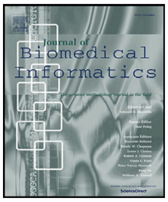
Contents lists available at ScienceDirect

Journal of Biomedical Informatics

journal homepage: www.elsevier.com/locate/yjbin



Original Research

Multimodal learning on graphs for disease relation extraction

Yucong Lin [a,b,1], Keming Lu [c,1], Sheng Yu [d,e], Tianxi Cai [f,g], Marinka Zitnik [g,h,i,*]

[a] *Institute of Engineering Medicine, Beijing Institute of Technology, Beijing, China*
[b] *Beijing Engineering Research Center of Mixed Reality and Advanced Display, School of Optics and Photonics, Beijing Institute of Technology, Beijing, China*
[c] *Viterbi School of Engineering, University of Southern California, Los Angeles, CA, 90007, USA*
[d] *Center for Statistical Science, Tsinghua University, Beijing, China*
[e] *Department of Industrial Engineering, Tsinghua University, Beijing, China*
[f] *Department of Biostatistics, Harvard T.H.Chan School of Public Health, Boston, MA, 02115, USA*
[g] *Department of Biomedical Informatics, Harvard Medical School, Boston, MA, 02115, USA*
[h] *Broad Institute of MIT and Harvard, Boston, MA, 02142, USA*
[i] *Harvard Data Science Initiative, Cambridge, MA, 02138, USA*

ARTICLE INFO

Dataset link: https://github.com/Lukeming-tsinghua/REMOD

Keywords:
Disease relation extraction
Medical relation extraction
Multimodal learning
Knowledge graphs
Graph neural networks
Language models

ABSTRACT

Disease knowledge graphs have emerged as a powerful tool for artificial intelligence to connect, organize, and access diverse information about diseases. Relations between disease concepts are often distributed across multiple datasets, including unstructured plain text datasets and incomplete disease knowledge graphs. Extracting disease relations from multimodal data sources is thus crucial for constructing accurate and comprehensive disease knowledge graphs. We introduce REMAP, a multimodal approach for disease relation extraction. The REMAP machine learning approach jointly embeds a partial, incomplete knowledge graph and a medical language dataset into a compact latent vector space, aligning the multimodal embeddings for optimal disease relation extraction. Additionally, REMAP utilizes a decoupled model structure to enable inference in single-modal data, which can be applied under missing modality scenarios. We apply the REMAP approach to a disease knowledge graph with 96,913 relations and a text dataset of 1.24 million sentences. On a dataset annotated by human experts, REMAP improves language-based disease relation extraction by 10.0% (accuracy) and 17.2% (F1-score) by fusing disease knowledge graphs with language information. Furthermore, REMAP leverages text information to recommend new relationships in the knowledge graph, outperforming graph-based methods by 8.4% (accuracy) and 10.4% (F1-score). REMAP is a flexible multimodal approach for extracting disease relations by fusing structured knowledge and language information. This approach provides a powerful model to easily find, access, and evaluate relations between disease concepts.

1. Introduction

Disease knowledge graphs offer numerous benefits for artificial intelligence (AI) by providing a means to connect, organize, and access diverse data and information resources related to diseases. By incorporating systematized knowledge, AI methods can emulate human expert reasoning, enabling the identification, access, and validation of medical hypotheses. For example, disease knowledge graphs (KGs) power various AI systems, including disease treatment identification [1] and electronic health record (EHR) retrieval [2]. However, constructing high-quality knowledge graphs requires extracting relations between diseases from disparate information sources, such as free-text in biomedical literature and diverse knowledge representations.

Traditional knowledge graphs (KGs) were constructed manually, requiring human input for every fact [3]. While rule-based [4] and semi-automated [5,6] methods are scalable, they can suffer from low accuracy and recall rates. An alternative method is to extract relations from biomedical language, known as relation extraction (RE), which can contribute to building large-scale KGs that comprehensively cover a domain of interest. However, current RE methods require a knowledge-intensive corpus for training language models [7,8], and distant supervision [9] can introduce noise due to heuristic rules [10,11]. Another approach for populating KGs with relations is knowledge graph embeddings (KGE), which are trained on existing, incomplete KGs. However, KGs with large incompleteness may introduce bias in relation extraction, and graph-based methods can suffer from out-of-dictionary problems, limiting the ability to model relations involving entities not previously in the KG [12,13]. Therefore, relying on a single data type for relation extraction may suffer from bias, noise, and incompleteness.

[*] Correspondence to: Department of Biomedical Informatics, Harvard Medical School, 10 Shattuck Street, Boston, MA 02115, USA.
E-mail address: marinka@hms.harvard.com (M. Zitnik).

[1] Equal contribution.

https://doi.org/10.1016/j.jbi.2023.104415
Received 27 July 2022; Received in revised form 24 February 2023; Accepted 30 May 2023
Available online 3 June 2023

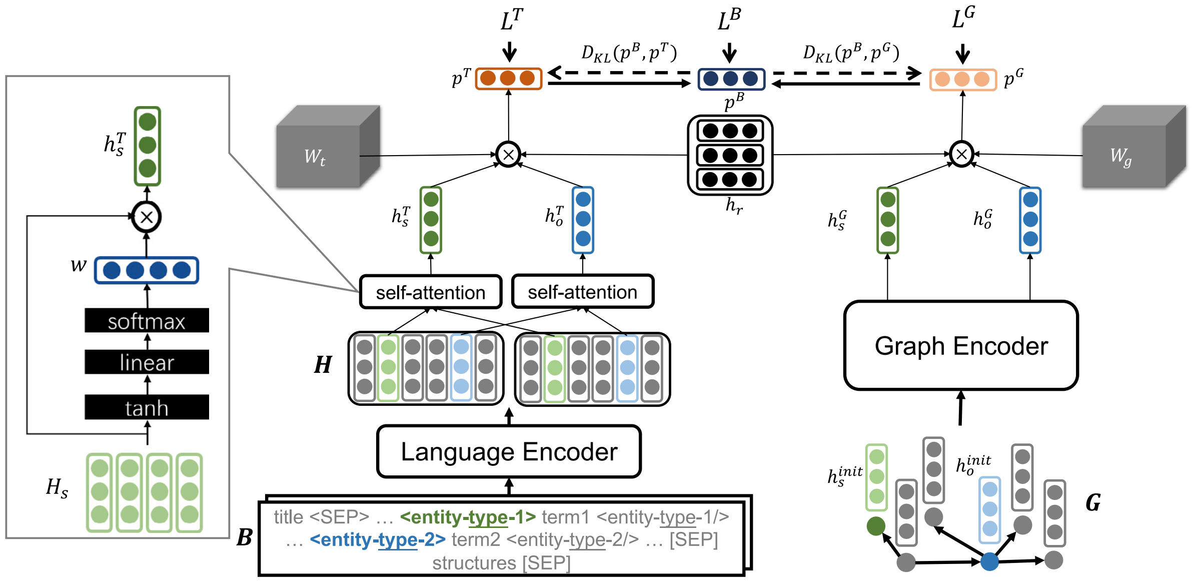


Fig. 1. Overview of REMAP architecture. REMAP introduces a novel co-training learning strategy that continually updates a multimodal language-graph model for disease relation extraction and classification. Language and graph encoders specify deep transformation functions that embed disease concepts (i.e., subject entities s and object entities o) from the language data D^T and disease knowledge graph D^G into compact embeddings, producing condensed summaries of language semantics and biomedical knowledge for every disease. Embeddings output by the encoders (i.e., $\mathbf{h}_s^T$, $\mathbf{h}_o^T$, $\mathbf{h}_o^G$, $\mathbf{h}_s^G$) are then combined in a disease relation type-specific manner (e.g., "differential diagnosis" and "may cause" relation types) and passed to a scoring function that calculates the probability representing how likely two diseases are related to each other and what kind of relationship exists between them.

Both language-based and graph-based methods have their own strengths. Language-based methods can reason over large datasets, such as those generated with distant supervision techniques. On the other hand, graph-based methods can operate on knowledge graphs that have less noise and provide more robust predictions compared to relation extraction. To advance relation extraction, an emerging strategy is to leverage multiple data types simultaneously with multimodal learning [14].

Knowledge graphs (KGs) provide only positive samples, which are examples of true disease–disease relationships. However, existing methods require negative disease pairs, which are disease pairs with unknown or not available (NA) relations. To address this challenge, methods for positive-unlabeled learning [15] and contrastive learning [16,17] can sample random disease pairs from the dataset as negative proxy samples and ensure a low false-positive rate. However, random negative samples may not generalize well in real-world applications because they fail to represent boundary cases. To improve the quality of negative sampling in disease relation extraction, we introduce an EHR-based negative sampling strategy. Our approach generates negative samples using disease pairs that rarely appear together in electronic health records (EHRs). This strategy results in more realistic negative samples, enabling the broad generalization of the approach.

There are many scenarios where only graph or language information is available, but not both modalities. For example, in Lin et al. [18], over 60% of disease pairs in the KG had no corresponding text information, and there were also cases with text but no graph information. Therefore, multimodal approaches must be flexible and able to make predictions when only one data type is available [19]. In this work, we develop a multimodal de-coupled architecture where the language and graph modules interact only through shared parameters and a cross-modal loss function. This approach ensures that our model can take advantage of both language and graph inputs and identify disease relations using either single or multimodal inputs (Fig. 2).

Present work. We introduce REMAP (Relation Extraction with Multimodal Alignment Penalty)[2], a multimodal approach for extracting and classifying disease–disease relations (Fig. 1). REMAP is a flexible algorithm that can jointly learn over language and graphs and can make predictions even under missing modality scenarios. To achieve this, REMAP specifies graph-based and language-based deep transformation functions that embed each data type separately and optimize unimodal embedding spaces to capture the topology of a disease KG and the text semantics of disease concepts. Finally, to fuse both data modalities, REMAP aligns unimodal embedding spaces through a novel alignment penalty loss using the shared disease concepts as anchors. REMAP can effectively model data type-specific distribution and diverse representations while aligning embeddings of distinct data types. Furthermore, REMAP can be jointly trained on both graph and text data types but evaluated and implemented on either of the two modalities alone. Our main contributions are:

- We introduce REMAP, a flexible multimodal approach for extracting and classifying disease–disease relations. REMAP combines knowledge graph embeddings with deep language models and can handle missing data types, which is a critical capability for disease relation extraction.
- We rigorously evaluate REMAP for identifying clinically significant disease–disease relations [20,21]. We create a training dataset using distant supervision and a high-quality test dataset of gold-standard annotations provided by three clinical domain experts. Our evaluations demonstrate that REMAP achieves an 88.6% micro-accuracy and 81.8% micro-F1 score on the human annotated dataset, outperforming text-based methods by 10.0% and 17.2%, respectively. Additionally, REMAP achieves the highest performance of 89.8% micro-accuracy and 84.1% micro-F1 score, surpassing graph-based methods by 8.4% and 10.4%, respectively.

2. Related work

Distantly supervised relation extraction. Distant supervision is a widely-used method for inferring relations by collecting context from a large corpus [9]. Distantly supervised relation extraction takes an incomplete knowledge graph as input and considers a bag of sentences containing an entity pair describing the relations between them. Recent methods leverage pre-trained language models [22,23] and

[2] Python implementation of REMAP is available on Github at https://github.com/Lukeming-tsinghua/REMOD. Our dataset of domain-expert annotations is at https://doi.org/10.6084/m9.figshare.17776865.

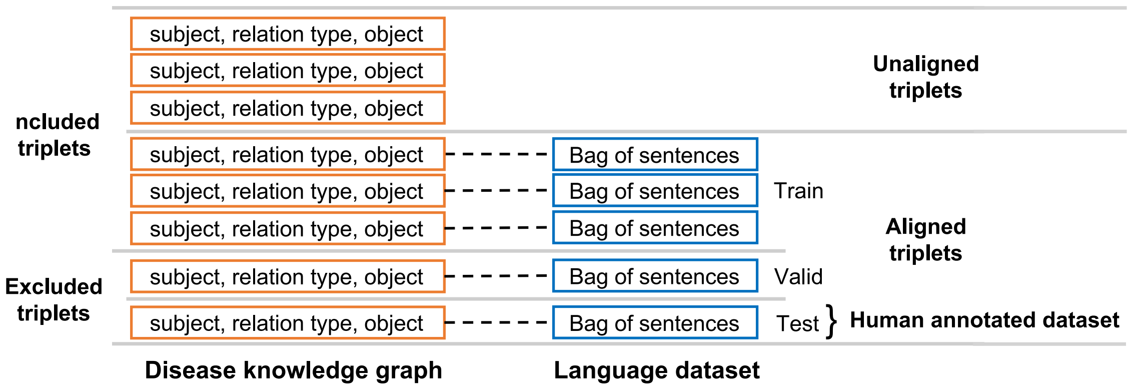
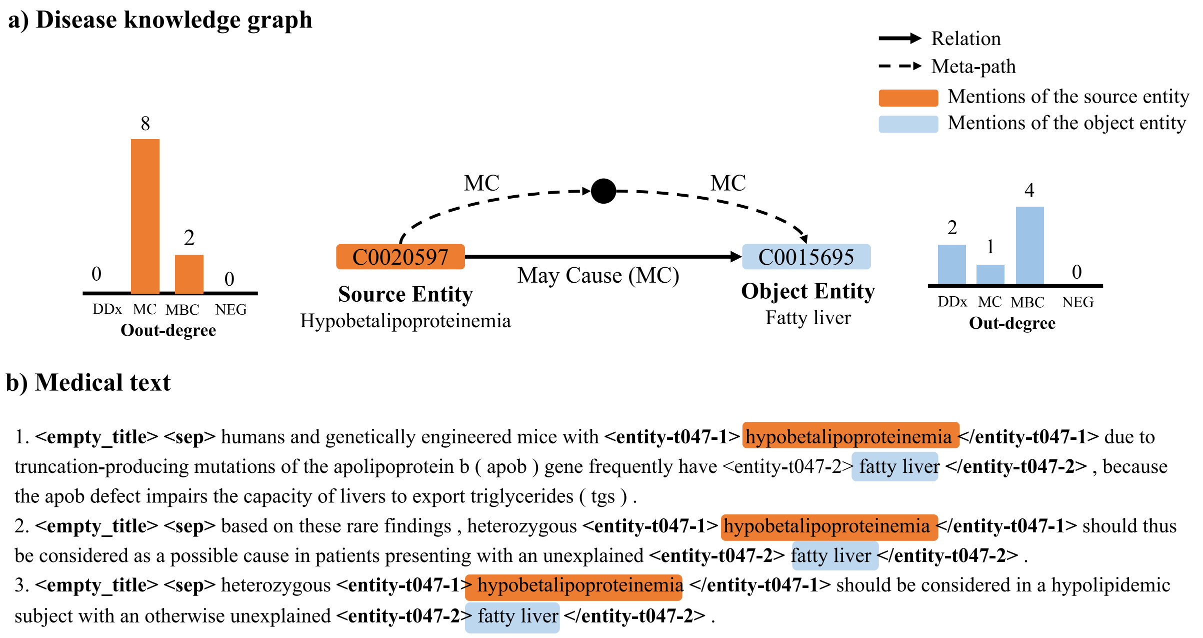

subject, relation type, object
subject, relation type, object
subject, relation type, object
Unaligned triplets
ncluded triplets
subject, relation type, object
Bag of sentences
subject, relation type, object
Bag of sentences
Train
subject, relation type, object
Bag of sentences
Aligned triplets
Excluded triplets
subject, relation type, object
Bag of sentences
Valid
subject, relation type, object
Bag of sentences
Test
Human annotated dataset
Disease knowledge graph
Language dataset

Fig. 2. Creating a dataset split for performance evaluation and benchmarking. Triplets in the disease knowledge graph are divided into aligned triplets and unaligned triplets. *Aligned triplets* are triplets that have corresponding sentences in the language dataset. *Unaligned triplets* have no corresponding sentences in the language dataset.



1. **<empty_title> <sep>** humans and genetically engineered mice with **<entity-t047-1>** hypobetalipoproteinemia **</entity-t047-1>** due to truncation-producing mutations of the apolipoprotein b (apob) gene frequently have <entity-t047-2> fatty liver **</entity-t047-2>** , because the apob defect impairs the capacity of livers to export triglycerides (tgs) .
2. **<empty_title> <sep>** based on these rare findings , heterozygous **<entity-t047-1>** hypobetalipoproteinemia **</entity-t047-1>** should thus be considered as a possible cause in patients presenting with an unexplained **<entity-t047-2>** fatty liver **</entity-t047-2>** .
3. **<empty_title> <sep>** heterozygous **<entity-t047-1>** hypobetalipoproteinemia **</entity-t047-1>** should be considered in a hypolipidemic subject with an otherwise unexplained **<entity-t047-2>** fatty liver **</entity-t047-2>** .

Fig. 3. Illustration of (Hypobetalipoproteinemia, May Cause, Fatty liver) triplet in REMAP. This triplet represents the Hypobetalipoproteinemia has the symptom of fatty liver. The subject entity (Hypobetalipoproteinemia, C0020597) is shown in orange and the object entity (Fatty liver, C0015695) is shown in blue. **(a)** The subject and object entities are identified with unique UMLS concept identifiers. The relation between them is the may cause (MC). There is a MC–MC meta-path between these entities and that information is leveraged by our graph encoder to predict the relationship between Hypobetalipoproteinemia and Fatty liver. The bar plots indicate distribution of relation types going out of the subject or object entities in the knowledge graph. **(b)** We show three sentences representing the triplet. We use *<sep>* token to separate the sentences and the title of article from which we mine the sentence. If the article is from PubMed, we use special token *<empty_title>* as a placeholder. We add two special tokens before and after terms in each sentence to identify the positions of entities. For example, we add *<entity-t047-1>* before *Hypobetalipoproteinemia* to mark the beginning of subject entity; *t047* serves as a identifier of the semantic type — *Disease or Syndrome* — for Hypobetalipoproteinemia. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

have advanced the analysis of biomedical knowledge graphs [7,8,18, 24,25]. We generate the training corpus for REMAP following the strategy in Lin et al. [18]. This distantly supervised corpus is labeled by heuristic rules, which makes it noisy [10,11]. To mitigate this noise, REMAP employs a cross-document alignment penalty during multimodal co-training.

Knowledge graph embedding. Knowledge graph embedding (KGE) methods enable the prediction of new relations in partial and incomplete knowledge graphs. These methods represent each entity (node) and relation (edge) in a graph as a distinct point in a low-dimensional vector space, or embedding. Algebraic operations in this learned space reflect the topology of the graph [26], making KGE methods powerful tools for biomedical applications [27–32]. Examples of widely used KGE methods include translation models [12,30,33,34], bilinear models [35–38], and graph neural networks (GNNs) [39–42]. The latter is the method we employ in this work as the encoder to generate entity representations from existing knowledge graphs. KGE methods enable the completion of sparse knowledge areas and systematic growth of existing KGs through the prediction of new relations using embeddings.

Multimodal learning. Multimodal learning, which involves taking various types of data as input, is widely adopted to improve performance on a variety of tasks [14,33,43–56]. However, these methods are not applicable when there are missing modalities. Further, existing multimodal methods that address this limitation [57–60] are not suitable for handling language and graph modalities. Some studies [61,62] use adversarial learning to impute data from missing modalities, but this approach can introduce unwanted bias, leading to distribution shifts. REMAP takes advantage of multimodal data to enhance disease relation extraction performance, and its decoupled model architecture enables inference on a single data type.

3. Methods

We provide a detailed explanation of the REMAP approach and illustrate it for disease relation extraction (Fig. 1). We begin with a description of the problem definition, an overview of REMAP, and notations as preliminaries (Section 3.1). Then, we provide details of the language and knowledge graph encoders (Section 3.2) and outline the multimodal learning strategy to incorporate knowledge into extraction (Section 3.3).

3.1. Preliminaries

Problem definition. REMAP aims to address the problem of disease relation extraction using both language and graph modalities. Given a pair of diseases, the language input is a bag of sentences that mention these diseases and is collected via distant supervision. The graph input is a knowledge graph that contains these diseases as nodes without any outgoing edges, i.e., there are no known relations between them. REMAP is trained on multimodal data and is expected to infer the disease relationship using either the language or graph input, thereby addressing the issue of missing modalities.

Overview. REMAP is a co-training method that combines language and graph encoders to accurately infer disease relations in both modalities. To achieve this, REMAP utilizes a biomedical pre-trained language model as the language encoder and a heterogeneous attention network as the graph encoder, both selected for their proven performance in an ablation study (Section 6.3).

The encoders provide representations of head and tail entities based on inputs from their respective modalities. Scoring functions, such as the TuckER scoring function, are then used to obtain logits for relation classification in each modality. The encoders from different modalities are first pre-trained separately and then combined with an alignment penalty during co-training (Section 3.3).

REMAP consists of two separate encoders for language and graph modalities, trained on a co-training loss function with an alignment penalty. This approach overcomes labeling noise in each modality by using supervision from both modalities and cross-modal alignment. Furthermore, the decoupled model architecture of REMAP enables accurate inference even when one modality is missing, which is not possible with fusion techniques used in previous multimodal methods such as K-BERT [43], ERNIE [44,45], and CoLAKE [48].

Notations. The training data for REMAP consists of language information $D^T = \{B_i\}_{i=1}^{M_0} = D_L^T \cup D_U^T = \{B_i\}_{i=1}^{M} \cup \{B_i\}_{j=M+1}^{M_0}$ given as M_0 bags of sentences and graph information $D^G = \{(s_i, r_i, o_i)\}_{i=1}^{M} \bigcup \{(s_i, r_i, o_i)\}_{i=M+1}^{N}$ given as N triplets (s_i, r_i, o_i) encoding the relationship between s_i and o_i as r_i. The subscript i simply denotes the index of a triplet in D^G. Head or tail entities with different subscripts can be referred to as the same entities. For example, s_i = "Hypobetalipoproteinemia", o_i = "fatty liver" and r_i = "May Cause" would indicate the fatty liver would be a possible symptom of hypobetalipoproteinemia. We assume that M bags of sentences in D^T overlap with the triplets from existing KG such that each sentence of the ith sentence bag contain (s_i, o_i). The remaining $N - M$ triplets in D^G cannot be mapped to sentences in D^T, and $M_0 - M$ sentences contain entity pairs that do not belong to existing KG. We represent the ith sentence bag as $B_i = \{(t_{ij}, I_{ij}^s, I_{ij}^o)\}_{j=1}^{l_i}$, where l_i is the number of sentences in bag B_i, t_{ij} is the tokenized sequence of jth sentence in B_i. Here, the tokenized sequence combines the mentions of subject and object entities, entity markers, the document title, and the article structure. Marker tokens are added to each entity's head and tail positions to denote entity type information. Last, I_{ij}^s and I_{ij}^o are the start indices of entity markers for subject and object entities, respectively.

3.2. Language and knowledge graph encoders

Embedding disease-associated sentences (Language encoder). We start by tokenizing entities in sentences and proceed with an overview of the language encoder. Entity tokens identify the position and type of entities in a sentence [7,8]. Specifically, tokens ⟨S-type⟩ and ⟨S-type⟩, and ⟨O-type⟩ and ⟨O-type⟩ are used to denote the start and end of subject (S) and object (O) entities, respectively. The entity marker tokens are type-related, meaning that entities of different types (e.g., disease concepts, medications) get different tokens. This procedure produces bags of tokenized sentences D^T that we encode into entity embeddings using a language encoder. We use SciBERT encoder [23] with the SciVocab vocabulary, which is a BERT language model optimized for scientific text with improved efficiency in biomedical domains than BioBERT or BERT model alone [8]. Tokenized sequences in a sentence bag B_i are fed into the language model to produce a set of sequence outputs $\mathbb{H}_i = \{\mathbf{H}_i\}_{i=1}^{l_i}, i = 1, 2, \dots, l_i$:

$$\mathbb{H}_i^{[m]} = \mathrm{SciBERT}(B_i; \mathbb{H}_i^{[m-1]}), \tag{1}$$

where $\mathbb{H}_i = [\mathbf{H}_{i1}, \dots, \mathbf{H}_{il_i}]$, l_i is the number of sentences in B_i, $\mathbf{H}_i \in \mathbb{R}^{d_l \times d_{hs}}$. We then aggregate representations of subject entities s_i across all sentences in bag B_i as: $\mathbf{H}_{s_i} = ||_{m=1}^{l_i} ||_{k \in I_{ij}^s} \mathbf{H}_{mk}$ and use self-attention to obtain the final language-based embedding $\mathbf{h}_{s_i}^T$ for subject entity s_i as:

$$\mathbf{h}_{s_i}^T = \mathbf{H}_{s_i} \cdot \mathrm{softmax}(\boldsymbol{\omega} \cdot \tanh(\mathbf{H}_{s_i})), \tag{2}$$

where $\mathbf{h}_{s_i}^T \in \mathbb{R}^{d_{hs}}$ is the embedding of s_i and $\boldsymbol{\omega}$ is a trainable vector. Embeddings of object entities (i.e., $\mathbf{h}_{o_i}^T$ for object entity o_i) are generated analogously by the language encoder. Self-attention is needed because some sentences in a bag may not contain relevant disease–disease relationships. The attention mechanism allows the model to down-weight those uninformative sentences when generating embeddings.

Embedding disease–disease knowledge relations (Graph encoder). We use a heterogeneous graph attention encoder(HAN) [41] to derive embeddings for nodes in the disease knowledge graph. Heterogeneous graph attention network (HAN) [41] is a graph neural network to embed a KG by leveraging meta paths. A meta path is a sequence of node types and relation types [63]. For example, in a disease KG, "Disease" → "May Cause" → "Disease" → "Differential Diagnosis" → "Disease" is a meta path. Node u_i is connected to node u_j via a meta path $\boldsymbol{\Phi}$ if u_i and u_j are the head and tail nodes, respectively, of this meta path. Each node u_i has an initial node embedding $\mathbf{h}_i^{\text{init}}$ and belongs to a type ϕ_i, e.g., ϕ_i = "disease concept". Graph attention network specified a parameterized deep transformation function f that maps nodes to condensed data summaries, i.e., embeddings, in a node-type specific manner as $\mathbf{h}_i' = f^{\phi_i}(\mathbf{h}_i^{\text{init}})$.

We denote all nodes adjacent to u_i via a meta-path $\boldsymbol{\Phi}$ as $u_j \in N_i^{\boldsymbol{\Phi}}$ and node-level attention mechanism provides information on how strongly $\mathbf{h}_i'$ attends to u_i's each adjacent node u_j when generating the embedding for u_i. In particular, the importance of u_j for u_i in meta path $\boldsymbol{\Phi}$ is defined as:

$$a_{ij}^{\boldsymbol{\Phi}} = \frac{\exp(\sigma(\mathbf{a}_{\boldsymbol{\Phi}}^T \cdot [\mathbf{h}_i' || \mathbf{h}_j']))}{\sum_{k \in N_i^{\boldsymbol{\Phi}}} \exp(\sigma(\mathbf{a}_{\boldsymbol{\Phi}}^T \cdot [\mathbf{h}_i' || \mathbf{h}_k']))}, \tag{3}$$

where σ is the sigmoid activation, $||$ indicates concatenation, and $\mathbf{a}_{\boldsymbol{\Phi}}$ is a trainable vector. To promote stable attention, HAN uses multiple, i.e., K, heads, and concatenates K vectors after node level attention to produce the final node embedding for node u_i. Denoting $a_{ij,k}^{\boldsymbol{\Phi}}$ as the kth attention head in meta-path $\boldsymbol{\Phi}$, the multi-head attention is calculated as followed:

$$\mathbf{z}_i^{\boldsymbol{\Phi}} = ||_{k=1}^{K} \sigma(\sum_{j \in N_i^{\boldsymbol{\Phi}}} a_{ij,k}^{\boldsymbol{\Phi}} \cdot \mathbf{h}_j'), \tag{4}$$

Given user-defined meta paths $\boldsymbol{\Phi}_1, \dots, \boldsymbol{\Phi}_P$, HAN uses the above specified node-level attention to produce node embeddings $\mathbf{Z}_{\boldsymbol{\Phi}_1}, \dots, \mathbf{Z}_{\boldsymbol{\Phi}_P}$. Finally, HAN uses semantic-level attention to combine meta path-specific node embeddings as:

$$\beta_{\boldsymbol{\Phi}_p} = \frac{\exp(\frac{1}{|E|} \sum_{i \in E} \mathbf{q}^T \cdot \tanh(\mathbf{W} \cdot \mathbf{z}_i^{\boldsymbol{\Phi}_p} + b))}{\sum_{p=1}^{P} \exp(\frac{1}{|E|} \sum_{i \in E} \mathbf{q}^T \cdot \tanh(\mathbf{W} \cdot \mathbf{z}_i^{\boldsymbol{\Phi}_p} + b))}, \tag{5}$$

$$\mathbf{H}^G = \sum_{p=1}^{P} \beta_{\Phi_p} \cdot \mathbf{Z}_{\Phi_p}, \tag{6}$$

where β_{Φ_p} represents the importance of meta path Φ_p towards final output embeddings $\mathbf{H}^G$, a compact matrix of knowledge associated with nodes in the KG., and $\mathbf{q}^T$, $\mathbf{W}$, and b are trainable parameters. For simplicity, we use a function $h_e^G = f_{\text{HAN}}(D^G, \mathbf{H}^{\text{init}}, e)$ denoting the forward process of HAN. The inputs are the graph, the matrix of initial embeddings, and an entity, and the output h_e^G is the output node representation of entity e.

We employ all one-hop and two-hop relation paths as our meta-paths in HAN. For example, if we have two relations r_1 and r_2 in our KG. We use $\{r_1, r_2, (r_1, r_2), (r_2, r_1)\}$ as our meta-paths. We have not involved more than two hops meta-paths because of the fast-increasing computing needs for extended computation. The encoder produces embeddings for every subject entity $\mathbf{h}_{s_i}^G$ and every object entity $\mathbf{h}_{o_i}^G$ that have corresponding nodes in the KG as follows:

$$\mathbf{h}_{s_i}^G = f_{\text{HAN}}(D^G, \mathbf{H}^{\text{init}}, s_i), \quad \mathbf{h}_{o_i}^G = f_{\text{HAN}}(D^G, \mathbf{H}^{\text{init}}, o_i). \tag{7}$$

Scoring functions. Taking language-based embeddings, $\mathbf{h}_{s_i}^T, \mathbf{h}_{o_i}^T$, and graph-based embeddings, $\mathbf{h}_{s_i}^G, \mathbf{h}_{o_i}^G$, for diseases that appear in either language or graph dataset, REMAP scores triplets (s_i, r_k, o_i) as candidate disease–disease relationships. Specifically, to estimate the probability that diseases s_i and o_i are associated through a relation of type r_k (e.g., r_k = "May Cause"), REMAP calculates scores p^T and p^G representing the amount of evidence in the combined language-graph dataset that supports the disease–disease relationship:

$$p^T(r_{ik} = 1|s_i, o_i) = \text{SF}(\mathbf{h}_{s_i}^T, \mathbf{h}_{o_i}^T, \mathbf{h}_r),\ k = 1, 2, \dots, K, \tag{8}$$

$$p^G(r_{ik} = 1|s_i, o_i) = \text{SF}(\mathbf{h}_{s_i}^G, \mathbf{h}_{o_i}^G, \mathbf{h}_r),\ k = 1, 2, \dots, K, \tag{9}$$

where SF is the scoring function, and K denotes the number of relation types. We consider three scoring functions, including the linear scoring function: $\text{SF}_{\text{Li}}(\mathbf{h}_{s_i}, \mathbf{h}_{o_i}, \mathbf{h}_{r_k}) = \sigma(\mathbf{W}_k(\mathbf{h}_{s_i} + \mathbf{h}_{o_i}) + b_k)$, the TransE scoring function [12]: $\text{SF}_{\text{Tr}}(\mathbf{h}_{s_i}, \mathbf{h}_{o_i}, \mathbf{h}_{r_k}) = \sigma(\|\mathbf{h}_{s_i} + \mathbf{h}_{r_k} - \mathbf{h}_{o_i}\|_2^2)$, and the TuckER scoring function [38]: $\text{SF}_{\text{Tu}}(\mathbf{h}_{s_i}, \mathbf{h}_{o_i}, \mathbf{h}_{r_k}, \mathbf{W}) = \sigma(\mathbf{W} \times_1 \mathbf{h}_{s_i} \times_2 \mathbf{h}_{r_k} \times_3 \mathbf{h}_{o_i})$, where separate kernels $\mathbf{W}^T$ and $\mathbf{W}^G$ are used for language and graph in the TuckER decomposition, and $\mathbf{h}_{r_k}$ denotes the encoding of relation type r_k in TransE that is shared across both modalities.

3.3. Co-training language and graph encoders

We proceed to describe the procedure for co-training language and graph encoders. From the last section, we obtain relationship estimates based on evidence provided by language information, $p^T(\hat{r}_{ik} = 1|s_i, o_i)$, and graph information, $p^G(\hat{r}_{ik} = 1|s_i, o_i)$, for every triplet (s_i, r_i, o_i). Finally, we use the binary cross entropy to optimize those estimates in each data type as:

$$L^T = \sum_{i=1}^{M} \sum_{k=1}^{K} r_{ik} \log(p^T(\hat{r}_{ik} = 1|s_i, o_i)) + (1 - r_{ik}) \log(1 - p^T(\hat{r}_{ik} = 1|s_i, o_i)), \tag{10}$$

$$L^G = \sum_{i=1}^{M} \sum_{k=1}^{K} r_{ik} \log(p^G(\hat{r}_{ik} = 1|s_i, o_i)) + (1 - r_{ik}) \log(1 - p^G(\hat{r}_{ik} = 1|s_i, o_i)), \tag{11}$$

and can combine language-based loss L^T and graph-based loss L^G as: $L_{\text{REMAP}} = L^T + L^G$.

On top of this joint learning loss function, we develop two REMAP variants, REMAP-M and REMAP-B, based on how language-based and graph-based losses are combined into a multimodal objective. These loss functions are motivated by the principle of knowledge distillation [64] to enhance multimodal interaction and improve classification performance. Using probabilities $p^T(\hat{r}_{ik} = 1|s_i, o_i)$ from the language encoder, we normalize them across K relation types to obtain distribution $\mathbf{p}_t(s_i, o_i)$ as:

$$\mathbf{p}^T(s_i, o_i) = \{\frac{e^{p^T(\hat{r}_{ik}=1|s_i,o_i)}}{\sum_{m=1}^{K} e^{p^T(\hat{r}_{im}=1|s_i,o_i)}}\}_{k=1}^{K}. \tag{12}$$

Similarly, we calculate a graph-based distribution $\mathbf{p}^G(s_i, o_i)$ using softmax normalization. In REMAP-M, prediction logits from different modalities are aligned by shrinking the distance between distributions $\mathbf{p}^T$ and $\mathbf{p}^G$ using the Kullback–Leibler (KL) divergence:

$$D_{\text{KL}}(\mathbf{p}^T, \mathbf{p}^G) = \sum_{i=1}^{M} \sum_{k=1}^{K} p^G(s_i, o_i)_k \log(\frac{p^G(s_i, o_i)_k}{p^T(s_i, o_i)_k}), \tag{13}$$

where we measure the misalignment between language and graph models as follows:

$$L_{\text{REMAP-M}} = L_{\text{REMAP}} + \lambda_M(D_{\text{KL}}(\mathbf{p}^T, \mathbf{p}^G) + D_{\text{KL}}(\mathbf{p}^G, \mathbf{p}^T)). \tag{14}$$

Instead of measuring how distribution $\mathbf{p}^T$ is different from $\mathbf{p}^G$, REMAP-B selects the strongest logit across data types using an ensemble distillation strategy [65]. In short, the lower score from language and graph encoders is selected if the relation label is negative and vice-versa. REMAP-B produces the final prediction p^B as:

$$p^B(\hat{r}_{ik} = 1|s_i, o_i) \tag{15}$$

$$= \begin{cases} p^T(\hat{r}_{ik} = 1|s_i, o_i), & \text{if } p^T(\hat{r}_{ik} = 1|s_i, o_i) \leq p^G(\hat{r}_{ik} = 1|s_i, o_i) \text{ and } r_{ik} = 0 \\ p^T(\hat{r}_{ik} = 1|s_i, o_i), & \text{if } p^T(\hat{r}_{ik} = 1|s_i, o_i) \geq p^G(\hat{r}_{ik} = 1|s_i, o_i) \text{ and } r_{ik} = 1 \\ p^G(\hat{r}_{ik} = 1|s_i, o_i), & \text{if } p^T(\hat{r}_{ik} = 1|s_i, o_i) \geq p^G(\hat{r}_{ik} = 1|s_i, o_i) \text{ and } r_{ik} = 0 \\ p^G(\hat{r}_{ik} = 1|s_i, o_i), & \text{if } p^T(\hat{r}_{ik} = 1|s_i, o_i) \leq p^G(\hat{r}_{ik} = 1|s_i, o_i) \text{ and } r_{ik} = 1 \end{cases} \tag{16}$$

that are softmax-normalized across K relation types:

$$\mathbf{p}^B(s_i, o_i) = \{\frac{e^{p^B(\hat{r}_{ik}=1|s_i,o_i)}}{\sum_{m=1}^{K} e^{p^B(\hat{r}_{im}=1|s_i,o_i)}}\}_{k=1}^{K}. \tag{17}$$

Finally, to minimize the discrepancy between predicted and known disease–disease relationships, REMAP-B uses minimizes the cross-entropy function:

$$L^B = \sum_{i=1}^{M} \sum_{k=1}^{K} r_{ik} \log(p^B(\hat{r}_{ik} = 1|s_i, o_i)) + (1 - r_{ik}) \log(1 - p^B(\hat{r}_{ik} = 1|s_i, o_i)), \tag{18}$$

and co-trains the multimodal encoder by promoting predictions that are aligned by both encoders:

$$L_{\text{REMAP-B}} = L_{\text{REMAP}} + \lambda_B[L^B + D_{\text{KL}}(\mathbf{p}^B, \mathbf{p}^T) + D_{\text{KL}}(\mathbf{p}^B, \mathbf{p}^G)]. \tag{19}$$

3.4. Training REMAP models

We first pretrain the language and graph encoders and their corresponding scoring modules separately on each modality before the multimodal co-training process (Section 3.2). The text-based model comprises the language encoder and the TuckER module. We denote the relation embeddings in this TuckER module as $\mathbf{r}^T$, and the corresponding loss function is denoted as L^T (Eq. (10)).

On the other hand, the graph-based model comprises the heterogeneous attention network encoder and another TuckER module. The relation embeddings produced by the TuckER module are denoted as $\mathbf{r}^G$. In the pre-training phase, the model is optimized for the loss function L^G (Eq. (11)).

After the pre-training is completed, the language and graph models are fused through cross-modal learning (Section 3.3). To achieve this, the shared relation vector $\mathbf{r}$ is initialized as the average of the relation vectors from both modalities, i.e., $\mathbf{r} = (\mathbf{r}^T + \mathbf{r}^G)/2$.

The complete REMAP-B training algorithm is outlined in Algorithm 1.

Algorithm 1: REMAP, multimodal learning on graphs for disease relation extraction and classification. Shown is the outline of REMAP-B (Section 3.3).

```
Input: Bag of sentences $\{B_i\}_{i=1}^{n}$, Knowledge graph $D^G = \{(s_i, r_i, o_i)\}$,
       Initial node embeddings $\mathbf{H}^{\text{init}}$, Scoring function SF,
       Regularization strength $\lambda_B$, Relation type vector pretrained on
       language dataset $\mathbf{r}^T$, Relation type vector pretrained on graph
       dataset $\mathbf{r}^G$ (Pretraining of relation type vectors are described in
       Section 3.4)
Output: Model parameters $\Theta$ of language-based and graph-based
        encoders
Initialize model parameters $\Theta$
Initialize relation representation as $\mathbf{r} = \frac{\mathbf{r}^T + \mathbf{r}^G}{2}$
for epoch=1:n_epochs do
    for i=1:n_bags do
        /* Encode language and the calculation logits, Section 3.2 */
        $\mathbf{H} = \text{SciBERT}(B_i)$
        $\mathbf{H}_{s_i} = ||_{m=1}^{l_i} ||_{k \in I_{ij}^s} \mathbf{H}_{mk}$
        $\mathbf{H}_{o_i} = ||_{m=1}^{l_i} ||_{k \in I_{ij}^o} \mathbf{H}_{mk}$
        $\mathbf{h}_{s_i}^T = \mathbf{H}_{s_i} \cdot \text{softmax}(\boldsymbol{\omega} \cdot \tanh(\mathbf{H}_{s_i}))$
        $\mathbf{h}_{o_i}^T = \mathbf{H}_{o_i} \cdot \text{softmax}(\boldsymbol{\omega} \cdot \tanh(\mathbf{H}_{o_i}))$
        $\mathbf{p}^T(s_i, o_i) = \text{SF}(\mathbf{h}_{s_i}^T, \mathbf{h}_{o_i}^T, \mathbf{r})$
        $L^T = \text{BinaryCrossEntropyLoss}(\mathbf{p}^T(s_i, o_i), \mathbf{r}(s_i, o_i))$
        /* Encode graph and calculate graph logits, Section 3.2 */
        $\mathbf{h}_{s_i}^G = \text{HAN}(G, \mathbf{H}^{init}, s_i)$
        $\mathbf{h}_{o_i}^G = \text{HAN}(G, \mathbf{H}^{init}, o_i)$
        $p^G(s_i, o_i) = \text{SF}(\mathbf{h}_{s_i}^G, \mathbf{h}_{o_i}^G, \mathbf{r})$
        $L^G = \text{BinaryCrossEntropyLoss}(\mathbf{p}^G(s_i, o_i), \mathbf{r}(s_i, o_i))$
        /* Find the best logit and calculate alignment penalty loss,
         Section 3.3 */
        $\mathbf{p}^B(s_i, o_i) = \text{CalculateBestLogic}(\mathbf{p}^T(s_i, o_i), \mathbf{p}^G(s_i, o_i))$
        $\mathbf{p}^T(s_i, o_i) = \text{softmax}(\mathbf{p}^T(s_i, o_i))$
        $\mathbf{p}^G(s_i, o_i) = \text{softmax}(\mathbf{p}^G(s_i, o_i))$
        $\mathbf{p}^B(s_i, o_i) = \text{softmax}(\mathbf{p}^B(s_i, o_i))$
        $L^B = \sum_{i=1}^{M} \sum_{k=1}^{K} r_{ik} \log(p^B(r_{ik} = 1 | s_i, o_i)) + (1 - r_{ik}) \log(1 - p^B(r_{ik} =$
         $1 | s_i, o_i))$
        $L_{\text{REMAP-B}} = L^T + L^G + \lambda_B [L^B + D_{\text{KL}}(\mathbf{p}^B, \mathbf{p}^T) + D_{\text{KL}}(\mathbf{p}^B, \mathbf{p}^G)]$
        $\Theta \leftarrow \text{Update}(\Theta, L_{\text{REMAP-B}})$
```

4. Experiments

We proceed with the description of datasets (Section 4.1), followed by implementation details of REMAP approach (Section 4.2) and the outline of experimental setup (Section 4.3).

4.1. Dataset

The datasets used in this study are multimodal and were obtained from various sources that we integrated and harmonized. They include a disease knowledge graph that was aggregated from multiple ontologies, a text corpus that was annotated using distant supervision, entity co-occurrences from electronic health records, and a validation dataset that was annotated by humans for relation classification.

Disease knowledge graph. We created a disease–disease KG by aggregating data from the Diseases database [66] and the MedScape [67] repository. This KG contains evidence on disease–disease associations derived from text mining, manually curated literature, cancer mutation data, and genome-wide association studies. Following the approach outlined in Lin et al. [18], we constructed a KG between disease concepts, which includes 9,182 disease concepts that are assigned concept unique identities (CUI) in the Unified Medical Language System (UMLS). The KG includes three types of relationships between disease concepts: 'may cause' (MC), 'may be caused by' (MBC), and 'differential diagnosis' (DDx), with other relations in the KG denoted as 'not available' relations (NA). The MBC relation type is the reverse relation of the MC relation type, while DDx is a symmetric relation between disease concepts. We involved MBC as one of our relation classes in relation classification, as it allows us to keep consistency between the problem definition and our disease knowledge graph structure. To achieve this, we constructed our disease knowledge graph as a heterogeneous directional graph, introducing MBC edges as the inverse edges of MB. Our knowledge graph contains only one node type (disease) and three different edges representing the three relations between diseases. Further details on the dataset statistics are available in [18] and Table 1.

Language dataset. We used a text corpus from Lin et al. [18], which was built from 42 million Pubmed abstracts[3], web pages from Uptodate and Medscape eMedicine, Wikipedia articles with medical titles, and the main text of four textbooks[4]. The corpus was segmented into sentences, resulting in 237,119,572 sentences. To compile a large disease–disease KG, we retrieved text descriptions from medical data repositories using distant supervision, following the data collection and preprocessing strategy described in Lin et al. [18]. We used forward maximum matching to identify all UMLS disease concepts in the corpus and linked triplets in our disease knowledge graph to sentences if subject and object entities were both in the sentences. This method aligned 1,466,065 sentences from these articles to disease–disease edges in the disease knowledge graph. Finally, we grouped sentences by triplets, resulting in a bag of sentences for each triplet. For example, the triplet *(Hypobetalipoproteinemia, may cause, Fatty liver)* was aligned to a bag of sentences containing four sentences. More statistical details are shown in Table 1, and further information on data preprocessing and feature engineering can be found in Appendix A.

Electronic health record dataset. We utilize two types of information from EHRs, both obtained from Beam et al. [68]. Specifically, we use a dataset of concept co-occurrences from 20 million notes collected at Stanford. Using this dataset, Beam et al. [68] created a dataset of disease concepts identified with SNOMED-CT that appear together in the same note. They used singular value decomposition (SVD) to decompose the resulting co-occurrence matrix and generate 500-dimensional embedding vectors for disease concepts. The co-frequencies were computed using a parallelized annotation, hashing, and counting pipeline applied over clinical notes from Stanford Hospitals and Clinics. We use the information on co-occurring disease concepts to guide the sampling of negative node pairs during the training of the disease knowledge graph. Moreover, we initialize the embeddings in REMAP's graph neural network using the 500-dimensional embedding vectors from [68]. For out-of-dictionary disease concepts, we use the average of all concept embeddings as their initial embeddings.

Human annotated dataset. Relation triplets retrieved from databases may still contain some incorrect data. To create a robust evaluation of our model, we randomly selected 500 disease triplets from our disease knowledge graph and split the corresponding sentences in the language dataset into sentences in the annotated dataset. We omitted this subgraph, along with meta paths, from the knowledge graph used for model training and created an annotated dataset with it. We recruited three senior physicians from Peking Union Medical College Hospital and asked them to independently assign candidate relations (MC, MBC, DDx, and NA) to these entity pairs without showing them

[3] Downloaded from The PubMed Baseline Repository provided by the National Library of Medicine, January 2021

[4] *Harrison's Principles of Internal Medicine 20th Edition, Kelley's Textbook of Internal Medicine 4th Edition, Sabiston Textbook of Surgery: The Biological Basis of Modern Surgical Practice 20th Edition, and Kumar and Clark's Clinical Medicine 7th Edition*

Table 1
Overview of the disease knowledge graph and the language dataset. Shown are statistics for the following relation types: Not Available (NA), Differential Diagnosis (DDx), May Cause (MC), and May Be Caused by (MBC). Total denotes the total number of all triplets (Fig. 2). Total KG Triplets are the summation of unaligned and aligned triplets..

Dataset			Total	NA	DDx	MC	MBC	Entities
Unaligned	–	–	96,913	30,546	20,657	23,411	22,298	9,182
Aligned	Train	Triplet	31,037	15,670	7,262	4,358	3,747	7,794
		Language	1,244,874	799,194	208,921	123,735	113,024	
	Validation	Triplet	7,754	3,918	1,821	1,065	950	4,433
		Language	206,179	68,934	60,165	43,706	33,474	
	Annotated	Triplet	500	8	210	160	122	733
		Language	15,012	96	4,980	6,699	3,237	
Total KG Triplets			136,204	50,142	29,950	28,994	27,117	–

relations collected from the database. We found that annotation experts disagreed on the labels for only 14 disease pairs (*i.e.*, 2.8% of the total number of disease pairs), and we resolved these disagreements through consensus voting. The human annotated dataset is used for model comparison and performance evaluation.

Data denoising. Our dataset was built using a distantly supervised framework [9]. This method annotates any sentences containing two entities with a specific relation between them. However, the distantly supervised approach only follows the at-least-one assumption, meaning it assumes that at least one of the sentences collected for a pair will describe the target relation between them. As a result, the distantly supervised corpus we used is noisy because sentences that contain two diseases may also include another relationship beyond MC, MBC, and DDx. All these noisy sentences are kept in our corpus. This is one of the main motivations for proposing a multimodal joint learning method: no data in a single modality is perfectly clean. The language data is noisy due to the naive assumptions of a distantly supervised framework, while the graph data is incomplete. Learning from the alignment between different modalities will help to avoid learning noise in the training process.

4.2. Implementation details

Next, we outline the implementation details of REMAP models, including negative sampling and encoders.

Negative sampling. To construct negative samples, we follow a hybrid method. First, we sample disease pairs that are not connected by any meta-paths in our disease knowledge graph. Then, from these candidates, we further sample pairs whose co-occurrence in the EHR co-occurrence matrix [68] is below a predefined threshold. We expect that unrelated diseases are unlikely to appear together in EHRs, so their corresponding values in the co-occurrence matrix should be low. By using this hybrid negative sampling approach, we aim to further reduce false negative rates and improve the performance of our models in classifying disease–disease relations.

Language encoder. We utilize the SciBERT tokenizer and SciBERT-SciVocab-uncased model [69] as the foundation of our language encoder. We add the entity markers to the SciVocab vocabulary and initialize their embeddings using a uniform distribution. The maximum number of sentences in a bag is set to $l_{m(max)}$. If the bag size exceeds $l_{m(max)}$, then $l_{m(max)}$ sentences are randomly selected for training.

Graph encoder. We use HAN to encode the disease knowledge graph. The initial embeddings for nodes in the disease knowledge graph are unique concept identifier (CUI) representations derived from the SVD decomposition of the EHR co-occurrence matrix [68].

Hyper-parameters. We use grid search on the validation set to select hyper-parameters for all methods. Table A4 lists REMAP's hyper-parameters.

4.3. Experimental setup

Baseline methods. We consider twelve baseline methods split into three groups: five methods for link prediction on KGs, five text-based methods for relation extraction, and two multimodal approaches.

For graph-based baselines, we take both disease and relation embeddings as parameters. We randomly initialize the embeddings and train the knowledge graph models or graph neural networks. Graph-based baselines are trained on the disease KG and include the following:

- **TransE** [12] embeds diseases and relations by translating embedding vectors in the learned embedding space. Given embeddings of a triplet $(\mathbf{h}, \mathbf{r}, \mathbf{t})$, the score is calculated as $s = ||\mathbf{h}+\mathbf{r}-\mathbf{t}||_2^2$. We train our TransE model using negative sampling and $L2$ penalty as recommended by the authors. The negative samples are constructed by randomly replacing objects $\mathbf{t}$ given subject $\mathbf{h}$ and relation $\mathbf{r}$. We begin with random embeddings for diseases and relations and then optimize them with the same margin-based ranking criterion and stochastic gradient descent as in the original TransE paper.
- **DistMult** [70] predicts disease–disease relationships using a bilinear decoder for edge in the KG. Given embeddings of a triplet $(\mathbf{h}, \mathbf{r}, \mathbf{t})$, the score is calculated as $s = \mathbf{h}\mathbf{M}_r\mathbf{t}$, where $\mathbf{M}_r$ is a trainable diagonal matrix for relation r. The disease and relation embeddings are parameters in this baseline. We initialize it with random embeddings and optimize them with margin-based ranking loss and batch stochastic gradient descent. The negative samples are constructed by corrupting the subject or object in a relation triplet.
- **ComplEx** [71] predicts disease–disease relationships by carrying out a matrix factorization using complex-valued embeddings. Denoting the complex-valued embeddings of a relation triplet as $\mathbf{h}_s$, $\mathbf{w}_r$, and $\mathbf{h}_o$. ComplEx assumes $P(Y = 1) = \sigma(\phi(\mathbf{h}_s, \mathbf{w}_r, \mathbf{h}_o))$, where $Y = 1$ represents the triplet (h, r, t) holds and σ denotes the sigmoid function. The score function $\phi(\cdot)$ is calculated as $s = Re(\langle \mathbf{h}, \mathbf{r}, \mathbf{t} \rangle)$. The product $\langle \cdot \rangle$ is a Hermitian product, and relation embedding $\mathbf{r}$ is a complex-valued vector. We take random embeddings as initial embeddings for both real and imaginary parts. The training object is minimizing the negative log-likelihood of this logistic model with L_2 penalty on the parameters of disease and relation embeddings.
- **RGCN** [72] predicts disease–disease relationships using relational graph convolutional network (RGCN). Following the authors' recommendations, we use a 2-layer RGCN to embed the KG and DistMult decoder for link prediction. We train the RGCN model using the cross-entropy loss on four relations and the Adam optimizer. The RGCN uses the sigmoid activation function and has a 0.4 dropout rate for self-loops and 0.2 for other edges.
- **TuckER** [13] is a KG embedding method that uses Tucker tensor decomposition. Given the embeddings of a triplet $(\mathbf{h}, \mathbf{r}, \mathbf{t})$, the score is calculated as $s = \mathbf{W} \times_1 \mathbf{h} \times_2 \mathbf{r} \times_3 \mathbf{t}$, where $\mathbf{W}$ is a trainable matrix and $\times_i$ denotes tensor multiplication on dimension i.

Text-based methods are trained on the language dataset following Lin et al. [18] strategy. We consider models with bag-of-words (BoW) engineered features, convolutional neural networks, and pre-trained language models. Sentence-level baselines, including RF, TextCNN, BiGRU, and PubmedBERT, use majority voting on sentence-level predictions to produce final predictions for input examples that are longer than one sentence. We consider the following text-based baselines:

- **Random Forest (RF)** uses scaled BoW features with 100 decision trees and the Gini criterion as the classifier. Punctuations and English stop words are removed from sentences [73]. We extract 40,000 most frequently occurring N-grams and calculate the TF–IDF to produce sentence-level features to RF. The N-gram TF–IDF transformation is a widely-used and effective feature construction procedure.
- **TextCNN** [74] is a convolutional neural network for text, a useful deep learning algorithm for sentence classification tasks, such as sentiment analysis and question classification. The TextCNN we take as a baseline has 64 feature maps for each size and 8 different sizes of feature maps whose lengths of filter windows range from 2 to 10. We use ReLU as the activation function, 1 dimension max pooling after convolutional layers, and a dropout layer with a 0.25 dropout rate. We use skip-gram embeddings [75] to initialize the TextCNN model and train it using cross-entropy loss and Adam optimizer.
- **BiGRU** [76] is a recurrent neural network initialized in the same way as TextCNN. The encoder is a 1-layer BiGRU whose hidden states are aggregated into sentence embedding using a one-headed self-attention. These sentence embeddings are fed into a linear layer and provide logits for relation classification. The dimension of input word embeddings is 200, and the dimension of BiGRU hidden states is 100. We also add dropout layers with a 0.5 dropout rate after BiGRU and the linear layer. The sentence-level predictions given by this model will generate an instance-level classification result by majority voting.
- **BiGRU+Attention** [76] is the same as the BiGRU baseline with an added instance-level attention. Instead of voting, this approach uses one-head instance-level attention to aggregate latent vectors across all sentences in an instance. This provides an instance embedding for each sentence bag. Then we use a linear layer to provide logits of relation classification from instance embedding. This baseline is trained with cross-entropy loss on four relations, including Not Available (NA). We use the same hyper-parameters for this baseline as for BiGRU.
- **PubmedBERT** [77] is a domain-specific pre-trained language model trained on the corpus of abstracts from Pubmed with mask language modeling and next sentence prediction as pretext tasks. We take the checkpoint of PubmedBERT and fine-tune the model for disease relation extraction. We use the [CLS] token's embedding as the sentence embedding. Embeddings of sentences in a bag are aggregated with instance-level attention. The fine-tuning employs the Adam optimizer and a linear scheduler to adjust the learning rate. The loss function is also cross-entropy loss on four different relations, including Not Available (NA).

We consider two multimodal methods that learn on language-graph datasets:

- **BioLinkBERT** [78] is a pretrained language model that captures citation links in PubMed Central articles in its pretraining. It performs significantly better than PubMedBERT in knowledge-intensive tasks like relation extraction. BioLinkBERT utilizes both text and citations between biomedical documents in the form of a knowledge graph. However, the training and inference in BioLinkBERT is different from REMAP. For example, as a language model, BioLinkBERT can only run inference on text input instead of graph input. Nevertheless, we consider BioLinkBERT as a knowledge-enhanced PubMedBERT. And comparing REMAP-B with BioLinkBERT will provide a more comprehensive study. So we fine-tune BioLinkBERT on the text part of our datasets and inference on a single modality.

Table 2
Results of disease relation extraction on the human annotated dataset comparing with single-modal baselines. DDx: differential diagnosis, MC: may cause, MBC: may be caused by. The "micro" columns denote micro average accuracy or F1-score for DDx, MC, and MBC relation types. Further results are in Appendix B.

Modality		Model	Accuracy				F1-score			
			micro	DDx	MC	MBC	micro	DDx	MC	MBC
Text	Baselines	RF	72.0	68.6	70.8	76.4	37.8	53.1	25.5	19.2
		TextCNN	76.7	75.4	73.0	81.8	60.9	67.5	59.9	48.6
		BiGRU	77.4	73.0	77.2	82.0	62.0	67.9	54.0	59.8
		BiGRU+attention	78.6	75.0	78.2	82.6	64.6	67.7	63.5	60.6
		PubmedBERT	77.7	82.4	75.5	82.0	78.5	75.2	74.6	81.7
		SciBERT	86.2	82.6	78.0	90.8	80.0	80.1	75.7	85.1
	Ours	REMAP	88.2	83.6	89.0	92.0	80.9	80.7	80.0	82.6
		REMAP-M	88.6	84.2	89.0	**92.8**	81.5	81.6	79.6	**83.8**
		REMAP-B	**88.6**	**84.4**	**89.2**	92.4	**81.8**	**81.9**	**80.3**	83.3
Graph	Baselines	TransE_l2	75.1	70.7	72.7	81.8	63.2	68.0	57.0	62.2
		DistMult	69.8	77.5	61.3	70.5	56.1	71.0	43.4	51.5
		ComplEx	79.0	75.2	77.8	84.2	65.0	69.3	56.5	66.9
		RGCN	71.8	78.6	62.5	74.3	62.2	75.1	50.8	58.6
		TuckER	81.5	77.6	82.3	84.7	73.7	76.2	71.7	71.9
		HAN	83.7	80.0	84.2	88.1	82.5	83.9	81.0	81.4
	Ours	REMAP	89.6	86.4	89.6	**92.8**	83.5	84.3	81.6	**84.2**
		REMAP-M	89.3	87.0	88.4	92.6	83.3	85.7	78.8	83.8
		REMAP-B	**89.8**	**87.3**	**89.9**	92.2	**84.1**	**85.8**	**82.4**	82.7

- **HRERE** [50] is a strong baseline that improves relation extraction using knowledge graph embeddings. It specifies a joint language-graph model that combines language and graph information using a dissimilarity loss. HRERE is trained on both modalities and can run separate inferences on each modality. Although REMAP shares the same motivation of joint language-graph representation learning, REMAP further takes advantage of the alignment penalty to promote predictions across both modalities. We fine-tune HRERE on our disease relation extraction datasets as we do with REMAP.

Performance metrics. To evaluate the performance of our models, we use the accuracy of predicted relations between disease concepts, which is a widely used approach for benchmarking relation extraction methods [79]. For a given triplet (s_i, r_i, o_i) and a predicted score $p_i \in [0,1]$, we predict that relation r_i exists between s_i and o_i if the predicted score $p_i \geq \text{threshold}_i$, where each relation type has a specific threshold value. We determine the optimal threshold values for each relation type using the development set to achieve maximal F1-score. This process corresponds to a binary classification task for each relation type. We report the classification accuracy, precision, recall, and F1-score for all experiments conducted in this study.

4.4. Variants of REMAP approach

We conduct an ablation study to examine the utility of three components in REMAP and analyze its performance with and without each component.

- **REMAP-B without joint learning:** In the text-only ablations, we use SciBERT to obtain concept embeddings $\mathbf{h}_s^T$ and $\mathbf{h}_o^T$, and we combine them to produce a relation embedding $\mathbf{r}_k$, which is based on text information. Finally, we consider three scoring functions to classify disease–disease relations, and we denote the models as SciBERT (linear), SciBERT (TransE), and SciBERT (TuckER). Similarly, in the graph-only ablations, we first use a heterogeneous attention network to obtain graph embeddings $\mathbf{h}_s^G$ and $\mathbf{h}_o^G$. These embeddings are then combined into predictions by different scoring functions, including HAN (linear), HAN (TransE), and HAN (TuckER).
- **REMAP-B without EHR embeddings:** REMAP-B uses EHR embeddings [68] as initial node embedding $\mathbf{H}^{init}$. To examine the utility of EHR embeddings, we design an ablation study that initializes node embeddings using the popular Xavier initialization [80] instead

Table 3
Results of disease relation extraction on the human annotated dataset comparing with multimodal baselines. The table structure is the same as Table 2.

Modality		Model	Accuracy				F1-score			
			micro	DDx	MC	MBC	micro	DDx	MC	MBC
Text	Baselines	BioLinkBERT	86.9	83.1	88.8	90.3	78.3	79.2	75.4	80.7
		HRERE	86.6	83.4	88.2	89.8	77.9	79.1	74.3	81.0
	Ours	REMAP	88.2	83.6	89.0	92.0	80.9	80.7	80.0	82.6
		REMAP-M	88.6	84.2	89.0	**92.8**	81.5	81.6	79.6	**83.8**
		REMAP-B	**88.6**	**84.4**	**89.2**	92.4	**81.8**	**81.9**	**80.3**	83.3
Graph	Baselines	HRERE	87.2	83.6	88.8	90.7	79.5	80.3	76.1	82.8
	Ours	REMAP	89.6	86.4	89.6	**92.8**	83.5	84.3	81.6	**84.2**
		REMAP-M	89.3	87.0	88.4	92.6	83.3	85.7	78.8	83.8
		REMAP-B	**89.8**	**87.3**	**89.9**	92.2	**84.1**	**85.8**	**82.4**	82.7

of EHR embeddings. Other parts of the model are the same as in REMAP-B.

- **REMAP-B without EHR-guided negative sampling:** We use EHR-guided filtering to reduce false negative rates in negative sampling in REMAP-B. To evaluate the effectiveness of this negative sampling strategy, we conduct an ablation study where we replace EHR-guided negative sampling with random negative sampling. We randomly sample entity pairs not linked by meta-paths as negative samples. All other parts of the model remain the same as in REMAP-B.
- **REMAP-B without unaligned triplets:** Unaligned triplets denote triplets in the disease knowledge graph that do not have the corresponding sentences in the language dataset. To demonstrate how these unaligned triplets influence model performance, we design an ablation study in which we train a REMAP-B model on the reduced disease knowledge graph with unaligned triplets excluded.
- **REMAP-B without 2-hop meta-paths:** Meta-paths are a vital part of HAN. We conduct an ablation study to evaluate how meta-path length influences model performance. In this ablation study, we remove all 2-hop meta-paths and only keep one-hop meta-paths, i.e., edge types or relations.

5. Results

REMAP is a multimodal language-graph learning model. We evaluate REMAP's ability to extract disease relations from either text (Section 5.1) or graph-structured data (Section 5.2) on a human annotated dataset. REMAP is trained on a multimodal language-graph dataset (Section 3.4) and then tasked to extract disease relations from text or graph data alone, demonstrating the flexibility of REMAP in answering either graph-based or text-based disease queries. In this setting, we compare REMAP with backbone encoders that are not jointly trained with multimodal data to demonstrate how REMAP's joint learning structure benefits single-modal inference. Additionally, we compare REMAP with strong multimodal models (Section 5.3).

5.1. Extracting disease relations from text

The upper part of Table 2 shows the performance of single-modal methods and REMAP on the human annotated dataset of disease relations. We observed that pre-trained language models such as PubmedBERT [77] and SciBERT, used as the backbone of the text encoder in REMAP, consistently outperform other neural network baselines and random forest in terms of F1 score, achieving 78.5 and 80.0 in micro average, respectively. Furthermore, SciBERT also achieves the highest accuracy. Comparing REMAP with SciBERT, we demonstrate that joint learning on multimodal data improves relation classification performance by 0.9 percent in micro F1 score and 2 percent in average accuracy. We also find that REMAP-B and REMAP-M achieve the best performance across all settings, outperforming the baselines significantly.

5.2. Completing disease KG with novel disease–disease relationships

The lower part of Table 2 presents the performance results obtained on the human annotated dataset when given only the disease concept nodes as the query pairs in the knowledge graph. We observed that HAN significantly outperforms other knowledge graph embedding baselines in terms of both accuracy and F1 score. However, all variants of REMAP consistently outperform HAN in accuracy and F1 score, which demonstrates the effectiveness of REMAP. Moreover, REMAP-B is the best-performing variant of REMAP, achieving an accuracy of 89.8 and an F1-score of 84.1.

5.3. Comparing REMAP with multimodal methods

Table 3 shows the performance of multimodal models and REMAP. All REMAP models significantly outperform BioLinkBERT in the text modality and HRERE in both modalities. Furthermore, performance of models in Table 3 is substantially higher than in Table 2, indicating the overall benefits of multimodal joint learning. These findings suggest that multimodal learning can substantially improve disease relation extraction when only one data type is available at test time. Specifically, REMAP outperforms the strongest baseline HRERE by 2 and 2.6 absolute percentage points (in accuracy, in micro average) and 3.9 and 4.6 absolute percentage points (in F1-score, in micro average) in the text and graph modalities, respectively. In addition, we observe a more noticeable improvement between REMAP-B and HRERE in inference on the graph modality. In conclusion, the alignment penalty in REMAP-B is a superior method for using cross-modal information in training.

6. Discussion

We discuss REMAP's strategy for selecting negative disease–disease samples and examine the impact of negative sampling on model performance. We also investigate the trade-offs between translation and bilinear methods (Section 6.1). Additionally, we present a case study that illustrates REMAP's predictions (Section 6.2), and provide an ablation study to identify the key components of REMAP (Section 6.3).

6.1. Selection of negative disease pairs

We constructed negative samples for our dataset by selecting disease–disease pairs with co-occurrence numbers below a certain threshold in the EHR co-occurrence matrix, which are more representative than previous approaches [18] that generate negative samples. Our method does not require conceptual replacement to generate negative samples, making them more aligned with the real-world distribution of disease–disease co-occurrence. Additionally, we used threshold control to reduce the false negative rate.

We observed that TuckER outperformed TransE in our experiments. We attribute this to TransE's inherent limitation in handling 1-to-N relationships, where a single source disease may have multiple target diseases associated with it through a particular relation. TransE fails to capture the local graph neighborhoods of all target diseases simultaneously, as the embedding $\mathbf{t}$ is approximated as $\mathbf{h}+\mathbf{r}$. This lack of capacity to differentiate between embeddings $\mathbf{t}$ for different target diseases is the reason for its suboptimal performance. In contrast, the bilinear representation of TuckER can address this limitation of TransE, leading to better results. Therefore, we have used the TuckER component in our REMAP approach to facilitate joint learning.

6.2. Case study into hypobetalipoproteinemia and fatty liver disease

We illustrate REMAP with an example involving both graph and language modalities to classify the May Cause (MC) relationship between hypobetalipoproteinemia and fatty liver disease. Previous studies [81, 82] have reported that hypobetalipoproteinemia can cause fatty liver.

Table 4

Illustrative examples showing REMAP-B's correct predictions for diseases with text-only or graph-only information. We report the cases in each type in our human annotated dataset. Shown are three randomly selected examples for each case type together with auxiliary information, which includes terms of their head and tail entities, relations between them, prediction scores from REMAP-B, and the p-values of prediction scores.

Case type	Count	Head entity	Tail entity	Relation	Score	*p*-value
Present only in text	31	purpura	thrombocytopenia	MBC	0.92	≤0.001
		MEA-I	beta cell tumor	DDx	0.87	0.003
		pelvic infection	hypermenorrhea	DDx	0.85	0.002
Present only in graph	49	African trypanosomiases	joint pain	MC	0.97	≤0.001
		aortic coarctation	hypertension	MC	0.96	0.002
		diabetic acidoses	pancreatitis acute	DDx	0.89	≤0.001

Table 5

Results of ablation study. DDx: differential diagnosis, MC: may cause, MBC: may be caused by. The "micro" columns denote micro average accuracy or F1-score for DDx, MC, and MBC relation types.

Modality	Model	Accuracy				F1-score			
		micro	DDx	MC	MBC	micro	DDx	MC	MBC
Text	REMAP-B	**88.6**	**84.4**	**89.2**	92.4	**81.8**	**81.9**	80.3	83.3
	w/o joint learning (linear)	87.3	84.1	86.3	91.3	78.9	81.2	73.4	80.9
	w/o joint learning (TransE)	86.1	79.1	89.1	90.2	80.5	79.3	**82.2**	80.8
	w/o joint learning (TuckER)	87.9	83.6	87.0	**93.2**	80.0	80.1	75.7	85.1
	w/o EHR embedding	87.6	83.2	88.2	91.4	79.6	79.8	78.2	81.1
	w/o EHR-guided negative sampling	87.7	83.1	88.4	92.2	79.1	80.2	77.9	81.1
	w/o unaligned triplets	88.2	83.0	88.5	**93.2**	81.0	80.0	78.8	**85.2**
	w/o 2-hop meta-paths	86.6	79.8	87.4	92.0	78.2	79.6	77.8	79.8
Graph	REMAP-B	**89.8**	**87.3**	**89.9**	**92.2**	**84.1**	**85.8**	**82.4**	**82.7**
	w/o joint learning (linear)	87.4	82.4	89.3	90.2	80.5	80.5	82.1	78.4
	w/o joint learning (TransE)	85.8	81.6	85.3	90.4	77.2	78.9	73.1	78.9
	w/o joint learning (TuckER)	88.9	85.8	89.4	91.6	82.5	83.9	81.0	81.4
	w/o EHR embedding	87.6	84.4	87.2	91.4	80.3	82.0	76.8	81.4
	w/o EHR-guided negative sampling	87.2	86.3	89.4	91.1	81.3	84.3	81.8	81.7
	w/o unaligned triplets	87.3	84.8	87.8	89.4	79.5	82.4	77.7	76.2
	w/o 2-hop meta-paths	85.1	80.9	86.3	89.0	77.3	80.2	75.9	78.4

Fig. 3 illustrates the prediction of the MC relationship between these two diseases made by the REMAP-B joint learning model. The graph provides structural information to REMAP-B. For instance, hypobetalipoproteinemia has the most outgoing edges of MC type among all edge types, and fatty liver disease has 4 outgoing edges of the May Be Caused (MBC) type. Moreover, the graph encoder can capture information from any possible meta-paths linking these two nodes, such as "Hypobetalipoproteinemia" $\xrightarrow{MC}$ "X" $\xrightarrow{MC}$ "Fatty liver", where X is another node in the knowledge graph. The language encoder can extract information from free text, semantic types, and tokens we added to the vocabulary of the pretrained language model. In the case of joint learning, the language model can update its internal representations by extracting part of the disease representation from the knowledge graph, leading to better disease relation extraction.

To investigate the discrepancies between language and graph predictions, we conducted another case study to identify samples that could only be correctly predicted by one of the two modalities. The results of this case study are presented in Table 4. Specifically, we found that 31 samples were only correctly predicted by the text, while 49 samples were only correctly predicted by the graph. To illustrate these findings, we randomly selected three cases from each category and displayed their terms and auxiliary information. In order to calculate the p-values of the prediction scores in the table, we randomly swapped the tail entities in the triplets with other entities that had no known relation to the head entities. The resulting p-values represent the percentiles of the prediction scores in the distribution of negative prediction scores.

6.3. Ablation study

We conducted an ablation study to evaluate the effectiveness of four key components in REMAP-B: joint learning loss, scoring functions, EHR embedding, and unaligned triplets in the knowledge graph. Table 5 presents the results of this study, comparing the performance of REMAP-B with the performance after excluding each component. Our results show that the joint learning techniques in REMAP-B are crucial for its performance, as we observe drops in performance across almost all metrics and edge types in both text and graph modalities, except for the MBC edge type in text modality. In this case, the accuracy increases by 0.8% from 92.4% to 93.2% when excluding joint learning with TuckER. Moreover, we find that the EHR embedding plays a significant role in REMAP-B, as the performance consistently decreases in all metrics, edge types, and modalities when we exclude it. The F1-score in graph modality experiences the largest drop, decreasing by 3.8% from 84.1% to 80.3%. This analysis demonstrates that REMAP-B can effectively employ EHR data for disease relation extraction.

We conducted an experiment to compare the performance of EHR-guided negative sampling and negative sampling based on meta-paths on the knowledge graph. The results, shown in Table 5, indicate a consistent drop in performance across all settings when we replace EHR-guided negative sampling with path-based negative sampling, particularly in graph inference. Specifically, we observed a decrease of 2.6% in micro accuracy and 2.8% in F1 score, suggesting that EHR-guided negative sampling drives the strong performance of REMAP-B.

Furthermore, accuracy and F1-score decrease when unaligned triplets are removed from the knowledge graph in all settings, except for the edge types MBC (May Be Caused). Interestingly, the accuracy, in this case, increases by 0.8% and the F1 score by 1.9%, indicating the importance of leveraging unsupervised information from unaligned triplets. Finally, the ablation study also reveals that REMAP-B's performance can significantly drop when 2-hop meta-paths are not considered, underscoring the importance of meta-paths in the graph encoder.

All algorithmic elements of REMAP-B are essential for strong performance in disease relation extraction. Among these components, joint learning is the most important as its exclusion leads to a significant drop in performance across all experiments.

7. Conclusion

We present REMAP, a multimodal approach for disease relation extraction that combines language modeling with knowledge graphs. Our experiments on a dataset of clinical expert annotations demonstrate that REMAP outperforms methods that learn from text or knowledge graphs alone. Furthermore, REMAP can extract and classify disease relationships in challenging scenarios where either the language or graph modality is missing. Finally, we provide a new dataset of disease relationships that can serve as a benchmark for evaluating and comparing disease relation extraction algorithms.

To advance multimodal learning in the biomedical field, it would be beneficial to create benchmarking biomedical language-graph datasets for relation extraction. REMAP can be applied to different knowledge graphs to capture various relations and entities in the biomedical domain [83,84]. A fruitful future direction would be to explore and incorporate diverse types of relationships between diseases into comprehensive disease knowledge graphs.

CRediT authorship contribution statement

Yucong Lin: Developed the method, Carried out all analyses in this study, Writing – original draft. **Keming Lu:** Developed the method, Carried out all analyses in this study, Writing – original draft. **Sheng Yu:** Experimental data, Writing – original draft. **Tianxi Cai:** Conceived and designed the study, Writing – original draft. **Marinka Zitnik:** Conceived and designed the study, Guidance, Designing methodology, Outlining experiments, Writing – original draft.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data and code availability

Python implementation of REMAP is available on Github at https://github.com/Lukeming-tsinghua/REMOD. The human annotated data set used for evaluation of REMAP is available at https://doi.org/10.6084/m9.figshare.17776865.

Acknowledgments

We gratefully acknowledge the support of the Harvard Translational Data Science Center for a Learning Healthcare System (CELeHS) (T.C.). M.Z. is supported, in part, by the National Science Foundation under Nos. IIS-2030459 and IIS-2033384, US Air Force Contract No. FA8702-15-D-0001, National Institutes of Health under No. R01HD108794, Harvard Data Science Initiative, Amazon Research Award, Google Research Scholar Program, Bayer Early Excellence in Science Award, AstraZeneca Research, and Roche Alliance with Distinguished Scientists Award. Any opinions, findings, conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funders.

Appendix A. Data preprocessing

Data preprocessing and feature engineering follow the strategy outlined in Lin et al. [18]. We outline the data preparation process in this section. This process transforms the raw datasets, including relation triplets, text corpus, and electronic health records embeddings, into AI-ready data for AI analyses. The code is publicly available on GitHub[5].

A.1. Relation triplet collection

Acquiring relation triplets is the first step in the preparation of training data. Most disease relation triplets are directly collected from the Diseases Database [66]. We also collect relation triplets by resolving semi-structured content on Medscape. Pages on it usually follow a very standard template, which allows one to easily locate sections about the target relations, where the entities are usually presented in lists and tables. Most commonly, the page title provides the head entity, and the section title specifies the relation, lists, and tables in the section, providing the tail entities. Therefore, we write simple web scraping scripts and apply maximum mapping to identify mentions of entities with UMLS CUIs and assemble them into relation triplets.

A.2. Document preparation

The input to this step is a set of documents collected from online sources. These documents are used to pretrain a model *en_core_web_sm* in *SpaCy* to split the document into sentences with additional structure information, including the title and subject headings.

A.3. Entity linking

The input to entity linking are sentences extracted from the corpus. We use the nested forward maximum matching algorithm to annotate the mentions with medical terms collected from UMLS.

A.4. Data cleaning

We discard sentences that are shorter than 5 words. And we generate the static position embedding for baseline models, such as TextCNN and RNNs. We also make sure the head entity appears before the tail entity in sentences, and those sentences with entities in reverse order are moved to the opposite relation (i.e., from may_cause to may_be_caused_by).

Appendix B. Further information on remap's performance

Table A1 extends Table 3 from the main text with additional information about REMAP's performance measured using precision, recall, and F1-score metrics. Shown is the performance that REMAP achieves when the REMAP model, although trained on the multimodal graph-text dataset, is asked to make predictions at test time using only text information.

Table A2 extends Table 3 from the main text with additional information about REMAP's performance measured using precision, recall, and F1-score metrics. Shown is the performance that REMAP achieves when the REMAP model, although trained on the multimodal graph-text dataset, is asked to make predictions at test time using only information from the disease knowledge graph.

Table A3 shows the performance of REMAP-B and the baseline HRERE on three distantly supervised relation extraction datasets in the general domain, including Wiki80m [85], Wiki20m [86], and NYT10m [86]. Specifically, NYT10 m contains 25 relations extracted from the Freebase knowledge graph and a distantly supervised corpus with about 470 thousand instances. Both Wiki80 m and Wiki20 m are distantly supervised relation extraction datasets created from Wikidata. They contain 80 and 81 relations, respectively. Wiki20 m includes about 700 thousand instances, while Wiki80 m is much smaller and contains only 55 thousand instances. In conclusion, all three datasets are distant supervised RE datasets and contain abundant relation triplets as a knowledge graph, which greatly fit the setting we introduce in this work.

However, these datasets do not focus on disease relation extraction. So we drop elements in REMAP-B designed for disease relation extraction in our work, including the EHR-guided negative sampling

[5] https://github.com/lychyzclc/High-throughput-relation-extraction-algorithm

Table A1
Performance of multimodal methods for identifying candidate disease–disease relations from text data. Shown is the average performance across multiple independent runs calculated on the human annotated set. Higher values indicate better performance.

Model	Precision				Recall				F1-score			
	Minor	DDx	MC	MBC	Minor	DDx	MC	MBC	Minor	DDx	MC	MBC
Random Forest	69.2	71.8	67.6	58.3	26.0	42.2	15.7	58.3	37.8	53.1	25.5	19.2
TextCNN	67.8	76.2	56.7	78.2	55.3	60.7	63.5	35.2	60.9	67.5	59.9	48.6
BiGRU	69.1	68.1	75.3	65.7	56.3	67.8	42.1	54.9	62.0	67.9	54.0	59.8
BiGRU+attention	70.6	74.4	67.9	67.7	59.6	62.1	59.7	54.9	64.6	67.7	63.5	60.6
PubmedBERT	71.2	64.0	71.2	78.4	87.4	91.2	85.5	85.3	78.5	75.2	74.6	81.7
SciBERT (linear)	86.6	81.0	96.9	88.3	72.5	81.4	59.1	74.6	78.9	81.2	73.4	80.9
SciBERT (TransE)	74.9	68.2	86.2	77.4	87.0	94.8	78.6	84.4	80.5	79.3	82.2	80.8
SciBERT (TuckER)	87.3	**81.7**	93.5	91.5	73.9	78.6	63.5	**79.5**	80.0	80.1	75.7	**85.1**
REMAP	85.8	79.9	94.8	88.0	76.6	81.4	**69.2**	77.9	80.9	80.7	80.0	82.6
REMAP-M	**87.4**	79.9	**97.3**	**93.0**	76.4	83.3	67.3	76.2	81.5	81.6	79.6	83.8
REMAP-B	86.2	79.7	95.7	89.6	**77.8**	**84.3**	**69.2**	77.9	**81.8**	**81.9**	**80.3**	83.3

Table A2
Performance of multimodal methods for identifying candidate disease–disease relations from graph-structured data. Shown is the average performance across multiple independent runs calculated on the human annotated set. Higher values indicate better performance.

Model	Precision				Recall				F1-score			
	Minor	DDx	MC	MBC	Minor	DDx	MC	MBC	Minor	DDx	MC	MBC
TransE_l2	61.3	63.0	57.3	63.0	65.2	73.8	56.6	61.5	63.2	68.0	57.0	62.2
DistMult	53.6	77.8	40.7	43.1	58.9	65.2	46.5	63.9	56.1	71.0	43.4	51.5
ComplEx	71.7	72.2	75.0	68.4	59.5	66.7	45.3	65.6	65.0	69.3	56.5	66.9
RGCN	56.2	74.9	44.2	48.9	69.7	75.2	59.7	73.0	62.2	75.1	50.8	58.6
TuckER	70.1	69.8	74.3	66.2	77.8	83.8	69.2	78.7	73.7	76.2	71.7	71.9
HAN (linear)	82.0	75.2	92.9	84.8	79.0	86.7	**73.6**	73.0	80.5	80.5	**82.1**	78.4
HAN (TransE)	81.3	76.1	88.4	84.9	73.5	81.9	62.3	73.8	77.2	78.9	73.1	78.9
HAN (TuckER)	85.7	80.1	94.2	88.5	79.4	88.1	71.1	75.4	82.5	83.9	81.0	81.4
REMAP	**87.0**	**81.7**	93.5	**90.6**	80.2	87.1	72.3	**78.7**	83.5	84.3	81.6	**84.2**
REMAP-M	85.6	79.8	**93.9**	89.7	81.1	**92.4**	67.9	**78.7**	83.3	85.7	78.8	83.8
REMAP-B	86.6	81.3	93.6	90.3	**81.7**	91.0	**73.6**	76.2	**84.1**	**85.8**	**82.4**	82.7

Table A3
Additional experiments on relation extraction datasets in the general domain for demonstrating the generalization ability of REMAP. We report **micro F1 score (F1)** and **average precision (AP)** on each dataset.

Modality	Model	Wiki80m		Wiki20m		NYT10m	
		F1	AP	F1	AP	F1	AP
Text	HRERE	80.04	87.42	77.33	88.38	30.50	54.40
	REMAP-B	88.73	92.16	84.99	89.92	52.19	66.39
Graph	HRERE	83.16	85.77	79.45	87.30	34.09	53.99
	REMAP-B	88.24	91.99	82.98	89.74	52.09	65.28

and initialization of node embeddings based on EHR embeddings of disease concepts. We follow the same experimental setup as described in Section 4. The code of these experiments is also available on Github[6].

Although we have eliminated all biomedically-related features in REMAP-B, we find that REMAP-B outperforms HRERE on both text and graph modalities across all three datasets. These results provide additional evidence of the broad generalization of REMAP-B and suggest that REMAP-B is broadly applicable, including in the general domain.

Appendix C. Selection of hyperparameters

Grid search on the validation set was used to select REMAP's hyper-parameters (Table A4).

Table A4
Selection of hyper-parameters in REMAP. We use grid search to select hyper-parameters on the validation set.

REMAP's component	Model parameter	Notation	Value
Neural architecture	Padding length of sentences	d_l	256
	Hidden size of SciBERT output	d_{hs}	768
	Hidden size of HAN output	d_{ha}	100
	Hidden size of initial node embedding	d_{hi}	1,000
	Hidden size of node embedding	d_h	100
	Hidden size of relation embedding	d_r	100
Multimodal training	Max sentence sample number	$l_{m(\max)}$	12
	Training batch size	b_{train}	4
	Test batch size	b_{test}	16
	Weight decay	wd	5×10^{-5}
	Learning rate	lr	1×10^{-5}
	Gradient accumulate step	step_g	4
	Optimizer		Adam
	Scheduler		Linear
	Warmup rate	r_{warmup}	0.1
Language model	Max sentence sample number	$l_{m(\max)}$	12
	Training batch size	b_{train}	4
	Test batch size	b_{test}	16
	Weight decay	wd	5×10^{-5}
	Learning rate	lr	1×10^{-5}
	Gradient accumulate step	step_g	4
	Optimizer		Adam
	Scheduler		Linear
	Warmup rate	r_{warmup}	0.1
Graph model	Training batch size	b_{train}	512
	Test batch size	b_{test}	512
	Weight decay	wd	1×10^{-8}
	Learning rate	lr	1×10^{-3}
	Optimizer		Adam
	Scheduler		StepLR
	StepLR scheduler	γ	0.9

[6] https://github.com/Lukeming-tsinghua/REMAP-General

Appendix D. Further description of the graphical abstract

The graphical abstract demonstrates the workflow of our work. First, we use an existing distantly supervised disease relation dataset obtained from various disease knowledge graphs and language corpora. We further enhance it with features from EHR data, such as the co-occurrence of diseases in EHR and disease concept embeddings via decomposition. Considering the noise in distant supervision, we also created a human annotated dataset to evaluate our method. In the negative sampling, we creatively use co-occurrence in EHR to reduce the false negative rates. Then, two parts of data from different modalities — triplets in the knowledge graph and sentences from corpora — will be encoded with two separate encoders. These two encoders are co-trained using the alignment penalty optimized for disease relation classification. In the end, encoders optimized for each modality can infer independently on corresponding modalities.

Appendix E. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.jbi.2023.104415.

References

[1] C. Ruiz, M. Zitnik, J. Leskovec, Identification of disease treatment mechanisms through the multiscale interactome, Nature Commun. 12 (1) (2021) 1–15.

[2] C. Hong, E. Rush, M. Liu, D. Zhou, J. Sun, A. Sonabend, V.M. Castro, P. Schubert, V.A. Panickan, T. Cai, et al., Clinical knowledge extraction via sparse embedding regression (KESER) with multi-center large scale electronic health record data, npj Digit. Med. 4 (1) (2021) 1–11.

[3] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, J. Taylor, Freebase: a collaboratively created graph database for structuring human knowledge, in: ACM SIGMOD, 2008, pp. 1247–1250.

[4] T.C. Rindflesch, H. Kilicoglu, M. Fiszman, G. Rosemblat, D. Shin, Semantic MEDLINE: An advanced information management application for biomedicine, Inf. Serv. Use (2011) 15–21.

[5] A. Carlson, J. Betteridge, B. Kisiel, B. Settles, E.R. Hruschka, T.M. Mitchell, Toward an architecture for never-ending language learning, in: AAAI, 2010.

[6] X. Dong, E. Gabrilovich, G. Heitz, W. Horn, N. Lao, K. Murphy, T. Strohmann, S. Sun, W. Zhang, Knowledge vault: A web-scale approach to probabilistic knowledge fusion, in: ACM SIGKDD, 2014, pp. 601–610.

[7] L.B. Soares, N. FitzGerald, J. Ling, T. Kwiatkowski, Matching the blanks: Distributional similarity for relation learning, in: ACL, 2019, pp. 2895–2905.

[8] Z. Zhong, D. Chen, A frustratingly easy approach for joint entity and relation extraction, in: NAACL-HT, 2021, pp. 50–61.

[9] M. Mintz, S. Bills, R. Snow, D. Jurafsky, Distant supervision for relation extraction without labeled data, in: ACL-IJCNLP, 2009, pp. 1003–1011.

[10] K. Lei, D. Chen, Y. Li, N. Du, M. Yang, W. Fan, Y. Shen, Cooperative denoising for distantly supervised relation extraction, in: COLING, 2018, pp. 426–436.

[11] C. Xiao, Y. Yao, R. Xie, X. Han, Z. Liu, M. Sun, F. Lin, L. Lin, Denoising relation extraction from document-level distant supervision, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP, Association for Computational Linguistics, 2020, pp. 3683–3688, http://dx.doi.org/10.18653/v1/2020.emnlp-main.300, Online, URL https://aclanthology.org/2020.emnlp-main.300.

[12] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, O. Yakhnenko, Translating embeddings for modeling multi-relational data, Adv. Neural Inf. Process. Syst. 26 (2013) 2787–2795.

[13] I. Balažević, C. Allen, T.M. Hospedales, TuckER: Tensor factorization for knowledge graph completion, in: EMNLP, 2019.

[14] M. Zitnik, F. Nguyen, B. Wang, J. Leskovec, A. Goldenberg, M.M. Hoffman, Machine learning for integrating data in biology and medicine: Principles, practice, and opportunities, Inf. Fusion 50 (2019) 71–91.

[15] Z. He, W. Chen, Y. Wang, W. Zhang, G. Wang, M. Zhang, Improving neural relation extraction with positive and unlabeled learning, in: AAAI, Vol. 34, 2020, pp. 7927–7934.

[16] P.H. Le-Khac, G. Healy, A.F. Smeaton, Contrastive representation learning: A framework and review, IEEE Access (2020).

[17] P. Su, Y. Peng, K. Vijay-Shanker, Improving BERT model using contrastive learning for biomedical relation extraction, in: BIONLP, 2021.

[18] Y. Lin, K. Lu, Y. Chen, C. Hong, S. Yu, High-throughput relation extraction algorithm development associating knowledge articles and electronic health records, 2020, arXiv:2009.03506.

[19] Y. Ektefaie, G. Dasoulas, A. Noori, M. Farhat, M. Zitnik, Multimodal learning with graphs, 2022, arXiv:2209.03299.

[20] A.M. Cray, A. The unified medical language system, Methods Inf. Med. (1993).

[21] Y. Lin, Y. Li, K. Lu, C. Ma, P. Zhao, D. Gao, Z. Fan, Z. Cheng, Z. Wang, S. Yu, Long-distance disorder-disorder relation extraction with bootstrapped noisy data, J. Biomed. Inform. 109 (2020) 103529.

[22] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, BERT: Pre-training of deep bidirectional transformers for language understanding, in: NAACL-HLT, 2019, pp. 4171–4186.

[23] I. Beltagy, K. Lo, A. Cohan, SciBERT: A pretrained language model for scientific text, in: EMNLP-IJCNLP, 2019, pp. 3615–3620.

[24] J. Chen, B. Hu, W. Peng, Q. Chen, B. Tang, Biomedical relation extraction via knowledge-enhanced reading comprehension, BMC Bioinformatics (2022) 1–19.

[25] G. Li, C. Wu, K. Vijay-Shanker, Noise reduction methods for distantly supervised biomedical relation extraction, in: BioNLP, 2017, pp. 184–193.

[26] Q. Wang, Z. Mao, B. Wang, L. Guo, Knowledge graph embedding: A survey of approaches and applications, IEEE Trans. Knowl. Data Eng. 29 (12) (2017) 2724–2743.

[27] M.M. Li, K. Huang, M. Zitnik, Graph representation learning in biomedicine and healthcare, Nat. Biomed. Eng. (2022) 1–17.

[28] M. Zitnik, B. Zupan, Collective pairwise classification for multi-way analysis of disease and drug data, in: The Pacific Symposium on Biocomputing, 2016, pp. 81–92.

[29] B. Shi, T. Weninger, ProjE: Embedding projection for knowledge graph completion, in: AAAI, Vol. 31, 2017.

[30] Y. Lin, Z. Liu, M. Sun, Y. Liu, X. Zhu, Learning entity and relation embeddings for knowledge graph completion, in: AAAI, 2015.

[31] X. Wang, X. He, Y. Cao, M. Liu, T.-S. Chua, KGAT: Knowledge graph attention network for recommendation, in: KDD, 2019, pp. 950–958.

[32] Z. Sun, J. Yang, J. Zhang, A. Bozzon, L.-K. Huang, C. Xu, Recurrent knowledge graph embedding for effective recommendation, in: ACM Conference on Recommender Systems, 2018, pp. 297–305.

[33] Z. Wang, J. Zhang, J. Feng, Z. Chen, Knowledge graph and text jointly embedding, in: EMNLP, 2014, pp. 1591–1601.

[34] G. Ji, S. He, L. Xu, K. Liu, J. Zhao, Knowledge graph embedding via dynamic mapping matrix, in: ACL, 2015, pp. 687–696.

[35] M. Nickel, V. Tresp, H.-P. Kriegel, A three-way model for collective learning on multi-relational data, in: ICML, 2011.

[36] B. Yang, W. tau Yih, X. He, J. Gao, L. Deng, Embedding entities and relations for learning and inference in knowledge bases, 2015, Computing Research Repository abs/1412.6575.

[37] T. Trouillon, C.R. Dance, J. Welbl, S. Riedel, É. Gaussier, G. Bouchard, Knowledge graph completion via complex tensor factorization, J. Mach. Learn. Res. 18 (2017).

[38] I. Balazevic, C. Allen, T. Hospedales, TuckER: Tensor factorization for knowledge graph completion, in: EMNLP-IJCNLP, 2019, pp. 5185–5194.

[39] M. Schlichtkrull, T.N. Kipf, P. Bloem, R. Van Den Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: European Semantic Web Conference, Springer, 2018, pp. 593–607.

[40] D. Busbridge, D. Sherburn, P. Cavallo, N.Y. Hammerla, Relational graph attention networks, 2019, arXiv:1904.05811.

[41] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, P.S. Yu, Heterogeneous graph attention network, in: WWW, 2019, pp. 2022–2032.

[42] M. Zitnik, M. Agrawal, J. Leskovec, Modeling polypharmacy side effects with graph convolutional networks, Bioinformatics 34 (13) (2018) i457–i466.

[43] W. Liu, P. Zhou, Z. Zhao, Z. Wang, Q. Ju, H. Deng, P. Wang, K-BERT: Enabling language representation with knowledge graph, in: AAAI, Vol. 34, 2020, pp. 2901–2908.

[44] Y. Sun, S. Wang, Y. Li, S. Feng, X. Chen, H. Zhang, X. Tian, D. Zhu, H. Tian, H. Wu, ERNIE: Enhanced representation through knowledge integration, 2019, arXiv:1904.09223.

[45] Z. Zhang, X. Han, Z. Liu, X. Jiang, M. Sun, Q. Liu, ERNIE: Enhanced language representation with informative entities, in: ACL, 2019, pp. 1441–1451.

[46] B. He, D. Zhou, J. Xiao, X. Jiang, Q. Liu, N.J. Yuan, T. Xu, BERT-MK: Integrating graph contextualized knowledge into pre-trained language models, in: Findings of EMNLP, 2020, pp. 2281–2290.

[47] R. Koncel-Kedziorski, D. Bekal, Y. Luan, M. Lapata, H. Hajishirzi, Text generation from knowledge graphs with graph transformers, 2019, arXiv:1904.02342.

[48] T. Sun, Y. Shao, X. Qiu, Q. Guo, Y. Hu, X. Huang, Z. Zhang, CoLAKE: Contextualized language and knowledge embedding, in: International Conference on Computational Linguistics, 2020, pp. 3660–3670.

[49] L. Hu, L. Zhang, C. Shi, L. Nie, W. Guan, C. Yang, Improving distantly-supervised relation extraction with joint label embedding, in: EMNLP, 2019, pp. 3812–3820.

[50] P. Xu, D. Barbosa, Connecting language and knowledge with heterogeneous representations for neural relation extraction, in: NAACL-HLT, 2019, pp. 3201–3206.

[51] N. Zhang, S. Deng, Z. Sun, G. Wang, X. Chen, W. Zhang, H. Chen, Long-tail relation extraction via knowledge graph embeddings and graph convolution networks, in: NAACL-HLT, 2019, pp. 3016–3025.

[52] Y. Wang, H. Zhang, G. Shi, Z. Liu, Q. Zhou, A model of text-enhanced knowledge graph representation learning with mutual attention, IEEE Access 8 (2020) 52895–52905.

[53] X. Han, Z. Liu, M. Sun, Joint representation learning of text and knowledge for knowledge graph completion, 2016, arXiv:1611.04125.

[54] Z. Ji, Z. Lei, T. Shen, J. Zhang, Joint representations of knowledge graphs and textual information via reference sentences, IEICE Trans. Inf. Syst. (2020) 1362–1370.

[55] Q. Dai, N. Inoue, P. Reisert, R. Takahashi, K. Inui, Distantly supervised biomedical knowledge acquisition via knowledge graph based attention, in: The Workshop on Extracting Structured Knowledge from Scientific Publications, 2019, pp. 1–10.

[56] G. Stoica, E.A. Platanios, B. Póczos, Improving relation extraction by leveraging knowledge graph link prediction, 2020, arXiv:2012.04812.

[57] Q. Wang, L. Zhan, P. Thompson, J. Zhou, Multimodal learning with incomplete modalities by knowledge distillation, in: SIGKDD, 2020, pp. 1828–1838.

[58] Q. Suo, W. Zhong, F. Ma, Y. Yuan, J. Gao, A. Zhang, Metric learning on healthcare data with incomplete modalities, in: IJCAI, 2019, pp. 3534–3540.

[59] T. Zhou, M. Liu, K.-H. Thung, D. Shen, Latent representation learning for Alzheimer's disease diagnosis with incomplete multi-modality neuroimaging and genetic data, IEEE Trans. Med. Imaging (2019) 2411–2422.

[60] Y. Yang, D.-C. Zhan, X.-R. Sheng, Y. Jiang, Semi-supervised multi-modal learning with incomplete modalities, in: IJCAI, 2018, pp. 2998–3004.

[61] L. Cai, Z. Wang, H. Gao, D. Shen, S. Ji, Deep adversarial learning for multi-modality missing data completion, in: ACM SIGKDD, 2018, pp. 1158–1166.

[62] N. Jaques, S. Taylor, A. Sano, R. Picard, Multimodal autoencoder: A deep learning approach to filling in missing sensor data and enabling better mood prediction, in: ACII, 2017, pp. 202–208.

[63] Y. Sun, J. Han, X. Yan, P.S. Yu, T. Wu, PathSim: meta path-based top-k similarity search in heterogeneous information networks, in: VLDB, 2011, pp. 992–1003.

[64] X. Lan, X. Zhu, S. Gong, Knowledge distillation by on-the-fly native ensemble, in: NeurIPS, 2018.

[65] Q. Guo, X. Wang, Y. Wu, Z. Yu, D. Liang, X. Hu, P. Luo, Online knowledge distillation via collaborative learning, in: CVPR, 2020, pp. 11020–11029.

[66] Laptop diseases database ver 2.0; medical lists and links diseases database [internet] 2015, [cited 2020 Aug 16]. Available from: http://www.diseasesdatabase.com/.

[67] P. Frishauf, Medscape–The first 5 years, Medscape Gen. Med. (2005) 5.

[68] A.L. Beam, B. Kompa, A. Schmaltz, I. Fried, G. Weber, N. Palmer, X. Shi, T. Cai, I.S. Kohane, Clinical concept embeddings learned from massive sources of multimodal medical data, in: Pacific Symposium on Biocomputing 2020, 2019, pp. 295–306.

[69] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, et al., Huggingface's transformers: State-of-the-art natural language processing, 2019, arXiv:1910.03771.

[70] B. Yang, W. tau Yih, X. He, J. Gao, L. Deng, Embedding entities and relations for learning and inference in knowledge bases, in: ICLR, 2015.

[71] T. Trouillon, J. Welbl, S. Riedel, E. Gaussier, G. Bouchard, Complex embeddings for simple link prediction, in: ICML, 2016.

[72] M. Schlichtkrull, T.N. Kipf, P. Bloem, R.v.d. Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, 2017, arXiv:1703.06103.

[73] C.D. Manning, M. Surdeanu, J. Bauer, J.R. Finkel, S. Bethard, D. McClosky, The stanford corenlp natural language processing toolkit, in: Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations, 2014, pp. 55–60.

[74] Y. Kim, Convolutional neural networks for sentence classification, in: EMNLP, 2014, pp. 1746–1751.

[75] T. Mikolov, I. Sutskever, K. Chen, G.S. Corrado, J. Dean, Distributed representations of words and phrases and their compositionality, Adv. Neural Inf. Process. Syst. 26 (2013).

[76] K. Cho, B. Van Merriënboer, D. Bahdanau, Y. Bengio, On the properties of neural machine translation: Encoder-decoder approaches, 2014, arXiv preprint arXiv:1409.1259.

[77] Y. Gu, R. Tinn, H. Cheng, M. Lucas, N. Usuyama, X. Liu, T. Naumann, J. Gao, H. Poon, Domain-specific language model pretraining for biomedical natural language processing, ACM Trans. Comput. Healthc. (HEALTH) (2021) 1–23.

[78] M. Yasunaga, J. Leskovec, P. Liang, LinkBERT: Pretraining language models with document links, in: Association for Computational Linguistics, ACL, 2022.

[79] W. Hu, M. Fey, M. Zitnik, Y. Dong, H. Ren, B. Liu, M. Catasta, J. Leskovec, Open graph benchmark: Datasets for machine learning on graphs, in: NeurIPS, 2020.

[80] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: AISTATS, 2010, pp. 249–256.

[81] G. Schonfeld, B.W. Patterson, D.A. Yablonskiy, T.S. Tanoli, M. Averna, N. Elias, P. Yue, J. Ackerman, Fatty liver in familial hypobetalipoproteinemia: triglyceride assembly into VLDL particles is affected by the extent of hepatic steatosis, J. Lipid Res. 44 (3) (2003) 470–478.

[82] J. Rodrigues, A. Azevedo, S. Tavares, C. Rocha, E.S. Silva, Non-alcoholic fatty liver disease associated with hypobetalipoproteinemia: report of three cases and a novel mutation in apob gene, Nascer e Crescer-Birth Growth Med. J. 25 (2) (2016) 104–107.

[83] S. Yu, Z. Yuan, J. Xia, S. Luo, H. Ying, S. Zeng, J. Ren, H. Yuan, Z. Zhao, Y. Lin, et al., Bios: An algorithmically generated biomedical knowledge graph, 2022, arXiv preprint arXiv:2203.09975.

[84] P. Chandak, K. Huang, M. Zitnik, Building a knowledge graph to enable precision medicine, Sci. Data 10 (1) (2023) 67.

[85] X. Han, T. Gao, Y. Yao, D. Ye, Z. Liu, M. Sun, OpenNRE: An open and extensible toolkit for neural relation extraction, 2019, arXiv preprint arXiv:1909.13078.

[86] T. Gao, X. Han, K. Qiu, Y. Bai, Z. Xie, Y. Lin, Z. Liu, P. Li, M. Sun, J. Zhou, Manual evaluation matters: reviewing test protocols of distantly supervised relation extraction, 2021, arXiv preprint arXiv:2105.09543.