Knowledge Guided Multi-instance Multi-label Learning via Neural Networks in Medicines Prediction

Junyuan Shang SJY1203@PKU.EDU.CN
Shenda Hong HONGSHENDA@PKU.EDU.CN
Yuxi Zhou JOY_YUXI@PKU.EDU.CN
Meng Wu WUMENG93@PKU.EDU.CN
Hongyan Li LIHY@CIS.PKU.EDU.CN
School of Electronics Engineering and Computer Science, Peking University, Beijing, China
Key Laboratory of Machine Perception (Peking University), Ministry of Education, Beijing, China

Editors: Jun Zhu and Ichiro Takeuchi

Abstract

Predicting medicines for patients with co-morbidity has long been recognized as a hard task due to complex dependencies between diseases and medicines. Efforts have been made recently to build high-order dependency between diseases and medicines by extracting knowledge from electronic health records (EHR). But current works failed to utilize additional knowledge and ignored the data skewness problem which lead to sub-optimal combination of medicines. In this paper, we formulate the medicines prediction task in multi-instance multi-label learning framework considering the multi-diagnoses as input instances and multi-medicines as output labels. We propose a knowledge-guided multi-instance multi-label networks called `KG-MIML-Net` where two types of additional knowledge are incorporated into a RNN encoder-decoder model. The utilization of structural knowledge like clinical ontology provides a way to learn better representation called tree embedding by utilizing the ancestors' information. Contextual knowledge is a global summarization of input instances which is informative for personal prediction. Experiments are conducted on a real world clinical dataset which showed the necessity to combine both contextual and structural knowledge and the `KG-MIML-Net` performs better than baselines up to 4+% in terms of Jaccard similarity score.

Keywords: Healthcare, Deep learning, Multi-Instance Multi-Label Learning

1. Introduction

Today abundant health data such as electronic health records (EHR) enables researchers and doctors to build better computational models for various healthcare related tasks (Xiao et al., 2018a). Among them, medicines prediction task considers how to make effective medicines prescription for patient with complicated conditions. For example, as shown in figure 1, two real patients in MIMIC-III dataset Johnson et al. (2016) are recorded as electronic health records (EHR) which consist of three parts including lab tests & demographics, diagnoses and prescriptions (medicines). To broaden the use of the designed algorithm, we formulate the medicines prediction task in multi-instance multi-label (MIML) learning framework Zhou et al. (2009) considering the multi-diagnoses as input instances and multi-medicines as output labels.

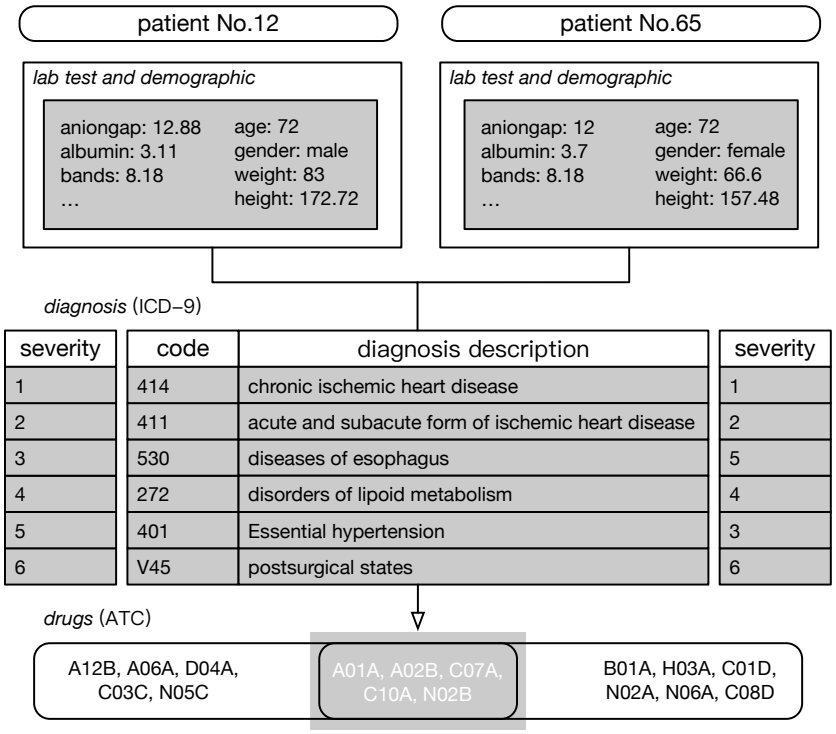


Figure 1: Two real patients' electronic health records in MIMIC-III dataset.

Thus, the medicines prediction task in MIML learning framework is to learn a function from EHR data to predict medicines given unseen diagnoses. As shown in figure 1, nine different medicines are predicted for patient No.12 given his six diagnoses including heart diseases (414), hypertension (401) and so on. When delving into this task, challenges come in two main aspects:

- **Complex dependencies** exists not only in the mapping between bag of instances and labels, but also among the instances or labels. On the one hand, there is an inner order among diagnoses by severity. Patient No.12 has more severe heart disease than postsurgical states and have more severe diseases of esophagus compared to patient No.65. On the other hand, the medicines are ordered by taken time. So it is crucial for the model to have the ability to learn complex dependencies, and contextual knowledge can be a summarization of instances to help the model perform better.

- **Data skewness** exists in both instance space and label space. The figure 2 shows the 50 most common diagnoses in MIMIC-III dataset. The less common seen diagnoses will not be fully trained which result in bad performance.

Given its importance, MIML learning methods have been successfully applied in many ares such as image classification Zha et al. (2008), relation extraction Surdeanu et al. (2012), video annotation Xu et al. (2011) and protein function prediction Wu et al. (2014). We broadly classify the existing MIML learning methods into two categories, traditional machine learning based approaches and deep learning based approaches which is described

in detail in related work. Briefly speaking, traditional machine learning based methods mentioned several clues to address the problem where complicated objects have multiple semantic meanings but most of them lack the ability to model high-order dependency and assume the representation of instances or labels are given first. Compared to traditional machine learning based approaches, deep learning based methods have shown their powerful abilities to learn robust representation and build complex dependencies which significantly outperform the traditional methods in text and image datasets.

Thus, to address above mentioned challenges and limitations, we propose a novel deep learning based knowledge guided MIML model called KG-MIML-Net. Instead of depending on previous given representation of instances or labels, we utilize the encoder-decoder framework which can jointly learn and update embedding for instances and labels and build mapping between bag of instances and bag of labels. The RNN structure is utilized as the implementation of both encoder and decoder to better capture high-order dependency among instances and labels as done in Sutskever et al. (2014). Moreover, a residual-supervised attention mechanism is embedded to assign weights to instances by their importance (severity). More importantly, two additional knowledge are extracted including contextual knowledge and structural knowledge. Contextual knowledge in medical area can be the personal summarization information like lab test and demographic. Structural knowledge like instances&labels ontology is a tree-structure classification scheme such as ICD-9 (shown in figure 3) in medical area which has been used in representation learning Choi et al. (2017) and concept linking Dai et al. (2018). In detail, we add a contextual layer after decoder to combine the personal contextual knowledge, and structural knowledge is utilized in a way that the representation of input instances as the leaf node in tree-structure classification scheme is learned depending on its ancestors'. The representation of ancestors will be generated by the mean of their direct children. A Bi-LSTM will output the tree-embedding given an instance and the tree-structure classification scheme.

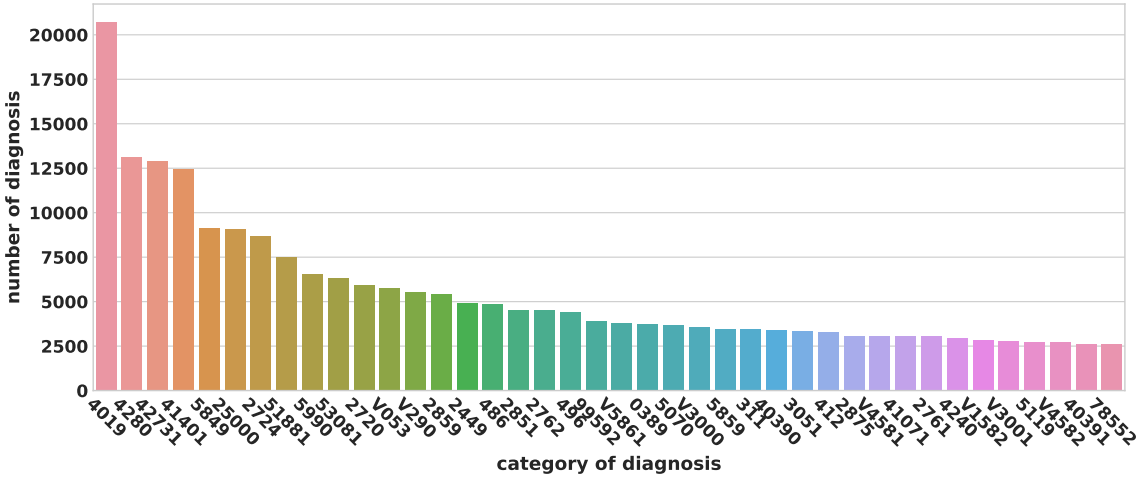
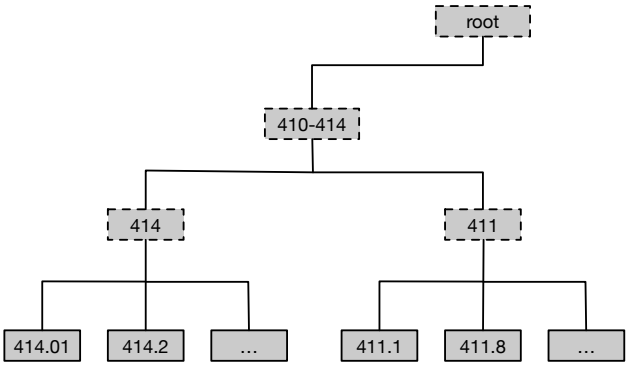


Figure 2: 50 most common diagnosis in MIMIC-III dataset

The contributions can be summarized as follows:

- We formulate the medicines prediction problem in MIML learning and demonstrate that the encoder-decoder model is a suitable choice for jointly modeling instances and outputs as well as building high-order dependency between them.
- A residual-supervised attention mechanism is proposed to assign different weights to instances and show better performance.
- We address the necessity to embed additional knowledge such as personal contextual knowledge and ontology structural knowledge to show better performance.
- We show the effectiveness of KG-MIMIL-Net compared with several state-of-the-art methods in MIML learning and traditional machine learning methods in real world clinical dataset.

In summary, the reminder of the paper will be organized as follows. In section 2, the related work will be given. Then problem formulation is introduced in section 3. The encoder-decoder model will be introduced first followed with tree embedding module and contextual layer module in section 4. The proposed model will be tested on the real world clinical dataset which demonstrate its effectiveness compared to traditional and recent state-of-art medicines prediction approaches in section 5. In section 6, a conclusion will be given.



ICD-9	Concept Meaning
410-414	Ischemic heart disease
414	Other forms of chronic ischemic heart disease
414.01	CAD NATIVE CORONARY VESSEL
414.02	CAD AUTOLOGOUS VEIN BYPASS GFT
411	Other acute and subacute form of ischemic heart disease
411.1	INTERMED CORONARY SYND
411.8	AC ISCHEMIC HRT DIS NEC

Figure 3: Hierarchical relation graph of ICD-9 ontology

2. Related Work

In multi-instance multi-label learning domain, various methods have been proposed. We categorize existing approaches into machine learning based and deep learning based approaches.

Machine learning based MIML approaches consist of methods from three aspects. Degeneration algorithms Zhou and Zhang (2007); Zhou et al. (2012) are the simplest which transformed and tackled the MIML task in multi-instance or multi-label learning framework. However, MIMLBOOST and MIMLSVM Zhou and Zhang (2007) degeneration methods will loss much information while transforming. Regularization based algorithms Zhang and Zhou (2008); Zha et al. (2008); Zhou et al. (2008); Li et al. (2017) find a way to consider the inner dependency among instances and labels by adding regularization term to the loss

function. D-MimlSvm Zhou et al. (2008) trained linear functions for every label and minimize the mean of these functions' weight as a regularization term to capture the dependency among labels. Specialized in image classification, mi-CRFs Li et al. (2017) captured latent probability distribution of instances, spatial context among adjacent instances and correlations between instances and labels into a conditional random fields (CRFs) framework. Joint learning approaches such as MIML-RE Surdeanu et al. (2012) and MIMLFast Huang and Zhou (2014) jointly model the instances and labels which are often combined with regularization based methods. However the above methods assume the representation of instances or labels are given in advance which lack the way to update the embedding for instances or labels and can not build high-order dependency between or among instances and labels. To address these issues, encoder-decoder RNN neural network is used to jointly model high-order dependency between and among instances and labels.

Deep learning based MIML approaches showed powerful representation learning ability from raw data. To list a few, DeepMIML Feng and Zhou (2017) exploited deep neural network formation to generate instance representation for MIML and showed better performance than traditional machine learning based MIML approaches on text and image data. MIML-FCN+ Yang et al. (2017) proposed a two-stream fully convolutional network with a novel Privileged Information(PI) loss which outperformed the state-of-the-art methods in the application of multi-object recognition. In healthcare area, the state-of-the-art method Leap Zhang et al. (2017) tackled the medicines prediction tasks by using a recurrent decoder to model label dependencies and content-based attention to capture label instance mapping. Besides the instances and labels associated to an object, we showed additional knowledge such as personal contextual knowledge and ontology structural knowledge can be used to make better model in medicines prediction task.

3. Problem Formulation

In this section, the definition of Knowledge-Guided Multi-instance Multi-label Learning will be given first which is the extension of Multi-instance Multi-label learning. Then an example based on figure 1 demonstrates the medicines prediction task in Knowledge Guided MIML learning framework.

Definition 1 (Knowledge-Guided Multi-instance Multi-label Learning) *The goal of Knowledge-Guided Multi-instance Multi-label Learning (KG-MIML) is to learns a function* $f : 2^{\mathcal{X}} \to 2^{\mathcal{Y}}$ *from training data* $\mathcal{D} = \{(X_1, Y_1), (X_2, Y_2), \cdots, (X_N, Y_N)\}$ *and knowledge* $(\mathcal{G}, \{\mathbf{c}_i, \boldsymbol{w}_i\}_1^N)$*, then it can predict the label set for a previously unseen bag, where* $X_i \subset \mathcal{X}$ *is a set of instances* $\{x_1^{(i)}, x_2^{(i)}, \cdots, x_{z_i}^{(i)}\}$*,* $x_j^{(i)} \in \mathcal{X}, (j = 1, 2, \cdots, z_i)$*,* $Y_i \subset \mathcal{Y}$ *is a set of labels* $\{y_1^{(i)}, y_2^{(i)}, \cdots, y_{l_i}^{(i)}\}$*,* $y_k^{(i)} \in \mathcal{Y}, (k = 1, 2, \cdots, l_i)$*,* $\mathcal{G} = (\mathcal{E}, \mathcal{R})$ *and* $\{\mathbf{c}_i, \boldsymbol{w}_i\}_1^N$ *are the structural knowledge and contextual knowledge set respectively. Here* z_i *is the number of instances in* X_i*,* l_i *is the number of labels in* Y_i*,* $\mathcal{E}$ *is the entity set,* $\mathcal{R}$ *is the relation set,* $\mathbf{c}_i$ *is personal context and* $\boldsymbol{w}_i$ *is instance weight for i-th object.*

For example, the No.65 patient can be represented as an object (X_i, Y_i) in $\mathcal{D}$ and X_i={'414', '411', '530', '272', '401', 'V45'}, Y_i ={A12B, N02B, A06A, C07A, D04A, C10A, A01A, C03C, A02B, N05C}. That's to say, the patient have $z_i = 6$ diseases and prescribed $l_i = 10$ medicines by doctors. Additional, the contextual knowledge consists of

Table 1: Notations

notation	description
$\mathcal{X}, \mathcal{Y}$	instance set and label set
$X_i \subset \mathcal{X}$	multi-instance set for i-th object
$Y_i \subset \mathcal{Y}$	multi-label set for i-th object
$\mathcal{D} \in \mathbb{R}^N$	training set $\{(X_i, Y_i)\}_1^N$
$x_j^{(i)} \in X_i$	j-th instance in i-th object's instances set
$y_k^{(i)} \in Y_i$	k-th label in i-th object's labels set
z_i	number of instances for i-th object
l_i	number of labels for i-th object
$\{\mathbf{c}_i, \boldsymbol{w}_i\}_1^N$	contextual knowledge set
$\mathbf{c}_i \in \mathbb{R}^d$	personal context for i-th object
$\boldsymbol{w}_i \in \mathbb{R}^{z_i}$	instance weight for i-th object
$\mathcal{G} = (\mathcal{E}, \mathcal{R})$	structural knowledge with entity set $\mathcal{E}$ and relation set $\mathcal{R}$
$e_i \in \mathcal{E}$	i-th entity
$p \in \mathcal{R}$	partent relation

personal context $\boldsymbol{c}_i$ and instance weight $\boldsymbol{w}_i$ like $\mathbf{c}_i = [12.88, 3.11, 8.18, \cdots, 72, 0, 83, 172.72]$ and $\boldsymbol{w}_i = [0.1, 0.3, \cdots, 0.5]$, the structural knowledge is the ontology such as ICD-9 tree described in figure 3. The medicines prediction problem is to predict Y_i given X_i and knowledge $(\mathcal{G}, \mathbf{c}_i, \boldsymbol{w}_i)$.

Specifically, structural knowledge in medicines prediction task is a kind of directed acyclic tree. Label set $\mathcal{Y} \subset \mathcal{E}$ are all entities in leaf nodes. The entities not in leaf nodes can be assumed as the virtual nodes which are high-level categories of their child entities. The relation $\mathcal{R} = \{p\}$ consists of only one relation called parent relation p, i.e, $p(e_i)$ is the parent of e_i. As shown in figure 3, the diseases '414.01' in ICD-9 code means 'CAD NATIVE CORONARY VESSEL'. p('414.01') can find the parent node '414' which means 'Other forms of chronic ischemic heart disease'. Further p('414') can find the parent node '410-414' which means 'Ischemic heart disease'. The parent information can be used to better learned representation for its child nodes.

For conciseness, the notations and their meanings can be found in table 1.

4. Method

As shown in figure 4, the overall model is correspondent with the RNN encoder-decoder model framework discussed in 4.1. We enhance the encoder module by adding supervised attention mechanism(B) discussed in which can pay different attention to the instances (diagnoses) and further enhance the ability of decoder by adding contextual layer(C) to fuse the personal contextual knowledge discussed in subsection 4.2. To tackle the data skewness problem, the tree embedding module(A) based on ontology structural knowledge is put before encoder and decoder which will be discussed in subsection 4.3. To reduce clutter, we will describe the algorithms for a single patient and drop the superscript (i) or subscript i whenever it is unambiguous.

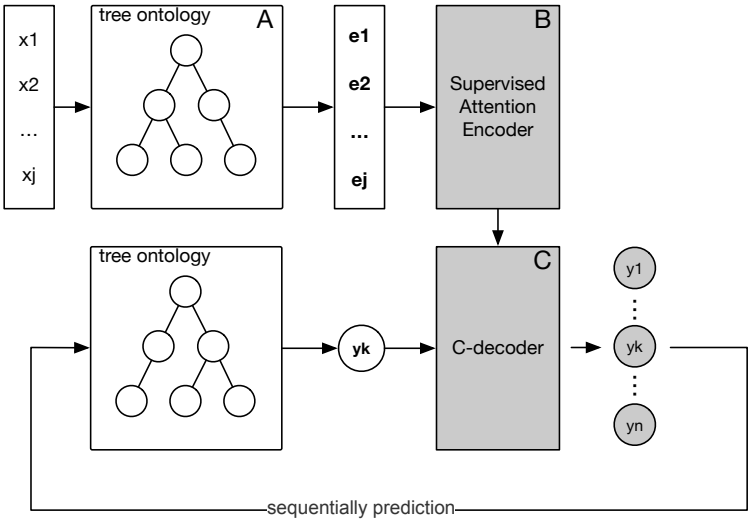


Figure 4: Overall framework of KG-MIML-Net.

4.1. Basic RNN encoder-decoder model

For completeness, we will briefly introduce the variational RNN called long short term memory (LSTM) and the encoder-decoder framework.

LSTM has been utilized in many areas like machine translation, time-series prediction. A common architecture for LSTM is composed of a memory cell, an input gate, an output gate and a forget gate. When LSTM is trained with backpropagation through time, the gradient will not vanish because its cell store the state for either long or short time periods. To model the relation among instances, we can input $\mathbf{x_j}$ into LSTM one by one, the $\mathbf{x_j}$ will be transformed to output state $\mathbf{o_j}$ depended on previous hidden state $\mathbf{h_{j-1}}$ as follows:

$$\begin{aligned}
\mathbf{f_j} &= \sigma_g(\mathbf{W}_j\mathbf{x_j} + \mathbf{U}_f\mathbf{h_{j-1}} + \mathbf{b}_f) \\
\mathbf{i_j} &= \sigma_g(\mathbf{W}_i\mathbf{x_j} + \mathbf{U}_i\mathbf{h_{j-1}} + \mathbf{b}_i) \\
\mathbf{o_j} &= \sigma_g(\mathbf{W}_o\mathbf{x_j} + \mathbf{U}_o\mathbf{h_{j-1}} + \mathbf{b}_o) \\
\mathbf{c_j} &= \mathbf{f_t} \odot \mathbf{c_{j-1}} + \mathbf{i_j} \odot \sigma_c(\mathbf{W}_c\mathbf{x_j} + \mathbf{U}_c\mathbf{h_{j-1}} + \mathbf{b}_c) \\
\mathbf{h}_j &= \mathbf{o}_j \odot \sigma_h(\mathbf{c}_j)
\end{aligned} \tag{1}$$

where the forget gate $\mathbf{f}$ controls the extent to which a value remains in the cell $\mathbf{c}$, the input gate $\mathbf{i}$ controls the extent to which a new value flows into the cell and the output gate $\mathbf{o}$ controls the extent to which the value in the cell is used to compute the output activation of LSTM. For simplicity, we represent the j-th output of LSTM for input instance $\mathbf{x_j}$ as follows:

$$\mathbf{h_j} = g(\mathbf{x}_j; \mathbf{h}_{j-1}) \tag{2}$$

where g is the simple form for LSTM unit of instances.

The encoder-decoder model firstly encodes a variable-length sequence into a fixed-length vector representation and then decodes a given fixed-length vector representation back into a variable-length sequence (e.g. $p(y_1, y_2, \cdots, y_k|x_1, x_2, \cdots, x_j)$. The instances are input to

encoder sequentially and encoder summaries all inputs to a context vector called $\mathbf{g}$. Then the decoder utilizes the context vector $\mathbf{g}$ to sequentially predict labels at step k as follows:

$$y_k = arg\max_{y \in \mathcal{Y}} p(y|y_1, y_2, \cdots, y_{k-1}, \mathbf{h}_{k-1}, \mathbf{g}) \tag{3}$$

The intuition to utilize RNN encoder-decoder is for two main reasons:

- RNN encoder-decoder is suitable for multi-instance multi-label learning. It naturally accepts multiple instances and predicts multiple labels sequentially which can jointly learn representation for both instances and labels.
- Relation among instances and labels is hard to build. RNN has powerful ability to build high-order dependency.

4.2. Residual-Supervised Attention Mechanism and Contextual Layer

We show the expanded view of RNN encoder-decoder framework in figure 4 as follows:

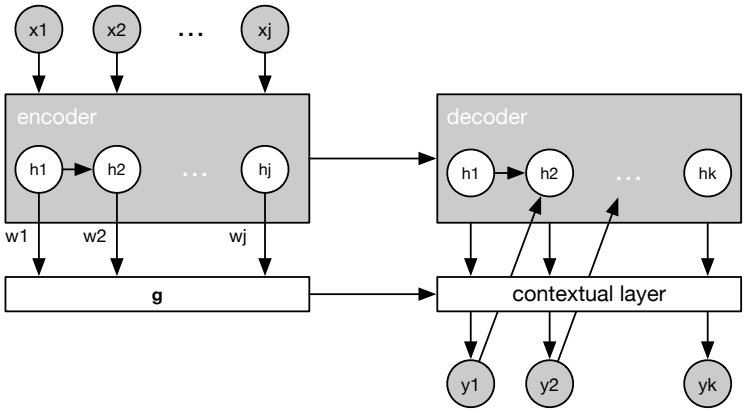


Figure 5: Residual-Supervised attention mechanism and Contextual layer.

Residual-Supervised attention mechanism as shown in figure 5 (left part) assigns different weight to different instances (diseases) in a supervised way as follows:

$$\mathbf{g} = \sum_{j}^{z} w_j \mathbf{h}_j^{enc} \tag{4}$$

where $\mathbf{h}_j^{enc} \in \mathbb{R}^d$ is the hidden state of encoder in j-th step and $w_j \in \mathbf{w}$ is the instance weight. The attention vector is connected to contextual layer behaving like the residual link He et al. (2016) which skips the decoder module to pass the supervised instances information to output space directly.

The contextual layer (CL) as shown in figure 5 right part fuses the contextual knowledge $\mathbf{c}$ into decoder's sequential outputs as follows:

$$\mathbf{o}_k = \mathbf{W}_c[\mathbf{c}, \mathbf{g}] + \mathbf{h}_k^{dec} \tag{5}$$

where $\mathbf{W}_c \in \mathbb{R}^{|\mathcal{Y}| \times 2d}$ is the transformation matrix to transform concatenated $[\mathbf{c}, \mathbf{g}]$ vector to decoder's output feature space, $\boldsymbol{h}_k^{dec} \in \mathbb{R}^{|\mathcal{Y}|}$ is the decoder hidden state and $\mathbf{o}_k$ is the k-th step contextual layer's output. A softmax layer can be added after $\mathbf{o}_k$ to make prediction.

The above mentioned two units will make medicines prediction personalized based on severity of diseases and personal context which improve the performance of medicines prediction and meanwhile make it more personalized.

4.3. Tree Embedding Module

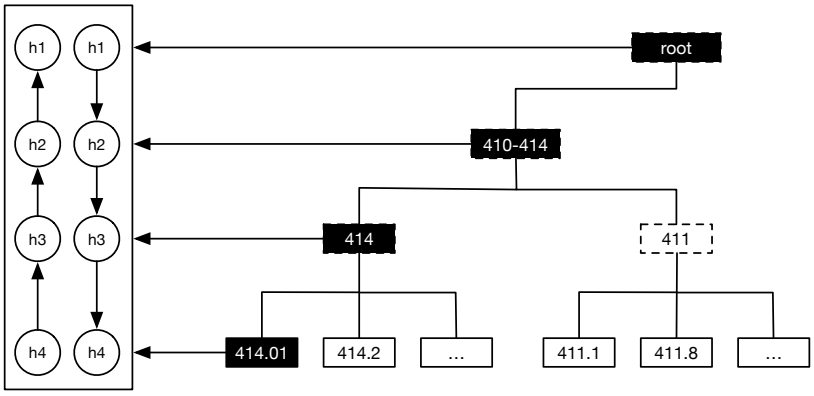


Figure 6: Tree Embedding.

The tree ontology is the inner structure in input space or output space which can be utilized to better learn tree embedding (TE) for fine-grained entity in leaf node using the ancestors' information as shown in figure 6.

In detail, the representation of the i-th leaf node is $\mathbf{x}_i$, the index of the parent's i-th leaf node can be found using relation function $p(\cdot)$. The representation of the parent node $\mathbf{x}_{p(i)}$ can be generated by its child as follows:

$$\mathbf{x}_{p(i)} = \frac{1}{N} \sum_{\{k|p(k)=p(i), 0 \leq k \leq |Y|\}} \mathbf{x}_k \tag{6}$$

The tuition to represent parent node as the mean sum of its child nodes' representation is that the parent node is a virtual node in the tree ontology which should have equal distance to every child node.

When every node's representation has been updated from the bottom to top, the target leaf node and its ancestors will be input into a bi-directional LSTM (Bi-LSTM) to generate a fixed-dimensional representation which captures the information of the target node and its ancestors.

$$\begin{aligned} \mathbf{h}_f^i, \mathbf{h}_b^i &= \text{Bi-LSTM}(\mathbf{x}_i, \mathbf{h}_f^{i-1}, \mathbf{h}_b^{i-1}) \\ \mathbf{e}_i &= [\mathbf{h}_f^i, \mathbf{h}_b^i] \end{aligned} \tag{7}$$

where $\mathbf{h}_f^i$, $\mathbf{h}_b^i \in \mathbb{R}^d$ is the forward and backward output of Bi-LSTM at i-th step and $\mathbf{e}_i$ concatenates the hidden state to produce the tree-embedding for $\boldsymbol{x}_i$.

4.4. Training and Inference

As shown in figure 4, the input instance $x_j^{(i)} \in X_i$ and $y_k^{(i)} \in Y_i$ will be first transformed to embedding vector $\mathbf{x}_k^{(i)}$ and $\mathbf{y}_k^{(i)}$ using embedding look up matrix $\phi(\cdot)$. Then we search ancestor nodes of $x_k^{(i)}$ in hierarchical relation graph to transfer to embedding $\mathbf{e}_k^{(i)}$ and feed into encoder sequentially. The global context embedding will be produced by encoder as $\mathbf{g}$ using residual-supervised attention mechanism. Then the decoder will utilize $\mathbf{g}$ fusing with contextual information $\mathbf{c}$ to predict sequentially. The training algorithm is shown in detail in Alg. 1

Algorithm 1 Training algorithm

Input: Training dataset $\mathcal{D} = \{(X_1, Y_1), (X_2, Y_2), \cdots, (X_m, Y_m)\}$ and knowledge $(\mathcal{G}, \{\boldsymbol{c}_i, \boldsymbol{w}_i\}_i^N)$
Output: Optimal θ^*
Initialize: transform instances x_i, y_j to vector $\mathbf{x}_i = \phi(x_i)$ and $\mathbf{y}_i = \phi(y_i)$ Sample B samples from training set
for sample (X_i, Y_i) in B **do**
  for $\mathbf{x_i}$ in X_i **do**
    Use Eq.6 and Eq.7 to get $\mathbf{e}_i$
    Input $\mathbf{e}_i$ to encoder and Use Eq.2 to calculate $\mathbf{h}_i$
  end for
  Use Eq.4 to calculate $\mathbf{g}$
  for $\mathbf{y}_i$ in Y_i **do**
    Use Eq.2 to calculate $\mathbf{h}_i$
    Use Eq.5 to calculate output $\mathbf{o}_i$
    $\hat{y}_i \leftarrow$ argmax Softmax($\mathbf{o}_i$)
  end for
  Use Eq.8 to update parameter $\theta \leftarrow \theta - \Delta_\theta L(B; \theta)$
end for
return $\theta^* \leftarrow \theta$

The model is to find an optimal parameter θ by minimizing the cross entropy loss given training dataset $\mathcal{D}$ as follows:

$$\begin{aligned} L(\mathcal{D}; \theta) &= -\sum_{(X_i, Y_i,) \in \mathcal{D}} \log p(Y_i | X_i, \mathbf{c}_i, \mathbf{w}_i, \mathcal{G}; \theta) \\ &= -\sum_{(X_i, Y_i) \in \mathcal{D}} \sum_{j}^{l_i} \log p(y_j | y_1, y_2, \cdots, y_{j-1}, X_i, \mathbf{c}_i, \mathbf{w}_i, \mathcal{G}; \theta) \end{aligned} \tag{8}$$

For unseen bag $(X_i, \mathbf{c}_i, \mathbf{w}_i)$, the simple greedy prediction approach is utilized which will sequentially predict K number of labels $\{y_1, y_2, \cdots, y_k\}$ where K is a hyper-parameter.

5. Experiments

In this section, we demonstrate the effectiveness of our proposed model `KG-MIML-Net` on real-world dataset MIMIC-III. We compare `KG-MIML-Net` with several state-of-the-art methods

from multi-instance multi-label learning and heathcare area. Then we also studied the effect of different components proposed in our method where we use Attention, Tree-embedding, Context and ALL to denote the single residual-supervised-attention module, single tree-embedding module, single contextual layer module and the combination of the above three mentioned module embedded on basic encoder-decoder framework. At last, we make a case study with current state-of-the-art method in medicines prediction task. We make the source code of KG-MIML-Net publicly available at https://github.com/sjy1203/KG-MIML-Net.

5.1. Dataset

Table 2: Statistics of MIMIC-III Dataset.

MIMIC-III Dataset	Diseases (ICD-9)	Drugs (NDC)
# of patients	46520	
# of distinct codes	6986	4212
Avg # of sequence length	14	89
Max # of sequence length	540	2378

Source of data. MIMIC-III Johnson et al. (2016) is an openly available dataset developed by the MIT Lab for Computational Physiology, comprising deidentified health data associated with ∼40,000 critical care patients. We exploit the diagnosis data and medication data from table diagnoses_icd and prescriptions and further extract the lab test and demographic information. Basic statistics of dataset about our task can be found in table 2. Drug in MIMIC-III is in National Drug Code (NDC) version which is a unique 10-digit, 3-segment number. Diagnosis in MIMIC-III is in International Classification of Diseases 9 version (ICD-9). The ICD-9 ontology [1]and ATC ontology[2] can be used to generate the tree embedding.

Data processing. Every patient is associated with more than one diagnosis and drug, so the sequence within diagnoses and drugs is determined by diagnosis priority and drug taken time. The 3 segments of the NDC identify the labeler, the product, and the commercial package size without the ontology information. So we first transform the NDC code to ATC code and filter the least common diagnosis as done in Zhang et al. (2017) which result in 2000 different diagnoses and 306 different medicines. Then we choose the first-24 hour clinical measurements and medicines as labels. We assume the first-24 hour is the most important time for patient. The clinical measurements is a 101 dimension vector including lab tests (e.g. aniongap, bands), demographics (e.g. height, age and weight) and so on. Finally, the dataset has 43201 samples.

5.2. Comparison Methods

We compare the proposed method with the following methods:

- **K-most frequent**: The simple baseline choose K medicines for a disease which is the most common K medicines co-occur with that disease. K is the hyper parameter determined according to the performance in evaluation set.

1. https://github.com/clinicalml/embeddings/blob/master/eval/icd9Tree.txt
2. https://en.wikipedia.org/wiki/Anatomical_Therapeutic_Chemical_Classification_System

- **MLP**: The diseases and medicines are firstly transformed to multi-hot vector, the a 3-layer MLP is carried to make multi-label prediction. A global threshold is used to select positive medicines. The value of the threshold and hyper parameters are tuned on a validation set using grid search.
- **MIMLfast**: The traditional state-of-art MIML method transfered instances to label specific space and considered sub-concepts for each label. The representation of instances should be given first, so we use the skip-gram Mikolov et al. (2013) method to pre-generate the instances' representation. We utilize the code[3] in matlab version and keep the default setting to predict multi-label results.
- **Leap**: Leap models label instance mapping and label dependency by attention mechanism and RNN which is currently the state-of-art method.

5.3. Evaluation

We randomly divide the dataset into the training, validation and testing set in a 8:1:1 ratio. For deep learning models, the best size of embedding and hidden dimensions are 100 and 200, respectively. The Leap and our model are implemented using PyTorch (a deep learning framework with extensive support for accelerating training using GPUs), and all the methods are trained on Ubuntu 16.04 with 8GB memory and Nvidia 1080 GPU. We use Adam Kingma and Ba (2014) as the optimization algorithm for all deep learning models for 30 training epochs.

We measure the relative quality of model performances by using common multi-label metric Jaccard similarity score also called multi-label accuracy. The Jaccard similarity score is defined as the size of the intersection divided by the size of the union of ground truth label set and predicted label.

$$Jaccard\_similarity\_score = \frac{1}{|T|}\sum_{i}^{|T|}\frac{Y_i \cap \hat{Y}_i}{Y_i \cup \hat{Y}_i} \tag{9}$$

where $|T|$ is the number of samples in test set.

Table 3 shows the performance of the aforementioned metric on MIMIC-III dataset. The K-most frequent method is not effective because the co-occurrence matrix is easy to be influenced as the number of diagnoses and medicines associated with each patient is large due to high severity of patients at ICU.

Both MLP and MIMLfast lack the way to learn representation for instances and labels, when instances are represented as multi-hot vector, the method assumes equal contribution of all the instances where in our task different patients have different severity of diseases. The MIMLfast methods work poorly on the MIMIC-III dataset because it highly relies on the given representation of instances. The unsupervised learning method like Skip-gram Kiros et al. (2015) is not capable of learning effective representation for MIMLfast from complicated MIMIC-III dataset.

Leap outperforms the traditional methods which shows the RNN can be utilized to build high-order dependency among labels and incorporate attention mechanism to assign

3. http://lamda.nju.edu.cn/code_MIMLfast.ashx

different weight to instances . But the performance is constrained because it ignores the additional knowledge like the contextual information and ontology.

KG-MIML-Net consistently outperforms traditional methods by 8+% and Leap by 4+% with respect to Jaccard similarity score. The reason is that KG-MIML-Net effectively captures the label and instances dependency by RNN-encoder-decoder, more importantly it fully utilize the additional knowledge like contextual knowledge and structural knowledge. The compared methods among KG-MIML-Net shows 1+% and 3+% performance boost by single tree embedding and contextual layer module with respect to Leap. All proposed module can be combined to make better performance.

Figure 7 (a) illustrates the performance of Leap and KG-MIML-Net over 30 training epochs and (b) shows the performance of KG-MIML-Net with different module component over training epochs. We can see the same result with the performance in table 3 and an interesting observation is that the combination of proposed modules not only have better accuracy score but also more stable during training.

Table 3: MIMIC-III Drug Prediction test results.

Methods		Jaccard similarity score
K-most frequent		0.2659
MLP		0.2187
MIMLfast		0.1706
Leap		0.3026
KG-MIML-Net	Attention	0.3012
	Tree Embedding	0.3188
	Context	0.3385
	ALL	**0.3466**

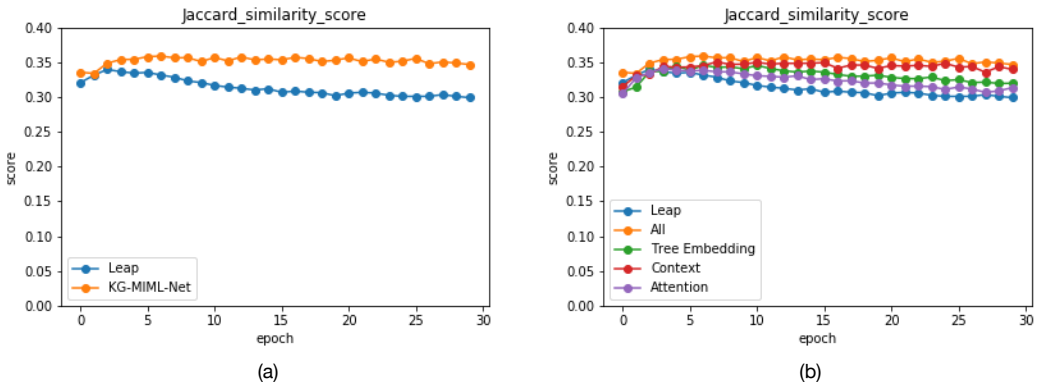


Figure 7: Jaccard similarity score in evaluation dataset along 30 epochs.

5.4. Case Study

In Table 4, we show an example of drugs prescribed for two patient given his context information and diagnoses. Diagnoses are encoded by ICD-9 starting with severer diagnosis

from left to right, and drugs encoded by ATC where bold denotes matching against ground-truth. They have the same diseases native coronary vessel encoded as '41401', hypertension encoded as '4019', type-2 diabetes encoded as '25000' and '2720' for Hypercholesterolemia.

The female patient in the right part have severer Type-2 diabetes than the male in the left part. The higher anion gap measurement for the female patient also show the more risk to have diabetes.

Leap method prescribes the same set drugs for the two patients with three wrongly prescribed drugs. In contrast to Leap, `KG-MIML-Net` prescribes most common set of drugs for the two patients and predicts four different drugs for them. The correct number of medicines is one more than the current state-art-of method in this random case.

Table 4: An Example of Recommended Medicines by Leap and `KG-MIML-Net` on MIMIC-III dataset.

	No.14156	No.17542
Personal Context	aninopgap: 9, albumin:3.19, band: 10.04, age: 67, gender: male, weight: 104, height: 180	aninopgap: 12, albumin:3.19, band: 10.04, age: 56, gender: female weight:88, height: 157
Diagnoses (ICD-9)	'41401', '4019', '2720', '25000'	'41401', '25000', '4019', '2720'
Predicted Drugs (Leap)	**'A02BA'**, **'N02BE'**, **'B05CX'**, **'A12CA'**, **'A12AA'**, **'A10AB'**, **'N07AA'**, **'A01AD'**, **'A06AA'**, **'C01CA'**, 'A07AA', 'M01AB', **'N01AX'**, **'A02BX'**, **'C01DA'**, **'A01AB'**, 'A03FA', **'N02AA'**, **'A06AD'**	
Predicted Drugs (`KG-MIML-Net`)	**'A02BA'**, **'N02BE'**, **'B05CX'**, **'N07AA'**,'M01AB', **'J01DB'**, **'C01DA'**, **'A03FA'**, **'A06AD'**, **'N01AX'**, **'A06AA'**, 'A12CA', **'A12AA'**, **'A10AB'**, **'C01CA'**, 'N02AA'	
	'A01AB', **'A02AA'**	**'A02BX'**, **'C01DA'**

6. Conclusion

In this paper, we propose `KG-MIML-Net`, an end to end learning model for medicines prediction that jointly models drug disease dependency. `KG-MIML-Net` formulate the medicines prediction problem in knowledge-guided multi-instance multi-label learning framework. Based on RNN encoder-decoder framework, residual-supervised attention, contextual layer and tree-embedding module is embedded to overcome complex dependencies and data skewness problem by incorporating the contextual and structural knowledge. Throughout experiments, we successfully demonstrated the performance of `KG-MIML-Net` by 4+% over all baselines.

Future works on the one hand will focus on extending our model by adding more knowledge like drug-drug interactions (Ma et al., 2018; Xiao et al., 2018b) to make safe medicines prediction, on the other hand, it may be an valuable idea to combine longitudinal patient history to further improve prediction performance.

Acknowledgment

This work was supported by Peking University Medicine Seed Fund for Interdisciplinary Research. We also thank NVIDIA for the support of a GPU.

References

Edward Choi, Mohammad Taha Bahadori, Le Song, Walter F Stewart, and Jimeng Sun. GRAM: Graph-based Attention Model for Healthcare Representation Learning. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 787–795. ACM, 2017.

Jian Dai, Meihui Zhang, Gang Chen, Ju Fan, Kee Yuan Ngiam, and Beng Chin Ooi. Fine-grained Concept Linking using Neural Networks in Healthcare. In *Proceedings of the 2018 International Conference on Management of Data*, pages 51–66. ACM, 2018.

Ji Feng and Zhi-Hua Zhou. Deep MIML Network. In *AAAI*, pages 1884–1890, 2017.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

Sheng-Jun Huang and Zhi-Hua Zhou. Fast Multi-instance Multi-label Learning. *AAAI*, 2014.

Alistair EW Johnson, Tom J Pollard, Lu Shen, H Lehman Li-wei, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. MIMIC-III, A Freely Accessible Critical Care Database. *Scientific Data*, 3:160035, 2016.

Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980, 2014.

Ryan Kiros, Yukun Zhu, Ruslan R Salakhutdinov, Richard Zemel, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Skip-thought Vectors. In *Advances in Neural Information Processing Systems*, pages 3294–3302, 2015.

Xingyue Li, Shouhong Wan, Chang Zou, and Bangjie Yin. Multi-instance Multi-label Learning for Image Categorization Based on Integrated Contextual Information. In *International Conference on Image and Graphics*, pages 639–650. Springer, 2017.

Tengfei Ma, Cao Xiao, Jiayu Zhou, and Fei Wang. Drug Similarity Integration Through Attentive Multi-view Graph Auto-Encoders. In *IJCAI*, 2018.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. *arXiv preprint arXiv:1301.3781*, 2013.

Mihai Surdeanu, Julie Tibshirani, Ramesh Nallapati, and Christopher D Manning. Multi-instance Multi-label Learning for Relation Extraction. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 455–465. Association for Computational Linguistics, 2012.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to Sequence Learning with Neural Networks. In *Advances in Neural Information Processing Systems*, pages 3104–3112, 2014.

Jian-Sheng Wu, Sheng-Jun Huang, and Zhi-Hua Zhou. Genome-wide Protein Function Prediction through Multi-instance Multi-label Learning. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 11(5):891–902, 2014.

Cao Xiao, Edward Choi, and Jimeng Sun. Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review. *Journal of the American Medical Informatics Association*, 2018a.

Cao Xiao, Ying Li, Inci M Baytas, Jiayu Zhou, and Fei Wang. An MCEM Framework for Drug Safety Signal Detection and Combination from Heterogeneous Real World Evidence. *Scientific reports*, 8(1):1806, 2018b.

Xin-Shun Xu, Xiangyang Xue, and Zhi-Hua Zhou. Ensemble Multi-instance Multi-label Learning Approach for Video Annotation Task. In *Proceedings of the 19th ACM International Conference on Multimedia*, pages 1153–1156. ACM, 2011.

Hao Yang, Joey Tianyi Zhou, Jianfei Cai, and Yew Soon Ong. MIML-FCN+: Multi-instance Multi-label Learning via Fully Convolutional Networks with Privileged Information. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5996–6004. IEEE, 2017.

Zheng-Jun Zha, Xian-Sheng Hua, Tao Mei, Jingdong Wang, Guo-Jun Qi, and Zengfu Wang. Joint Multi-label Multi-instance Learning for Image Classification. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

Min-Ling Zhang and Zhi-Hua Zhou. M3MIML: A Maximum Margin Method for Multi-instance Multi-label Learning. In *2008 Eighth IEEE International Conference on Data Mining*, pages 688–697. IEEE, 2008.

Yutao Zhang, Robert Chen, Jie Tang, Walter F Stewart, and Jimeng Sun. Leap: Learning to Prescribe Effective and Safe Treatment Combinations for Multimorbidity. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1315–1324. ACM, 2017.

Zhi-Hua Zhou and Min-Ling Zhang. Multi-instance Multi-label Learning with Application to Scene Classification. In *Advances in Neural Information Processing Systems*, pages 1609–1616, 2007.

Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang, and Yu-Feng Li. MIML: A Framework for Learning with Ambiguous Objects. *CORR abs/0808.3231*, 112, 2008.

Zhi-Hua Zhou, Yu-Yin Sun, and Yu-Feng Li. Multi-instance Learning by Treating Instances as Non-iid Samples. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pages 1249–1256. ACM, 2009.

Zhi-Hua Zhou, Min-Ling Zhang, Sheng-Jun Huang, and Yu-Feng Li. Multi-instance Multi-label Learning. *Artificial Intelligence*, 176(1):2291–2320, 2012.