N., Liu, X., Naumann, T., Gao, J., Poon, H.: Domain-specific language model pretraining for biomedical natural language processing. ACM Transactions on Computing for Healthcare **3**(1) (2021)
13. Hamilton, W., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. Advances in neural information processing systems **30** (2017)
14. Lee, J., Yoon, W., Kim, S., Kim, D., Kim, S., So, C.H., Kang, J.: Biobert: pre-trained biomedical language representation model for biomedical text mining. Bioinformatics (09 2019)
15. Lee, S., Kim, D., Lee, K., Choi, J., Kim, S., Jeon, M., Lim, S., Choi, D., Kim, S., Tan, A.C., et al.: Best: next-generation biomedical entity search tool for knowledge discovery from biomedical literature. PloS one **11**(10), e0164680 (2016)
16. Li, J., Sun, Y., Johnson, R.J., Sciaky, D., Wei, C.H., Leaman, R., Davis, A.P., Mattingly, C.J., Wiegers, T.C., Lu, Z.: Biocreative v cdr task corpus: a resource for chemical disease relation extraction. Database **2016** (2016)
17. Liu, F., Shareghi, E., Meng, Z., Basaldella, M., Collier, N.: Self-alignment pretraining for biomedical entity representations. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 4228–4238 (Jun 2021)
18. Liu, F., Vulić, I., Korhonen, A., Collier, N.: Learning domain-specialised representations for cross-lingual biomedical entity linking. In: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing. pp. 565–574 (2021)
19. Lou, Y., Qian, T., Li, F., Zhou, J., Ji, D., Cheng, M.: Investigating of disease name normalization using neural network and pre-training. IEEE Access **8** (2020)
20. Michalopoulos, G., Wang, Y., Kaka, H., Chen, H., Wong, A.: Umlsbert: Clinical domain knowledge augmentation of contextual embeddings using the unified medical language system metathesaurus. In: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1744–1753 (2021)
21. Miftahutdinov, Z., Alimova, I., Tutubalina, E.: On biomedical named entity recognition: Experiments in interlingual transfer for clinical and social media texts. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) **12036 LNCS**, 281–288 (2020)
22. Miftahutdinov, Z., Kadurin, A., Kudrin, R., Tutubalina, E.: Medical concept normalization in clinical trials with drug and disease representation learning. Bioinformatics **37**(21), 3856–3864 (07 2021)
23. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: Advances in neural information processing systems. pp. 3111–3119 (2013)

24. Mondal, I., Purkayastha, S., Sarkar, S., Goyal, P., Pillai, J., Bhattacharyya, A., Gattu, M.: Medical entity linking using triplet network pp. 95–100 (2019)
25. Morgan, A.A., Lu, Z., Wang, X., Cohen, A.M., Fluck, J., Ruch, P., Divoli, A., Fundel, K., Leaman, R., Hakenberg, J., et al.: Overview of biocreative ii gene normalization. Genome biology **9**(S2), S3 (2008)
26. Niu, J., Yang, Y., Zhang, S., Sun, Z., Zhang, W.: Multi-task character-level attentional networks for medical concept normalization. Neural Processing Letters **49**, 1239–1256 (2019)
27. Phan, M.C., Sun, A., Tay, Y.: Robust representation learning of biomedical names. In: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. pp. 3275–3285 (2019)
28. Roberts, K., Demner-Fushman, D., Tonning, J.M.: Overview of the tac 2017 adverse reaction extraction from drug labels track. In: TAC (2017)
29. Soni, S., Roberts, K.: An evaluation of two commercial deep learning-based information retrieval systems for covid-19 literature. Journal of the American Medical Informatics Association **28**(1), 132–137 (2021)
30. Sung, M., Jeon, H., Lee, J., Kang, J.: Biomedical entity representations with synonym marginalization. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 3641–3650 (2020)
31. Sutton, R.T., Pincock, D., Baumgart, D.C., Sadowski, D.C., Fedorak, R.N., Kroeker, K.I.: An overview of clinical decision support systems: benefits, risks, and strategies for success. NPJ digital medicine **3**(1), 17 (2020)
32. Tutubalina, E., Kadurin, A., Miftahutdinov, Z.: Fair evaluation in concept normalization: a large-scale comparative analysis for bert-based models. In: Proceedings of the 28th International Conference on Computational Linguistics. pp. 6710–6716 (2020)
33. Van Mulligen, E., Afzal, Z., Akhondi, S.A., Vo, D., Kors, J.A.: Erasmus MC at CLEF eHealth 2016: Concept recognition and coding in French texts. CLEF (2016)
34. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph Attention Networks. International Conference on Learning Representations (2018), `https://openreview.net/forum?id=rJXMpikCZ`, accepted as poster
35. Wang, X., Han, X., Huang, W., Dong, D., Scott, M.R.: Multi-similarity loss with general pair weighting for deep metric learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5022–5030 (2019)
36. Yang, B., Yih, S.W.t., He, X., Gao, J., Deng, L.: Embedding entities and relations for learning and inference in knowledge bases. In: Proceedings of the International Conference on Learning Representations (ICLR) 2015 (2015)
37. Yuan, Z., Zhao, Z., Sun, H., Li, J., Wang, F., Yu, S.: Coder: Knowledge-infused cross-lingual medical term embedding for term normalization. Journal of biomedical informatics **126**, 103983 (2022)
38. Zhu, M., Celikkaya, B., Bhatia, P., Reddy, C.K.: Latte: Latent type modeling for biomedical entity linking. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 34, pp. 9757–9764 (2020)