Diagnosis Ranking with Knowledge Graph Convolutional Networks

Bing Liu(✉) [0000−0002−7858−7468], Guido Zuccon[0000−0003−0271−5563], Wen Hua[0000−0001−5456−7035], and Weitong Chen[0000−0003−1001−7925]

The University of Queensland, St Lucia, Australia
{bing.liu, g.zuccon, w.hua, w.chen9}@uq.edu.au

Abstract. The automatic diagnosis of a medical condition provided the symptoms exhibited by a patient is at the basis of systems for clinical decision support, as well as for applications such as symptom checkers. Existing methods have not fully exploited medical knowledge: this likely hinders their effectiveness. In this work, we propose a knowledge-aware diagnosis ranking framework based on medical knowledge graph (KG) and graph convolutional neural network (GCN). The medical KG is used to model hierarchy and causality relationships between diseases and symptoms. We have evaluated our proposed method using realistic patient cases. The empirical results show that our knowledge-aware diagnosis ranking framework can improve the effectiveness of medical diagnosis.

Keywords: Knowledge Graph · Graph Convolutional Networks · Diagnosis Ranking

1 Introduction

A common task in medical practice is to identify a diagnosis for a patient presenting with one or more symptoms. To do so, clinicians rely on their extensive medical knowledge about the relationships between symptoms and the possible diagnoses, and weight up symptoms (and laboratory findings) to determine the most likely diagnosis, often through a process called differential diagnosis [25]. Computer assisted or automated methods for medical diagnosis have emerged where computer algorithms are used to mine a large amount of medical data (from medical literature or electronic health records) to provide clinicians with recommendations regarding a patient case [15]. Current methods are limited in that they do not sufficiently exploit medical knowledge [5,6]. In addition, most methods formulate the problem as a classification task and assume diagnosis classes are independent: this is a problem as medical conditions are instead related (e.g., hierarchy of conditions, causality between conditions – see Section 2 for details).

We posit that the exploitation of medical knowledge, in particular as encoded in medical KGs, within an end-to-end deep learning architecture for diagnosis identification may improve the effectiveness of current automated medical diagnosis systems. To this end, we propose a Knowledge Graph Convolutional Network (KGCN) method for ranking diagnosis (Section 3), that exploits medical KGs to enable capturing insightful diagnosis patterns. In our method, a patient's symptoms are identified within the KG and used to derive likely diagnoses (diseases) for the patient based on the representations of medical concepts and their relationships encoded in the KGs. We use the concept of message diffusion in Graph Convolutional Networks (GCN) [9,17] to model the relationships between symptoms and diseases encoded in the KG. Specifically, we inject a special node - patient node - to the medical KG and connect its symptoms to it (see Fig. 1). We refer to

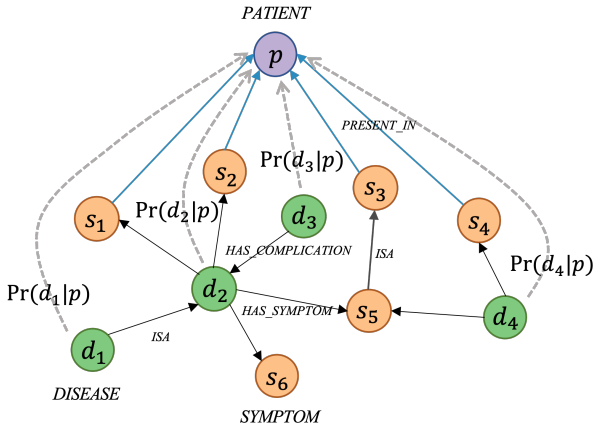


Fig. 1: Exemplified medical KG. Concepts (nodes) belong to different types (e.g., symptoms (orange), diseases (green)) and are linked by various relationships, e.g., ISA, $HAS\_SYMPTOM$.

the formed graph as *diagnosis graph* and each node in this graph has an initial representation. We then employ stacked GCN layers to the diagnosis graph to learn, for each node, a comprehensive representation. Through the message-passing mechanism of GCN, nodes share their information with their neighbours and meanwhile aggregate the received information from their neighbours. By stacking l GCN layers, the nodes can receive messages from their l-hop neighbours. This allows to use different types of relations and multi-hop contexts of nodes. We experiment with different fusion functions to study the most effective way of aggregating context information within a node. After obtaining comprehensive representations of disease concepts and patients, we predict the likelihood of a disease node to be connected to the patient node (link prediction) with a match model. Finally, we use this inferred probability to rank diagnoses for a given patient case.

We have evaluated the proposed method on a dataset of realistic patient vignettes redacted by medical experts (Sections 4 and 5). Results show that our KGCN provides better diagnosis predictions than existing methods. We further tease out the impact of data sparsity, different medical relations, fusion functions, number of GCN layers, on the effectiveness of KGCN.

2 Related Work

Automatic medical diagnosis aims to assist clinicians with diagnosing patients by using computer algorithms to identify the most probable diagnoses for a patient, given their case description (disease history, symptoms, signs) [15].

Many Machine Learning algorithms have been explored to learn diagnosis patterns automatically from existing medical records to support this task [24,16,1,27], but often the learned models achieved limited effectiveness. This has been because of insufficient data being available and the fact that relationships between medical concepts not being modelled and exploited by these methods.

To improve effectiveness, recent methods have attempted to learn distributed representations of medical concepts, e.g., from ontologies or electronic health records [6,5], and use them to enhance predictive models. Other work has introduced prior medical knowledge in the form of knowledge graph [28] or rules [18] into models to improve the effectiveness of disease prediction. Though promising, also this line of work has limitations.

A first limitation is that existing work formulates medical diagnosis as a (multi-class) classification problem. The underlying assumption in doing so is that the classes (diseases) are assumed independent: this assumption is not true as often diseases are related e.g., due to presenting the same symptoms, being a more specific instance of a general condition, or being common co-morbidities. Adequately modelling this relatedness, instead, may likely allow for better discrimination among diagnoses and thus better diagnosis effectiveness. In this work we take a different stand by formulating medical diagnosis as a matching problem, where patient's descriptions (symptoms) and diagnoses are represented within a knowledge graph using rich features and are matched to produce a ranking of possible diagnoses, starting from the most likely.

Another limitation of previous work is that medical knowledge has often not been fully exploited. Medical knowledge has been extensively modelled by manually curated domain-specific resources such as medical ontologies and terminology, e.g., SNOMED CT [23], MedRA[1], UMLS [3], and automatically mined medical Knowledge Graphs (KGs), e.g., KnowLife [7], Rotmensch et al.'s [19], HighLife [8], etc. In Fig.1 we provide a schematic example of a Knowledge Graph in this context. While previous work has used such medical knowledge for diagnosis identification, this came with limitations. Some works [5,6] mainly focused on hierarchy information (i.e., *ISA*) and ignored other important relationships, such as *HAS_SYMPTOM* between disease and symptom, *HAS_COMPLICATION* between diseases, etc.. Some other works only considered to add direct contexts in KGs to the model but neglected multi-hop contexts. However, multi-hop contexts are common in medicine, often being used for modelling properties such as the transitivity of hierarchy or chains of relationships for causality. Fully relying on the extensive medical knowledge captured in these domain-specific resources, instead, may likely lead to better diagnosis effectiveness.

As mentioned above, our solution relies on a medical KG to estimate the match between a set of symptoms and the likely diagnosis. Three main avenues have been explored in the literature when relying on KGs for matching:

1. use knowledge graph embedding (KGE) algorithms to learn the vector representations of entities and relationships in a KG, and then use these within the downstream applications related to matching [10]. In this way, KGs are used independently of the end-task and thus their use is rather flexible. However, there is a mismatch between the goal of the KGE construction process, which is to encode the semantic relatedness among entities in the KG, and the end-task goal for which the learned embeddings are used, i.e. matching
2. identify various connection patterns among entities in a KG to exploit as additional matching signals. This provides intuitive methods that heavily rely on manually designed meta-graphs: these however are often hard to tune in practice.
3. integrate matching models and KGs in a hybrid graph and inject the structure information of KGs into the matching problem to form an end-to-end task. This solution can avoid the shortcomings of the first two alternatives described above.

The proposed KGCN follows the third solution, integrating the matching model and the KG in a hybrid graph to be used within an end-to-end pipeline.

Our proposed method relies on Graph Convolution Networks (GCN) [20,4,14], which generalized convolutional neural networks to non-Euclidean spaces such as a graph. The key idea of GCNs is to generate node embeddings through message passing or information diffusion processes executed on the graph [9].

[1] https://www.meddra.org/

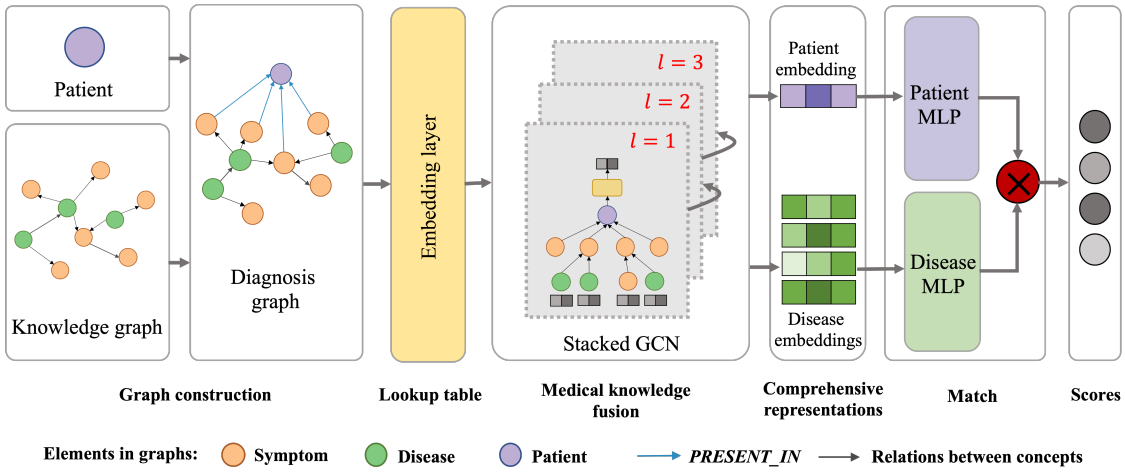


Fig. 2: Overview of our method. The framework consists of several stages: 1) construct the diagnosis graph by linking the patient to the medical KG, 2) fuse medical knowledge using stacked GCN layers to obtain a comprehensive representation of each node, 3) transform the new representations of patient and disease nodes into the same latent space using MLP layers and obtain similarity scores using the inner product.

3 Knowledge Graph Convolution Networks for Diagnosis Ranking

Figure 2 provides an overview of our model. In particular, we add a special node to an existing medical KG to form a diagnosis graph, in which the patient node is linked to nodes representing the symptoms exhibited by the patient (as described in the patient case vignette). GCN is then adopted to learn comprehensive representations of the patient and medical concepts. Finally, we predict the likelihood that a disease node may be linked to the patient node and rank diagnoses based on the probability distribution with respect to the patient case. We elaborate on each component of the proposed model in the following.

3.1 Problem Formulation

Medical diagnosis is the process that attempts to determine the disease $d \in \mathbb{D}$ ($\mathbb{D}$ being the set of possible diseases) affecting a patient p who exhibits a set of symptoms $p = \{s_1, s_2, ..., s_n\}, s_i \in \mathbb{S}$ ($\mathbb{S}$ being the set of possible symptoms). We refer to the pair (p, d) as a *case*. To assist the diagnosis process, we exploit a medical KG $\mathscr{K} = \{(h, r, t)|h, t \in \mathbb{D} \cup \mathbb{S}, r \in \mathbb{R}\}$, where $\mathbb{R}$ is the set of relations between medical concepts. The KG is essentially a directed heterogeneous graph.

In the learning process, some cases $\mathrm{Y} = \{(p_i, d_i)\}, 0 \leq i \leq |Y|$ are provided for training the model, with the goal to derive a prediction function $y_{p,d} = \mathscr{F}(p, d \mid \Theta, \mathscr{K}, \mathrm{Y})$. Here, $y_{p,d}$ represents the probability that the disorder d is the true diagnosis for patient p, and Θ denotes the parameters of the prediction function $\mathscr{F}$. In the diagnosis process, given a patient with symptoms, the model uses $\mathscr{F}$ to obtain his matching score with each disease $d \in \mathbb{D}$ and outputs a ranked disease list.

3.2 Construction of the Diagnosis Graph

We construct the diagnosis graph $\mathscr{G}$ by injecting the patient node p to an existing medical KG $\mathscr{K}$. In this paper, we use a subset of SemmedDB [13] as the KG. SemmedDB contains a large amount

of predications extracted from biomedical texts (scientific articles); our subset only contains the triples whose head and tail entities are symptom or disease concepts and the relation is of type *isa* or *causes*[2]. To construct the diagnosis graph, we create a special patient node, identify the symptoms of the patient in the KG, and link these symptom nodes to the patient node with edges of type *present_in* [3]. The obtained diagnosis graph is denoted as $\mathscr{G} = \{(u, e_{uv}, v)|u, v \in \mathbb{D} \cup \mathbb{S} \cup \{p\}, e_{uv} \in \{present\_in, causes, isa\}\}$.

3.3 Embedding Layer

The embedding layer is used to assign an initial vector representation to each node in the diagnosis graph with a look-up table operation. Every concept node $c \in \mathbb{D} \cup \mathbb{S}$ is assigned a corresponding embedding $\boldsymbol{h}_c \in R^{N^0}$ while different patients share a single initial representation $\boldsymbol{h}_p \in R^{N^0}$. The embedding matrix is:

$$\boldsymbol{E}^{(1+|\mathbb{D}|+|\mathbb{S}|)\times N_0} = [\underbrace{\boldsymbol{h}_p}_{\text{patient}}, \underbrace{\boldsymbol{h}_{d_1}, ..., \boldsymbol{h}_{d_{|\mathbb{D}|}}}_{\text{disease}}, \underbrace{\boldsymbol{h}_{s_1}, ..., \boldsymbol{h}_{s_{|\mathbb{S}|}}}_{\text{symptom}}]. \tag{1}$$

These embeddings are initialized randomly and optimized in an end-to-end fashion.

3.4 Medical Knowledge Fusion Layer

The medical knowledge fusion layer is designed based on GCN, which employs message-passing architecture to capture the relatedness between medical concept nodes. In this process, the patient node also obtains its representation by fusing the symptoms and the potential causes of those symptoms. In the following, we first illustrate the first-order knowledge fusion and then generalize to high-order knowledge fusion.

First-order Medical Knowledge Fusion. Within a single GCN layer, the message-passing process has two stages: (1) each node constructs messages and sends them to its neighbours through the outbound edges. The content of each message depends on the information contained in the source node, the type of edge, the information contained in the destination node. (2) each node aggregates the received messages from all inbound edges and fuses them with the information it contains.

Message Construction. The message sent from node u to v is represented by $\boldsymbol{m}_{u\to v} = f^{conc}(\boldsymbol{h}_u, r_{uv}, \alpha_{uv})$, where r_{uv} is the type of edge e_{uv}, α_{uv} is the decay factor of passing a message on edge e_{uv}, and $f^{conc}(\cdot)$ is the message construction function which takes the representation of node u, the edge type r_{uv} and the decay factor α_{uv} as input. In this work, we implement $f^{conc}(\cdot)$ as:

$$\boldsymbol{m}_{u\to v} = \alpha_{uv}(\boldsymbol{W}_{r_{uv}}\boldsymbol{h}_u + \boldsymbol{b}_{r_{uv}}), \tag{2}$$

where $\boldsymbol{W}_{r_{uv}} \in \boldsymbol{R}^{N^0\times N^1}$ and $\boldsymbol{b}_{r_{uv}} \in \boldsymbol{R}^{N^1}$ are trainable parameters to distill useful information for propagation.

[2] Note that the relation *causes* in SemmedDB is rather coarse and encompasses relations that would normally be treated as separate in other medical KGs, including relations such as *has_complication*, *has_symptom*.

[3] We link a patient with the KG through the symptoms' Concept Unique Identifiers (CUIs). Medical concept recognition tools like QuickUMLS [22] and MetaMap [2] can recognize and map terms in patients' records to CUIs; each entity in the medical KG is represented by a CUI.

Message Aggregation. We aggregate the received messages at node v by summing them as $\boldsymbol{a}_v = \sum_{u' \in \mathscr{N}_v} \boldsymbol{m}_{u' \to v}$, where $\mathscr{N}_v$ is the set of neighbours. Then, we fuse the aggregated context $\boldsymbol{a}_v$ with the node $\boldsymbol{h}_v$ itself as $\boldsymbol{h}_v^{(1)} = f^{fuse}(\boldsymbol{h}_v, \boldsymbol{a}_v)$, where $f^{fuse}(\cdot)$ is the fusion function. In this work, we exploit *GRU* as the fusion function as done by Li et al. [17]:

$$\boldsymbol{h}_v^{(1)} = GRU(\boldsymbol{h}_v, \boldsymbol{a}_v). \tag{3}$$

Comparison of Context Fusion Methods. The fusion function is a key component of our method since it determines if the context information can be effectively introduced. Intuitively, a node eagerly seeks to incorporate context when its representation is not informative enough, and its context can provide beneficial information. The way in which the context is to be fused with the node should depend on the representation of the node itself, the messages received from the context, and their interaction. In our method, we use GRU as the fusion function because its model structure can support this intuition. As comparison methods, we also implemented two alternative fusion functions, which are comparatively simple even though widely used in other tasks – these are described next.

SumFus takes the summation of two context vectors, followed by a non-linear transformation: $\boldsymbol{h}_v^{(1)} = \sigma(\boldsymbol{W}^{sg}(\boldsymbol{a}_v + \boldsymbol{h}_v) + \boldsymbol{b}^{sg})$,, where $\boldsymbol{W}^{sg}$ and $\boldsymbol{b}^{sg}$ are the parameters, σ is the activation function.

ConcatFus concatenates two context vectors first before non-linear activation $\boldsymbol{h}_v^{(1)} = \sigma(\boldsymbol{W}^{cg}(\boldsymbol{a}_v \oplus \boldsymbol{h}_v) + \boldsymbol{b}^{cg})$,, where $\oplus$ is the concatenation operation, $\boldsymbol{W}^{cg}$ and $\boldsymbol{b}^{cg}$ are the parameters, σ is the activation function.

High-order Medical Knowledge Fusion First-order context aggregation is primary for our medical diagnosis model since only symptom concepts are connected to the patients. To make the patient aware of the potential causes of the symptoms he shows, we need to do high-order context aggregation. By stacking l context aggregation layers, one node in the graph can receive messages propagated from l-hop neighbours. Formally, we repeat the context aggregation process by applying graph convolution operation on the graph and use the context vectors obtained from $(l-1)$-th GCN layer as the node representations, as in equation

$$\boldsymbol{m}_{u \to v}^{(l)} = \alpha_{uv}(\boldsymbol{W}_{r_{uv}}^{(l)} \boldsymbol{h}_u^{(l-1)} + \boldsymbol{b}_{r_{uv}}^{(l)}) \ . \tag{4}$$

Then, the new context representation of node v is obtained by aggregating the received messages from its neighbours $u' \in \mathscr{N}_v$ and fusing it with $\boldsymbol{h}_v^{(l-1)}$:

$$\boldsymbol{a}_v^{(l)} = \sum_{u' \in \mathscr{N}_v} \boldsymbol{m}_{u' \to v}^{(l)} \ , \quad \boldsymbol{h}_v^{(l)} = GRU(\boldsymbol{h}_v^{l-1}, \boldsymbol{a}_v^{(l)}) \ . \tag{5}$$

Here, $\boldsymbol{W}^l \in \boldsymbol{R}^{N^{l-1} \times N^l}$, $\boldsymbol{b}^l \in \boldsymbol{R}^{N^l}$ are trainable parameters in the l-th GCN layer.

3.5 Feature Transformation and Matching

After aggregating the medical knowledge with L GCN layers, each node obtained a comprehensive representation, which entails its original representation as well as the aggregated context information at each GCN layer. At the matching stage, we transform the patient node and disease nodes using MLP layers separately to get their final representation in the same latent space as $\boldsymbol{h}_p^o =$

$MLP^p(\boldsymbol{h}_p^{(L)})$, $\boldsymbol{h}_d^o = MLP^d(\boldsymbol{h}_d^{(L)})$. Both of the MLPs have hyper-parameters: the number of hidden layers and the unit number of each hidden layer. After getting the final representations of the patient and each disease concept, we conduct inner product to calculate their similarity score as $y_{d_i,p} = \boldsymbol{h}_p^{o\top} \boldsymbol{h}_d^o$. We can further apply the softmax to these similarity scores to get the probability $\Pr(d_i|p)$ that a certain disease d_i is the true diagnosis of the patient p.

3.6 Ranking Diagnosis

We can rank the diseases $d \in \mathbb{D}$ according to their matching scores with a certain patient and then return a ranked list of diseases. It should be noticed that the patient nodes only have inbound edges and thus have no effect on the contextual representations of medical concepts. Therefore, the contextual representations $\boldsymbol{h}_c^{(l)}, c \in \mathbb{D} \cup \mathbb{S}, 0 \leq l \leq L$ of medical concepts only have to be calculated once and then put in cache for subsequent usage.

3.7 Training Model

To learn the model parameters, we choose Ranking Cross-Entropy, which has been widely used in matching models, as the loss function. Specifically, for a given patient $p_i = \{s_j\}$ and his ground truth diagnosis d_i^T, we sample N diseases $\{d_{i,k}^F\}_{1 \leq k \leq N}$ randomly from the disease set $\mathbb{D} \setminus \{d_i\}$ as negative diagnoses. Then, we calculate their matching scores $y_{p_i d_i^T}$ and $\{y_{p_i d_{i,k}^F}\}_{0 \leq k \leq N}$. Afterwards, we apply softmax function on those scores and get their normalized probabilities

$$\begin{aligned} &[\Pr(d_i^T|p_i), \Pr(d_{i,1}^F|p_i), ..., \Pr(d_{i,N}^F|p_i)] \\ &= \text{softmax}(y_{p_i,d_i^T}, y_{p_i,d_{i,1}^F}, ..., y_{p_i,d_{i,N}^F}). \end{aligned} \tag{6}$$

The cross entropy loss of training instance (p_i, d_i^T) is formulated as $loss_{p_i} = -\log \Pr(d_i^T|p_i)$. For a batch of training instances $\{(p_i, d_i^T)\}$, the batch loss is

$$Loss = -\sum_i \log \Pr(d_i^T|p_i) + \lambda \parallel \Theta \parallel^2 , \tag{7}$$

where the L2 norm of parameters are added with factor λ. Besides, we adopt min-batch Adam to optimize the model and update the parameters.

4 Experimental Setup

4.1 Dataset and Evaluation Measures

Training data. Although ML is now widely used to assist with numerous medical tasks, publicly available datasets are limited. To train the proposed method we require datasets containing patient cases, consisting of reports of symptoms and associated diagnoses. The MIMIC-III [12] and the TREC Medical Records [26] datasets both contain patient records and associated diagnoses. However, MIMIC III data contains little information about symptoms, and the diagnosis codes (in ICD) do change over time during the patient encounter (no discharge diagnosis is recorded). MIMIC III also presents a strong bias in that the records relate to intensive care unit hospitalisations only. The TREC Medical Records dataset contains descriptions of complaints and symptoms for each patient encounter along with diagnoses (also at discharge); however it is not any more publicly available.

Previous work by Xia et al. [27] has shown that the abstracts from biomedical literature articles contain descriptions of diseases and associated key symptoms can be used for disease diagnosis. Motivated by this observation, we then constructed training instances from medical literature abstracts, following a similar procedure to that used by Xia et al. [27]. Specifically, we acquired biomedical abstracts annotated with UMLS concepts, made available from Medline 2019[4]. Then, we only selected articles associated with diseases and symptoms. Finally, we generated several cases from each abstract using the occurring symptoms as the description of patients and each occurring diseases as the possible diagnoses.

Test collection. To test the effectiveness of automated diagnosis methods, we constructed a test collection using the free-text vignettes from a previous work that evaluated the correctness of symptom checkers [21]. These vignettes were sourced from clinical notes and text-book cases; each vignette contains a brief free-text description of the patient, a diagnosis made by a clinician, and a triage urgency (three levels: *emergent care is required*, *non-emergent care is reasonable*, and *self care is sufficient*).

In our collection, a test instance was constructed using a vignette by extracting symptom concepts from the patient's free-text description and mapping the free-text of the correct diagnosis provided for the patient case to a disease concept, using QuickUMLS [22], a tool that performs unsupervised biomedical concept extraction from free-text. When assembling our collection, we had to exclude two of the vignettes from the original dataset by Semigran et al. [21] as the free-text associated with the correct diagnosis could not be mapped to any disease concept by QuickUMLS. In total, 43 test instances were obtained for evaluation.

Limitation of experiments. Our experimental findings are limited by the following factors: 1) the used test collection is small – this aspect makes it less likely experiments will detect statistical significant differences between methods 2) clinical notes are not available as training data and thus there may be a mismatch between training and test data, 3) the public medical KG we are using is noisy.

Evaluation metrics. For each vignette, the ground truth contains only one correct diagnosis. In addition, when considering the medical diagnosis task, it is likely that end-users may be wanted only to consider a handful of diagnoses: the cognitive load of considering a large array of diagnoses would render a clinical decision support application for diagnosis recommendation not worth it. These characteristics are akin to the problem of known-item retrieval, with a strong preference on early rank retrieval, if not even a dismissal of results above a certain rank cut-off. With this in mind, we select $hit@k$ (with $k = 1, \ldots, 5$) as evaluation metrics for our experiments – $hit@k = 1$ if the correct diagnosis is ranked among the top k results, 0 otherwise. We also include $nDCG@k$ in our evaluation. While we do not have graded relevance in our task at the moment, this may be introduced in the future if approximate matching of ground truth diagnosis was added. For example, a diagnosis may be considered as *partially* correct if it is a specification or generalisation of the ground truth diagnosis (e.g., tension headache vs. headache). Nevertheless, $nDCG@k$, unlike $hit@k$, does assign a discount to the rank position at which the correct diagnosis is retrieved, and thus it rewords methods that retrieve the correct diagnosis early in the ranking.

4.2 Baselines

To contextualise the effectiveness of the proposed method, we implemented a number of baseline systems for the disease diagnosis task. Naïve Bayes Classifier (NB) [27] and Multiple Layer Per-

[4] https://mbr.nlm.nih.gov/Download/MetaMapped_Medline/2019/MMO/

ceptron (MLP) [24] are two simple baselines commonly used for the disease prediction task. NB assumes all medical concepts are independent of each other, while MLP, as a multi-class classification model, assumes the disease concepts are independent. Deep Structured Semantic Models (DSSM) [11] is a representative neural matching model, which represents medical concepts as vectors, and then, similar to our method, matches a group of symptoms (associated to a patient) with disease concepts to obtain an overall similarity score, which is then used to rank diagnoses. ContextCare treats diagnosis ranking as a link prediction problem, similarly to what we do, but models the diagnosis pattern with an energy function, a popular method for link prediction task. The Graph-based Attention Model (GRAM) [5] and LSTM-KGAtt [28] address the task of risk prediction, e.g., mortality risk prediction, using time series data regarding the progression of the patient picture. We adapt these methods to the diagnosis prediction (ranking) task considered in this paper. GRAM obtains representations of medical concepts by combining their hierarchy information (ancestors) within their representations. LSTM-KGAtt incorporates the direct context of medical concepts in KG into the diagnosis process using the attention mechanism.

4.3 Parameter Settings

The GCN was implemented using Python 3.7, PyTorch 1.3.1 and DGL 0.4.3 (`https://docs.dgl.ai/`). The hyper-parameters were selected using the following strategies. The dimension of concept embeddings and node features in the graph share a single value. The number of hidden layers and the unit numbers of hidden layers in the two MLP modules are set to the same value. The hyper-parameters were optimised using grid-search and 5-fold cross-validation. The number of GCN layers was chosen from $\{1, 2, 3, 4, 5\}$, the dimension of features was selected from $\{100, 200, 400\}$, the number of MLP hidden layers was tuned in $\{0, 1, 2\}$, the unit number of MLP hidden layers was tuned amongst $\{100, 200, 400\}$, the dropout rate was chosen from $\{0.0, 0.1, 0.3, 0.5\}$. The learning rate was set as $1e^{-3}$ and reduces when the validation loss stops decreasing. The number of negative samples for matching was set to 1000 and the kaiming initializer was used to initialize the model parameters.

5 Results and Analysis

With our empirical experiments, we aimed to answer the following research questions related to the proposed KGCN method:

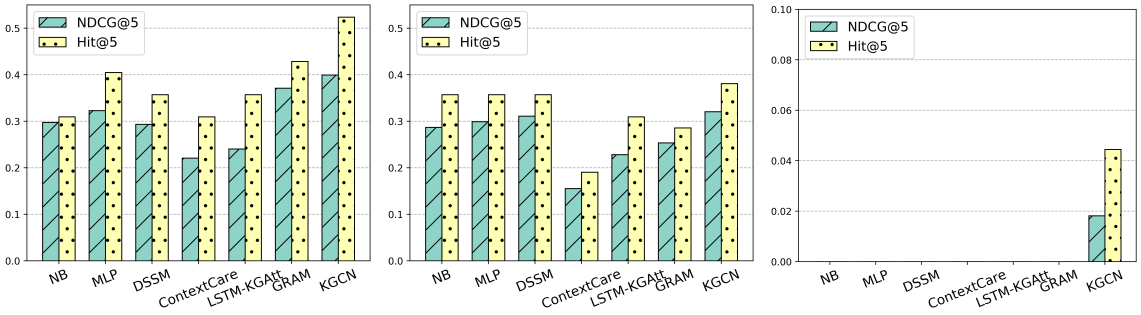
RQ1: Does our KGCN method outperform the baselines?
RQ2: How does our KGCN method perform with respect to the level of urgency of the patient case (triaging)?
RQ3: How does relationship type affect the effectiveness of our KGCN method?
RQ4: How does the fusion function affect the effectiveness of our KGCN method?
RQ5: How does the number of GCN layers affect the effectiveness of our KGCN method?

5.1 RQ1: Overall effectiveness

Table 1 reports the overall effectiveness of each method. Note that none of the differences are statistically significant (paired t-test, $alpha = 0.05$); this is likely due to the limited number of vignettes and to all methods not identifying a correct diagnosis for a subset of cases (self-care vignettes, see Section 5.2) and thus obtaining the same evaluation scores in these cases.

Table 1: Overall effectiveness of methods for diagnosis rannking. The proposed KGCN achieved the best effectiveness across all metrics.

	Hit@1	Hit@2	Hit@3	Hit@4	Hit@5	NDCG@1	NDCG@2	NDCG@3	NDCG@4	NDCG@5
NB	0.1473	0.2093	0.2171	0.2171	0.2171	0.1473	0.1864	0.1903	0.1903	0.1903
MLP	0.1550	0.1860	0.2171	0.2248	0.2481	0.1550	0.1746	0.1901	0.1934	0.2024
DSSM	0.1550	0.1860	0.2171	0.2326	0.2326	0.1550	0.1746	0.1901	0.1968	0.1968
CtxCare	0.0775	0.1163	0.1318	0.1473	0.1628	0.0775	0.1020	0.1097	0.1164	0.1224
LSTM-KGAtt	0.0775	0.1473	0.1705	0.2093	0.2326	0.0775	0.1215	0.1332	0.1499	0.1589
GRAM	0.1550	0.2016	0.2326	0.2481	**0.2636**	0.1550	0.1844	0.1999	0.2066	0.2126
KGCN	**0.1783**	**0.2248**	**0.2403**	**0.2558**	**0.2636**	**0.1783**	**0.2076**	**0.2154**	**0.2221**	**0.2251**



(a) *Emergent care is required.* (b) *Non-emergent care is reasonable.* (c) *Self care is sufficient.*

Fig. 3: Effectiveness with respect to level of urgency. Note that all methods cannot find a correct diagnosis among the top 5 ranks for any of the self-care scenarios, apart from our KGCN, which does retrieve the correct diagnosis for a handful of self-care vignettes.

NB and MLP, which are representative traditional methods for disease diagnosis, provided quite good effectiveness, especially when compared with more complex methods.

DSSM obtained similar performance to MLP, suggesting that formulating disease diagnosis as a matching problem does not effect effectiveness, while though offering greater flexibility in the way external knowledge can be incorporated.

ContextCare obtained the worst result: this highlights the limitation of the energy function in the diagnosis ranking task.

LSTM-KGAtt also performed poorly, although this method relied on the medical KG and thus exploits medical knowledge. This may be because the underlying LSTM architecture is not suitable for this task, even though it is widely adopted for tasks such as disease progression task.

GRAM provided improvements over NB, MLP and DSSM. This is done by exploiting the hierarchy information associated with medical concepts; a characteristic that simpler deep learning methods like MLP and DSSM do not model.

Finally, our model achieved the highest effectiveness across all metrics. Compared with MLP, our method is more flexible in that it exploits relationships between medical concepts. When compared with DSSM, we observe that our model does make effective use of the KG. Unlike GRAM, which only models hierarchy relationships, our method can model different types of knowledge in the medical KG: the empirical comparison with GRAM shows this is an important factor.

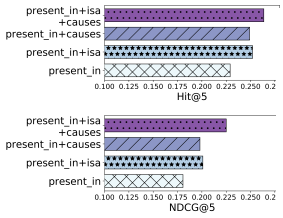


Fig. 4: Effect of Medical Knowledge.

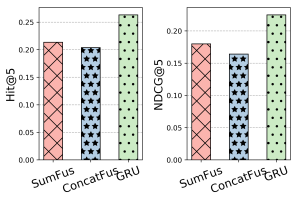


Fig. 5: Effect of fusion functions.

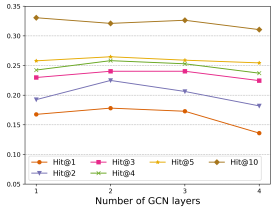


Fig. 6: Effect of the number of GCN layers.

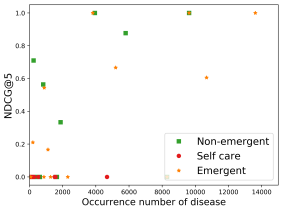


Fig. 7: Correlation between effectiveness and training data size.

5.2 RQ2: Effectiveness with respect to the level of urgency (triaging)

We further analyse the empirical results by considering the level of urgency (triaging) of each patient case. The results of our analysis are shown in Fig. 3 and suggest that KGCN outperforms other methods across all urgency levels. It also highlights how the effectiveness of the diagnosis ranking methods largely varies across the different levels of urgency, regardless of the actual method used. In particular, we find that all methods performed poorly for patient cases that required self-care, while they did perform well for the emergent and non-emergent care cases (vignettes).

We further analysed the results to understand why this may have been the case. In particular, we considered the number of occurrences of the target disease concepts used by the ground truth diagnoses in the vignettes. Specifically, we studied whether the effectiveness of KGCN was correlated with the number of such disease concepts in the data used for training (the analysis provided similar results for the other methods). Results are reported in Fig.7 and suggest that the more a target disease concepts occurred in the training data, the better the KGCN performed on the associated patient case (vignette). We further analysed these results with respect to the level of urgency associated with each vignette. Diseases that require self-care were typically rare in the training data and indeed KGCN performed poorly on this type of patient cases. Conversely, diseases that require emergent and non-emergent care occurred more frequently in the training data, and our KGCN obtained higher effectiveness on these types of cases.

5.3 RQ3: Effect of relationship type

To explore the effect of the type of relationships (edges) present in a medical KG, we execute the proposed KGCN method on medical KGs populated with different combinations of relationship types. Our experiments considered three relationship types: *isa*, *present_in* and *causes*. The results of this comparison are reported in Fig. 4. When only *present_in* was used, our method performed worst. When adding to this relationships either *isa* or *causes*, effectiveness increased. This suggests that both hierarchy information and causality are helpful relationships for medical diagnosis. The best effectiveness is however achieved when all relationships are considered (*present_in+isa+causes*): this is likely because hierarchy and causality provide complementary information.

5.4 RQ4: Effect of fusion function

A key component of the proposed KGCN is the fusion of knowledge of different orders. To do so, our method relies on *GRU* as the fusion function, although we have indicated how other two widely

used fusion functions, *SumFus* and *ConcatFus*, can also be used. In the next set of experiments, we compared the effectiveness of GRU compared to the two alternatives.

Empirical results related to this comparison are shown in Fig. 5. According to the results, the *GRU* substantially outperformed *SunFus* and *ConcatFus*, with the latter being the worst-performing fusion function amongst the three considered. performs the worst.

These results may be due to the fact that the architecture design of the *GRU* allows the parameters in low layers to be optimized better than when using the two alternative fusion functions. This caters to the fact that, for medical diagnosis, low-order information is more preferable than high-order knowledge. For example, if a clinician could have diagnosed a case simply by the symptoms, without considering the relationships between symptoms and conditions, they would not require the complex reasoning that underpins medical diagnosis. Another explanation for these results may be that the *GRU* fuses the representation of the input node and the aggregated context using their content interaction, while *SunFus* and *ConcatFus* can only combine them linearly. This advantage renders the model able to fuse these variables according to their contents. For instance, if the medical concepts do not have good representations, more medical knowledge would be needed.

5.5 RQ5: Effect of number of GCN layers

Finally, we analyzed the effect of the number of GCN layers in KGCN, while keeping the other hyper-parameters fixed. Overall, the KGCN method performs best when using two GCN layers, as shown in Fig. 6, while more GCN layers led to a decrease in diagnosis effectiveness. These results can be explained by that it is beneficial to aggregate more broad context to the representations of medical concepts and the patient in the disease diagnosis process. When the number of GCN layers is 3 or more, however, more noise is introduced; in addition, a model with more layers makes optimization more challenging.

6 Conclusions

In this paper we proposed a Knowledge Graph Convolutional Networks model, named KGCN, for ranking diagnosis. This method exploits medical KGs, which contain rich relations between medical concepts, in a more effective and general way compared with existing approaches. We formulated the disease diagnosis as a matching problem instead of a classification problem (as done in most of the previous work). To aggregate the medical knowledge for each concept in the KG and surface it with respect to the patient case at hand (patient node in the diagnosis graph), we exploited the message-passing mechanism of GCN to learn comprehensive concept representations. By stacking GCN layers, our model can propagate multi-hop contexts to each node.

Experiments were executed to assess the effectiveness of KGCN and tease out the aspects that influence its effectiveness. Our method outperformed existing approaches and we showed that both hierarchy and causality relationships provide complementary, valuable information for the diagnosis ranking task. We also compared different fusion functions in the context of KGCN, showing that the *GRU* fusion function outperformed the alternatives, and investigated the effect of the number of GCN layers and the availability of training data regarding the target ground-truth diagnosis had on effectiveness.

Our future work will consider two directions: (1) acquire more patient vignettes for evaluation, also including partial matches between diagnoses; (2) design special message-passing mechanisms within the GCN architecture for disease diagnosis. For example, we will explore a message-passing model with multiple channels to maintain the transitivity of hierarchy and causality relationships. Along this line, we will also consider exploiting a wider array of relationships.

Acknowledgements. This research is supported by the Shenyang Science and Technology Plan Fund (No. 20-201-4-10), the Member Program of Neusoft Research of Intelligent Healthcare Technology, Co. Ltd.(No. NRMP001901)). A/Prof Guido Zuccon is the recipient of an Australian Research Council DECRA Research Fellowship (DE180101579) and a Google Faculty Award.

References

1. Amato, F., López, A., Peña-Méndez, E.M., Vaňhara, P., Hampl, A., Havel, J.: Artificial neural networks in medical diagnosis (2013)
2. Aronson, A.R., Lang, F.: An overview of metamap: historical perspective and recent advances. J. Am. Medical Informatics Assoc. **17**(3), 229–236 (2010)
3. Bodenreider, O.: The unified medical language system (umls): integrating biomedical terminology. Nucleic acids research **32**(suppl_1), D267–D270 (2004)
4. Bruna, J., Zaremba, W., Szlam, A., LeCun, Y.: Spectral networks and locally connected networks on graphs. In: Bengio, Y., LeCun, Y. (eds.) 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings (2014)
5. Choi, E., Bahadori, M.T., Song, L., Stewart, W.F., Sun, J.: GRAM: graph-based attention model for healthcare representation learning. In: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017. pp. 787–795. ACM (2017)
6. Choi, E., Xiao, C., Stewart, W.F., Sun, J.: Mime: Multilevel medical embedding of electronic health records for predictive healthcare. In: Bengio, S., Wallach, H.M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada. pp. 4552–4562 (2018)
7. Ernst, P., Siu, A., Weikum, G.: Knowlife: a versatile approach for constructing a large knowledge graph for biomedical sciences. BMC Bioinform. **16**, 157:1–157:13 (2015)
8. Ernst, P., Siu, A., Weikum, G.: Highlife: Higher-arity fact harvesting. In: Champin, P., Gandon, F.L., Lalmas, M., Ipeirotis, P.G. (eds.) Proceedings of the 2018 World Wide Web Conference on World Wide Web, WWW 2018, Lyon, France, April 23-27, 2018. pp. 1013–1022. ACM (2018)
9. Gilmer, J., Schoenholz, S.S., Riley, P.F., Vinyals, O., Dahl, G.E.: Neural message passing for quantum chemistry. In: Precup, D., Teh, Y.W. (eds.) Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017. Proceedings of Machine Learning Research, vol. 70, pp. 1263–1272. PMLR (2017)
10. Huang, J., Zhao, W.X., Dou, H., Wen, J., Chang, E.Y.: Improving sequential recommendation with knowledge-enhanced memory networks. In: Collins-Thompson, K., Mei, Q., Davison, B.D., Liu, Y., Yilmaz, E. (eds.) The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, SIGIR 2018, Ann Arbor, MI, USA, July 08-12, 2018. pp. 505–514. ACM (2018)
11. Huang, P., He, X., Gao, J., Deng, L., Acero, A., Heck, L.P.: Learning deep structured semantic models for web search using clickthrough data. In: He, Q., Iyengar, A., Nejdl, W., Pei, J., Rastogi, R. (eds.) 22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013. pp. 2333–2338. ACM (2013)
12. Johnson, A.E., Pollard, T.J., Shen, L., Li-Wei, H.L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G.: Mimic-iii, a freely accessible critical care database. Scientific data **3**(1), 1–9 (2016)
13. Kilicoglu, H., Shin, D., Fiszman, M., Rosemblat, G., Rindflesch, T.C.: Semmeddb: a pubmed-scale repository of biomedical semantic predications. Bioinform. **28**(23), 3158–3160 (2012)
14. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: 5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings. OpenReview.net (2017)

15. Kononenko, I.: Machine learning for medical diagnosis: history, state of the art and perspective. Artif. Intell. Medicine **23**(1), 89–109 (2001). https://doi.org/10.1016/S0933-3657(01)00077-X, https://doi.org/10.1016/S0933-3657(01)00077-X
16. Kononenko, I.: Machine learning for medical diagnosis: history, state of the art and perspective. Artificial Intelligence in medicine **23**(1), 89–109 (2001)
17. Li, Y., Tarlow, D., Brockschmidt, M., Zemel, R.S.: Gated graph sequence neural networks. In: Bengio, Y., LeCun, Y. (eds.) 4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings (2016)
18. Ma, F., Gao, J., Suo, Q., You, Q., Zhou, J., Zhang, A.: Risk prediction on electronic health records with prior medical knowledge. In: Guo, Y., Farooq, F. (eds.) Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD 2018, London, UK, August 19-23, 2018. pp. 1910–1919. ACM (2018)
19. Rotmensch, M., Halpern, Y., Tlimat, A., Horng, S., Sontag, D.: Learning a health knowledge graph from electronic medical records. Scientific reports **7**(1), 1–11 (2017)
20. Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M., Monfardini, G.: The graph neural network model. IEEE Trans. Neural Networks **20**(1), 61–80 (2009)
21. Semigran, H.L., Linder, J.A., Gidengil, C., Mehrotra, A.: Evaluation of symptom checkers for self diagnosis and triage: audit study. bmj **351**, h3480 (2015)
22. Soldaini, L., Goharian, N.: Quickumls: a fast, unsupervised approach for medical concept extraction. In: MedIR workshop, sigir. pp. 1–4 (2016)
23. Stearns, M.Q., Price, C., Spackman, K.A., Wang, A.Y.: SNOMED clinical terms: overview of the development process and project status. In: AMIA 2001, American Medical Informatics Association Annual Symposium, Washington, DC, USA, November 3-7, 2001. AMIA (2001)
24. Šter, B., Dobnikar, A.: Neural networks in medical diagnosis: Comparison with other methods. In: International conference on engineering applications of neural networks. pp. 427–30 (1996)
25. Stern, S.D.: Symptom To Diagnosis An Evidence-Based Guide Second Edition (2010)
26. Voorhees, E.M., Hersh, W.R.: Overview of the trec 2012 medical records track. In: TREC (2012)
27. Xia, E., Sun, W., Mei, J., Xu, E., Wang, K., Qin, Y.: Mining disease-symptom relation from massive biomedical literature and its application in severe disease diagnosis. In: AMIA Annual Symposium Proceedings. vol. 2018, p. 1118. American Medical Informatics Association (2018)
28. Yin, C., Zhao, R., Qian, B., Lv, X., Zhang, P.: Domain knowledge guided deep learning with electronic health records. In: Wang, J., Shim, K., Wu, X. (eds.) 2019 IEEE International Conference on Data Mining, ICDM 2019, Beijing, China, November 8-11, 2019. pp. 738–747. IEEE (2019)