Automatic Emergency Diagnosis with Knowledge-Based Tree Decoding

Ke Wang[1,2] , **Xuyan Chen**[3] , **Ning Chen**[1,2] and **Ting Chen**[1,2*]
[1]Institute for Artificial Intelligence, Department of Computer Science and Technology, Tsinghua University, Beijing 100084, China
[2]Tsinghua-Fuzhou Institute of Digital Technology, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China
[3]Beijing Tsinghua Changgung Hospital, School of Clinical Medicine, Tsinghua University, Beijing, China
wangke18@mails.tsinghua.edu.cn, tingchen@tsinghua.edu.cn,

Abstract

Automatic diagnosis based on clinical notes is critical especially in the emergency department, where a fast and professional result is vital in assuring proper and timely treatment. Previous works formalize this task as plain text classification and fail to utilize the medically significant tree structure of International Classification of Diseases (ICD) coding system. Besides, external medical knowledge is rarely used before, and we explore it by extracting relevant materials from Wikipedia or Baidupedia. In this paper, we propose a knowledge-based tree decoding model (K-BTD), and the inference procedure is a top-down decoding process from the root node to leaf nodes. The stepwise inference procedure enables the model to give support for decision at each step, which visualizes the diagnosis procedure and adds to the interpretability of final predictions. Experiments on real-world data from the emergency department of a large-scale hospital indicate that the proposed model outperforms all baselines in both micro-F1 and macro-F1, and reduce the semantic distance dramatically.

1 Introduction

The clinical note, an essential part of Electronic Health Record (EHR), generally contains a patient's past medical history, chief complaints and current symptoms. The physicians need to study and be on probation for years before they can give diagnosis individually, but the diagnosis is still time-consuming and error-prone.

To address existing drawbacks of human diagnosis, researchers started to study automatic diagnosis [Xiao *et al.*, 2018]. Automatic diagnosis takes raw texts of clinical notes as input, and gives the codes of diseases according to the ICD coding system [WHO, 1978], which is adopted in hospitals world-wide. The ICD codes are naturally organized as a tree structure. The tree starts from a virtual root node, goes deeper through intermediate nodes and finally reaches diseases in leaf nodes. Only leaf nodes correspond to a specific disease with its ICD code. Intermediate nodes represent a medical concept or a range of diseases.

Automatic diagnosis has become a popular research field recently [Perotte *et al.*, 2013; Wang *et al.*, 2016; Subotin and Davis, 2016], and the majority of existing works formalize it as a plain text classification task. For example, [Lipton *et al.*, 2016] and [Li *et al.*, 2018] use Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN) to predict diseases. To give support from clinical notes for final predictions, attention mechanisms are used in [Sha and Wang, 2017] and [Mullenbach *et al.*, 2018].

Although existing models have made progress on the accuracy of disease diagnosis by a considerable margin, automatic diagnosis is still confronted with two major problems.

Negligence of the Medical Relationships among Diseases. Existing models generally treat diseases as mutually independent. However, from a medical perspective, diseases are interconnected and they are organized hierarchically in the ICD tree. The constraints of hierarchical structure can prevent the predictions from being too far away from the ground truth. For example, without hierarchical restrictions, a patient with acute myocardial infarction (disease of the circulatory system) can be diagnosed with acute gastroenteritis (disease of the digestive system) by mistake, and this may bring about life loss and huge economic compensation.

Low Practicality of Support. Existing attention-based models can already tag words that have deep impact on final predictions. However, this one-step attention cannot reveal a transparent reasoning process, thus lacking in the practicality of assisting doctors in making decisions. In general, diagnosis is a step-by-step procedure, where the physician first locates the diseased organ and then uses the knowledge he has learned and the information from patients to stepwisely reach final results. Each step inside the diagnosis procedure requires different support from clinical notes.

To address these problems, we propose a knowledge-based tree decoding model named K-BTD, consisting of Clinical Notes Encoder, Knowledge Encoder, Judge Net and Fusion Net. K-BTD takes in raw clinical notes as input, and the inference procedure is a top-down decoding process of the ICD tree. Each node inside the tree is equipped with external med-

[*]Contact Author

ical knowledge extracted from Wikipedia or Baidupedia. At each decoding step, the Judge Net decides whether to expand the children of the current node, and the Fusion Net aggregates information from multiple resources to promote further decoding. The whole process is repeated recursively until there are no more children to expand. The stepwise decoding process imitates the diagnosis procedure of a human doctor, and at each step our model can give support for its decision, which visualizes the reasoning process and provides human doctors with better references.

We conduct experiments on real-world data from the emergency department of Beijing Tsinghua Changgung Hospital, a large-scale hospital in China. Experimental results indicate that our model achieves significant improvements over other state-of-the-art models in micro-F1, macro-F1 and semantic distance. We also show the superiority of our model in terms of interpretability. Ablation analysis and error analysis are conducted to verify the internal mechanism. To the best of our knowledge, this is the first empirical study to inference diagnosis with knowledge-based tree decoding.

2 Related Work

2.1 Automatic Diagnosis

Automatic Diagnosis is a long-standing task in the field of medical informatics. Early works utilize machine learning models such as hierarchical Support Vector Machine (SVM) [Perotte *et al.*, 2013]. With the rapid development of deep learning technologies, researchers start to formalize it as a text classification task. Long Short Term Memory (LSTM) [Lipton *et al.*, 2016] and CNN [Li *et al.*, 2018] are used to extract semantic features from textual content. Bag-of-words and disease correlation graph are explored in [Wang *et al.*, 2016]. However, these methods can only extract shallow features and cannot give support for final predictions, which block it from practical application.

With the widespread of attention mechanism, researchers begin to predict diseases and their support by incorporating deep learning models with it. [Mullenbach *et al.*, 2018] adopts per-label attention mechanism and allow the model to learn distinct representations for each disease label.

2.2 Tree-based Multi-label Classification

Tree-based multi-label classification is a branch of multi-label classification, and it is applicable when the predictor has a hierarchical structure. To be specific, the label space is a tree where nodes represent nested semantic concepts, and the specificity of them increases with depth. Its successful implementation can reduce a large discrete sample space to only a small number of candidate labels.

Some researchers focus on inducing tree structure label space to improve inference efficiency [Beygelzimer *et al.*, 2009; Daumé III *et al.*, 2017]. Some researchers employ the already existed label space structure. For example, the natural tree structure of Medical Subject Heading (MeSH) is utilized by [Singh *et al.*, 2018] in MeSH tagging task.

Some researchers have explored the tree structure of ICD coding system. [Perotte *et al.*, 2013] adopts hierarchical SVM to model the inclusion and exclusion relationships of diseases. [Kamkar *et al.*, 2015] reaches stable and better feature selection based on the ICD tree structure. [Xie and Xing, 2018] applies tree-of-sequences LSTM to model the latent representation of each node in the ICD tree with textual descriptions of the ICD codes and their hierarchical structure. However, none of them formalizes automatic diagnosis as a tree-decoding procedure along the ICD tree.

3 Model

3.1 Problem Formulation

Considering that a patient can be diagnosed with more than one disease, we treat automatic diagnosis as a multi-label classification task over ICD-9 codes. ICD-9 is a standard version of the ICD coding system, which contains over 15,000 codes in its taxonomy[1]. In our study, we only consider high-frequent ICD-9 codes such as top-100 and top-150 codes, which takes up more than 90% of all appeared ICD-9 codes. Each node in the ICD-9 tree is equipped with a piece of text representing the external knowledge extracted from Wikipedia or Baidupedia.

The input of our model, the clinical notes of a patient, is a word sequence $X = \{x_1, x_2, \dots, x_N\}$, where N is the length of sequence X. Let the ICD-9 codes for diseases to be the label space $\mathcal{L}$, and the labeling task is to determine $y_l \in \{0, 1\}$ for all $l \in \mathcal{L}$.

3.2 Model Overview

The clinical note from patient is first encoded by **Clinical Notes Encoder** to get feature sequence $T = \{t_1, t_2, \dots, t_N\}$ and flowing vector F_{root} for root node. We name F_n as flowing vector for node n because it contains all information flowing from the root node to current node n. Next, the tree decoding process starts from the root node of the ICD-9 tree.

Each step of the decoding process will be conducted on a particular node n in the tree (e.g., root). The **Knowledge Encoder** first encodes the external knowledge of each child of the current node n. Next, the **Judge Net** enumerates each child node and decides whether to expand this child node for further decoding. If a child node m is chosen to be expanded, the **Fusion Net** will aggregate flowing vector F_n and the external knowledge of node m to get flowing vector F_m for child node m. The decoding process will repeat recursively until there are no more nodes to expand. The expanded leaf nodes will be chosen as final predictions. The overall framework is shown in Figure 1.

3.3 Clinical Notes Encoder

Taking the word sequence X of clinical notes as input, the Clinical Notes Encoder computes the latent text description through two layers, i.e., embedding layer and RNN layer.

We first convert each word x_i to a vector with pretrained Tencent AI Lab Embedding Corpus [Song *et al.*, 2018], which features its strength in the medical domain. We have tried building word embeddings with the dataset we adopt,

[1] The ICD-9 tree structure can be found at https://bioportal.bioontology.org/ontologies/ICD9CM.

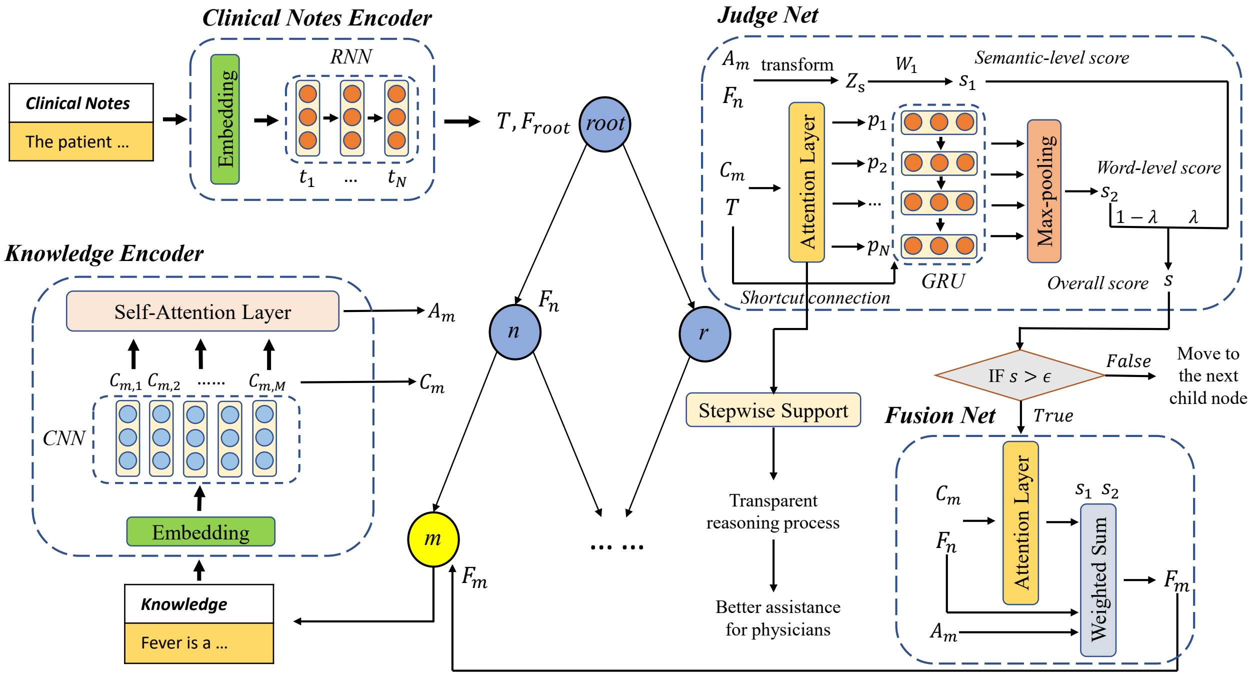

Figure 1: Overview of the proposed K-BTD model. The decoding process is now conducted on node n, and deciding whether to expand child node m for further decoding. The A_m, C_m, T on the right side are originally calculated from the left side.

and the performance is not as good. After the embedding layer, an RNN is used to extract the contextual information from clinical notes. It is noticeable that the RNN can be in any form such as Gated Recurrent Unit (GRU), LSTM and BERT [Devlin *et al.*, 2018]. After the RNN layer, we can get clinical notes feature sequence $T = \{t_1, t_2, \ldots, t_N\} \in \mathbb{R}^{N \times q}$, where q is the dimension of the hidden state. The flowing vector for the root node is set as $F_{root} = t_N$.

3.4 Knowledge Encoder

For Knowledge Encoder, we choose CNN due to its high efficiency. The input of Knowledge Encoder from node m is a word sequence of external medical knowledge $G_m = \{g_{m,1}, g_{m,2}, \ldots, g_{m,M}\}$, where M is the sequence length. The same embedding method is applied on the sequence to get $\tilde{G}_m = \{\tilde{g}_{m,1}, \tilde{g}_{m,2}, \ldots, \tilde{g}_{m,M}\}$.

The convolution operation is done with a convolution matrix $W \in \mathbb{R}^{u \times (h \times k)}$, where u is the number of filters, h is the length of the sliding window and k is the dimension of word embeddings. The convolutional feature matrix $C_m \in \mathbb{R}^{M \times u}$ is calculated by:

$$C_{m,i} = W \cdot \tilde{g}_{m,i:i+h-1} + b \tag{1}$$

where $\tilde{g}_{m,i:i+h-1}$ is the concatenation of word embeddings within the i-th sliding window, and $b \in \mathbb{R}^u$ is a bias vector.

To integrate the information from convolutional feature matrix C_m, we adopt self attention and calculate A_m, the feature vector of external knowledge for node m as follows:

$$A_m = \sum_i \left(\frac{e^{C_{m,i} W_a}}{\sum_j e^{C_{m,j} W_a}} \right) C_{m,i} \tag{2}$$

where W_a is a multi-layer perceptron.

3.5 Judge Net

Suppose the decoding process is now conducted on node n, and the network is deciding whether to expand child node m for further decoding. The Judge Net calculates the possibility that child node m will be expanded with two measurements, namely semantic-level score and word-level score.

Semantic-level score. We use semantic-level score s_1 to represent the semantic-level interactions between external knowledge feature vector A_m and the aggregated information F_n that has flown to current node n:

$$\begin{aligned} s_1 &= \text{Sigmoid}\left(Z_s W_1\right) \in \mathbb{R} \\ \text{where } Z_s &= \text{Concat}\left(F_n, A_m, F_n \circ A_m, |F_n - A_m|\right) \end{aligned} \tag{3}$$

where W_1 is a multi-layer perceptron and $\circ$ indicates element-wise multiplication.

Word-level score. Semantic-level score only considers the interactions between holistic feature vectors. Here, we use word-level score s_2 to introduce reactions at the word-level between clinical notes feature sequence $T = \{t_1, t_2, \ldots, t_N\}$ and convolutional feature sequence $\{C_{m,i}\}$.

First, we calculate the similarity matrix S with $\{t_i\}$ and $\{C_{m,j}\}$ as follows:

$$S_{ij} = \tanh\left(t_i W_2 C_{m,j}^T\right) \tag{4}$$

where $W_2 \in \mathbb{R}^{q\times u}$ is a parameter matrix. Then, row-wise and column-wise softmax are separately applied on S to get γ_{ij} and δ_{ij}:

$$\gamma_{ij} = \frac{exp(S_{ij})}{\sum_{v=1}^{M} exp(S_{iv})} \tag{5}$$

$$\delta_{ij} = \frac{exp(S_{ij})}{\sum_{v=1}^{N} exp(S_{vj})} \tag{6}$$

we use $\xi_i = \sum_{j=1}^{M} \delta_{ij}$ to measure the importance of the i^{th} word in clinical notes. It is noticeable that ξ_i is only calculated for interpretability, and is not involved in subsequent calculations.

Next, we generate intermediate representation $p_i \in \mathbb{R}^u$ by attentively aggregate $\{C_{m,j}\}$:

$$p_i = \sum_{j=1}^{M} \gamma_{ij} C_{m,j} \tag{7}$$

Then, we apply a GRU to process the generated intermediate representation $P = \{p_1, p_2, \ldots, p_N\} \in \mathbb{R}^{N\times u}$:

$$\overline{P} = \text{GRU}(P) \in \mathbb{R}^{N\times d_p} \tag{8}$$

where d_p is the dimension of the hidden state. Next, we project $\overline{P}$ to a compressed vector by max-pooling along the column axis to get the word-level score s_2:

$$s_2 = \text{Sigmoid}\left(\text{Maxpool}_{\text{col}}\left(\text{Concat}\left(\overline{P}, T\right)\right) W_3\right) \in \mathbb{R} \tag{9}$$

where W_3 is a multi-layer perceptron, and T is a shortcut connection to facilitate the training process.

Overall score. To consider both scores simultaneously, we formulate the overall score s as a weighted summation of semantic-level score and word-level score:

$$s = \lambda s_1 + (1-\lambda)s_2 \tag{10}$$

where $\lambda \in [0,1]$ is a hyper-parameter. If s is greater than a preset threshold ϵ, the child node m will be expanded. If s is smaller than ϵ, the Judge Net will check the next child node of current node n. ϵ is set to be 0.45 in our experiment to balance precision and recall.

# of admission records	72333
# of unique ICD-9 codes	2156
Ratio of top-100 codes	91.6%
Ratio of top-150 codes	95.2%
Avg. # of codes per admission (top-100)	2.35
Avg. # of codes per admission (top-150)	2.43
Avg. # of words per admission	105.58
Avg. # of words per external knowledge	268.17

Table 1: Statistics of the dataset from Beijing Tsinghua Changgung Hospital, a large-scale hospital in China.

3.6 Fusion Net

The Fusion Net comes into operation only if a child node m is chosen to be expanded by Judge Net. Its purpose is to generate F_m by fusing the information flowing to parent node n and the external knowledge of child node m.

First, the weight β_i for each position i in the convolutional feature sequence $\{C_{m,i}\}$ is calculated via dot-production with flowing vector F_n and subsequent softmax operation:

$$\begin{aligned} \alpha_i &= F_n \odot C_{m,i} \\ \beta_i &= \frac{exp(\alpha_i)}{\sum_{j=1}^{M} exp(\alpha_j)} \end{aligned} \tag{11}$$

Next, the flowing vector of child node m is the weighted summation of F_n, A_m and the weighted average of $\{C_{m,i}\}$:

$$F_m = F_n + \frac{s_2}{s_1+s_2} A_m + \frac{s_1}{s_1+s_2} \sum_{i=1}^{M} \beta_i C_{m,i} \tag{12}$$

The different weight for A_m and $\sum_{i=1}^{M} \beta_i C_{m,i}$ is designed to introduce more information from child node m. If s_1 is smaller than s_2, the model needs to pay more attention to the word-level information. If s_1 is greater than s_2, the model should absorb more information from the semantic-level.

3.7 Training

We use layerwise multi-label cross entropy to train our network. The loss function is as follows:

$$loss = \sum_{d=1}^{MaxDepth} \left[e^{-d} \sum_{k=1}^{|y_d|} L(y_{d,k}, p_{d,k}) \right] \tag{13}$$

where $MaxDepth$ is the depth of the ICD-9 tree, L is binary cross entropy loss, p_d is the prediction of Judge Net at layer d and y_d is the ground truth label at layer d. The exponential part is added to penalize more for early error and penalize less for late error.

4 Experiments

4.1 Dataset Description

The data from the emergency department of Beijing Tsinghua Changgung Hospital, a large-scale hospital in China, is collected from 2015 to 2017, and it is the first large automatic diagnosis dataset in Chinese. The dataset contains unstructured

Chronic Airway Obstruction (ICD-9 code: 496.0)
Clinical manifestations:
Chronic cough is often the earliest symptom, and it can be unhealed with the course of the disease. Cough ...
Cause of diseases:
The risk factors that have been found can be roughly divided into external factors like environment ...

Table 2: Partially shown example of external medical knowledge from Baidupedia (translated from Chinese).

	Tasks	Top-100 Codes			Top-150 Codes		
	Metrics	micro-F1	macro-F1	SD	micro-F1	macro-F1	SD
Type #1	TFIDF+SVM	0.6241±0.0089	0.5027±0.0076	1.818±0.047	0.5987±0.0077	0.4592±0.0082	2.371±0.061
	TextCNN	0.6695±0.0126	0.5239±0.0097	1.315±0.041	0.6409±0.0117	0.4887±0.0109	1.826±0.057
	BERT	0.6823±0.0118	0.5459±0.0104	1.250±0.051	0.6598±0.0097	0.5201±0.0129	1.588±0.072
	DeepLabeler	0.6703±0.0152	0.5357±0.0193	1.291±0.038	0.6385±0.0137	0.4977±0.0201	1.799±0.089
	CAML	0.6618±0.0134	0.5375±0.0093	1.282±0.035	0.6402±0.0161	0.5028±0.0144	1.766±0.093
	AIC	0.6875±0.0093	0.5602±0.0081	1.142±0.045	0.6557±0.0096	0.5152±0.0108	1.527±0.059
Type #2	C-MemNNs	0.6855±0.0177	0.5652±0.0203	1.131±0.058	0.6647±0.0135	0.5227±0.0121	1.476±0.033
	Fact-Law	0.6785±0.0147	0.5603±0.0111	1.189±0.040	0.6602±0.0181	0.5185±0.0166	1.481±0.049
Ours	K-BTD (LSTM)	0.7025±0.0104	0.5903±0.0138	0.899±0.014	0.6798±0.0117	0.5462±0.0142	1.182±0.047
	K-BTD (GRU)	0.7011±0.0149	0.5942±0.0126	0.920±0.021	0.6813±0.0206	**0.5493**±0.0178	1.169±0.051
	K-BTD (BERT)	**0.7085**±0.0128	**0.5973**±0.0097	**0.852**±0.036	**0.6855**±0.0175	0.5472±0.0134	**1.159**±0.032

Table 3: Automatic diagnosis results on the data from the emergency department of Beijing Tsinghua Changgung Hospital. Values after the plus minus sign denote standard deviations from 5-fold random data splits.

clinical notes including chief complaints, history of recent illness, past medical history and structured data such as auxiliary examination results.To keep track with previous work, we only utilize the unstructured part.

Each admission record is tagged with one or more ICD-9 codes by licensed physicians, denoting the identified diseases. A summary of the dataset statistics is provided in Table 1. We choose the top-100 and top-150 most frequent codes to conduct two separate experiments. When doing experiments on top-k frequent codes, we filter the dataset down to instances that have at least one of the top-k frequent codes. In experiment, we conduct random five fold cross-validation to examine the performance of our model as well as baselines.

For external medical knowledge, we use Wikipedia and Baidupedia as resources. [Trevena, 2011] has demonstrated the reliability of medical articles in Wikipedia, and medical terms in Baidupedia are under the supervision of National Health Care Commission of China. The extraction of external medical knowledge is divided into three steps. For each node in the ICD-9 tree, we first search the corresponding Wikipedia page. If the Wikipedia page does not exist, we use the corresponding Baidupedia page to make up for it. If the corresponding Baidupedia page also does not exist, we will consider highly related candidates in Baidupedia. We rank related candidates by the similarity between candidates and query terms and select the one with highest similarity. For Wikipedia pages, we use the content in the section of *Signs and Symptoms*. For Baidupedia pages, we extract the materials in sections of *Clinical Manifestations* and *Cause of Diseases*. An illustration of external medical knowledge is shown in Table 2.

4.2 Baselines

For comparison, we reproduce two major categories of baselines. The first category utilizes only clinical notes and the second employs external medical knowledge as well.

As for the first category baselines, we first choose **TFIDF+SVM**, **TextCNN** [Kim, 2014] and **BERT** [Devlin *et al.*, 2018], which are classic models for text classification. Besides, three classic automatic diagnosis models **DeepLabeler** [Li *et al.*, 2018], **AIC** [Xie and Xing, 2018] and **CAML** [Mullenbach *et al.*, 2018] are also adopted for comparison.

As for the second category baselines, we adopt **C-MemNNs** [Prakash *et al.*, 2017], which uses multi-hop memory networks to inference diagnosis. As there are few works in automatic diagnosis that utilize external knowledge, we adopt a classic reading comprehension model **Fact-Law Attention Model** [Luo *et al.*, 2017] from the legal judgement domain for comprehensive comparison. All baselines in type #2, as well as the proposed model use the same external knowledge to assure fair comparison.

To better demonstrate the advantage of our model, we use LSTM, GRU and BERT to replace the RNN in Clinical Notes Encoder and conduct three different experiments. The code for our model is publicly available at https://github.com/kaisadadi/K-BTD.

4.3 Evaluation Metrics

The distribution of diseases is highly imbalanced so we evaluate our model with both micro-F1 and macro-F1. Besides, we propose another metrics named semantic distance (SD), which is modified from [Singh *et al.*, 2018]. SD is defined as:

$$SD = \frac{1}{|\mathcal{Y}|}\sum_{u\in\mathcal{Y}}\min_{v\in\hat{\mathcal{Y}}} D(u,v) + \frac{1}{|\hat{\mathcal{Y}}|}\sum_{u\in\hat{\mathcal{Y}}}\min_{v\in\mathcal{Y}} D(u,v) \qquad (14)$$

where $\mathcal{Y}$ is the set of target codes and $\hat{\mathcal{Y}}$ is the set of predicted codes. D is a distance function that measures the shortest distance between two nodes in the ICD-9 tree. If $\hat{\mathcal{Y}}$ is empty, SD is set to be the maximum distance between two nodes in the ICD-9 tree. The semantic distance measures the distance between predictions and labels from a medical view. Mismatch with small semantic distance is tolerable, while a large semantic distance might result in completely wrong operations or even death.

4.4 Experimental Settings

In experiment, we set λ to 0.5, ϵ to 0.45 and d_p to 256. We adopt Adam [Kingma and Ba, 2015] for optimization. The size of the mini-batch is 64, and the learning rate is 10^{-5} for BERT-related models and 10^{-3} otherwise. Dropout is set to 0.5, and weight decay is 10^{-5}.

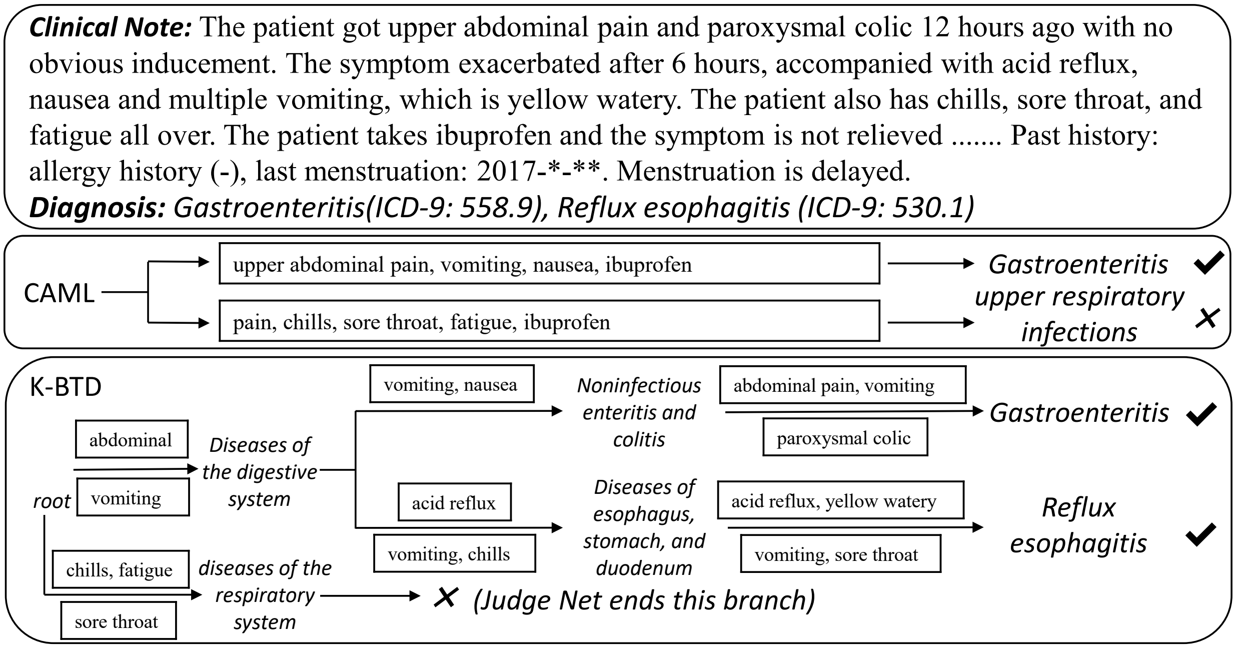
Clinical Note: The patient got upper abdominal pain and paroxysmal colic 12 hours ago with no obvious inducement. The symptom exacerbated after 6 hours, accompanied with acid reflux, nausea and multiple vomiting, which is yellow watery. The patient also has chills, sore throat, and fatigue all over. The patient takes ibuprofen and the symptom is not relieved .... Past history: allergy history (-), last menstruation: 2017-*-**. Menstruation is delayed.
Diagnosis: Gastroenteritis(ICD-9: 558.9), Reflux esophagitis (ICD-9: 530.1)
CAML
upper abdominal pain, vomiting, nausea, ibuprofen
Gastroenteritis ✓
pain, chills, sore throat, fatigue, ibuprofen
upper respiratory infections ✗
K-BTD
abdominal
vomiting
root
Diseases of the digestive system
vomiting, nausea
Noninfectious enteritis and colitis
abdominal pain, vomiting
paroxysmal colic
Gastroenteritis ✓
acid reflux
vomiting, chills
Diseases of esophagus, stomach, and duodenum
acid reflux, yellow watery
vomiting, sore throat
Reflux esophagitis ✓
chills, fatigue
sore throat
diseases of the respiratory system
✗ (Judge Net ends this branch)

Figure 2: Evaluation of interpretability with a partially shown clinical note (translated from Chinese). Support is placed in the rectangular bar. The middle part shows the support and predictions from CAML, and the bottom part shows the stepwise inference results and support from our proposed model.

4.5 Experimental Results

We evaluate our model on real-world data from the emergency department of Beijing Tsinghua Changgung Hospital, and experimental results are shown in Table 3. From the results we can see that:

(1) K-BTD exceeds all baselines in both experiments, which demonstrates the effectiveness and robustness of our model.

(2) K-BTD surpasses baselines in macro-F1 by a considerable margin, and this indicates that the utilization of tree structure enables the model to make the right decisions on diseases that are not common to appear.

(3) K-BTD reduces semantic distance significantly compared with baselines. This indicates that the results of our model are closer to the ground truth than baselines and can provide better references for human doctors.

(4) K-BTD can benefit from the research progress in text encoder. K-BTD is actually a framework, where the components inside can be updated easily. We are confident that our model can reach higher performance with better text encoders in the future.

4.6 Evaluation of Interpretability

Because our model employs tree decoding architecture, it can give support for decision at each step while conventional models can only provide support for final predictions. Here we demonstrate the interpretability of our model with an actual clinical note from the dataset, and compare it with another explainable model CAML [Mullenbach *et al.*, 2018].

For K-BTD, each step's support is chosen from words with value ξ_i greater than a preset threshold. For CAML, we follow the original settings. The results are shown in Figure 2.

We can see from results that support from both CAML and our model catch similar important words such as "vomiting" and "abdominal pain". However, without stepwise inference, CAML misdiagnoses *upper respiratory infections* because of the shared symptoms between diseases (e.g. sore throat). Besides, we can see that our model's decision at each step has different support. For example, the support for *diseases of the digestive system* are "abdominal" and "vomiting", and the support for *diseases of esophagus, stomach and duodenum* are "acid reflux", "vomiting" and "chills". This observation demonstrates that our model pays attention to different parts of the clinical notes at different positions in the decoding process. The stepwise support brings more interpretability to final predictions and can provide human doctors with better assistance in diagnosis inference.

4.7 Ablation Analysis

External medical knowledge is a critical part of our model. To further explore the role that external medical knowledge plays in the whole architecture, we conduct three experiments:

Experiment #1. Randomly mask from 10% to 50% of the knowledge texts for all nodes in the ICD-9 tree.

Experiment #2. Randomly shuffle the knowledge for 10% to 50% nodes in the ICD-9 tree.

Experiment #3. Randomly replace the knowledge for 10% to 50% nodes in the ICD-9 tree with totally irrelevant materials.

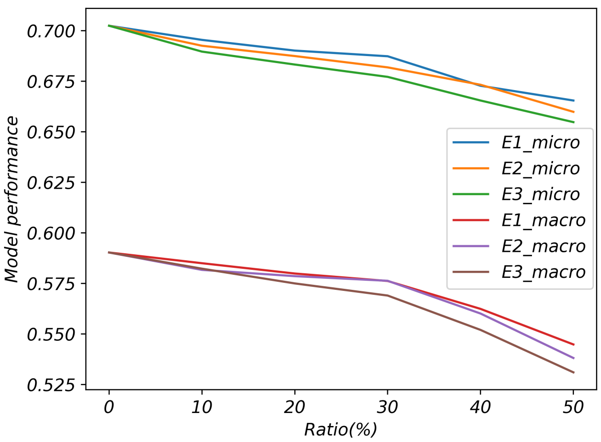

Figure 3: Results of ablation analysis. E1, E2 and E3 correspond to experiment #1, #2 and #3. Micro means micro-F1 grade and macro means macro-F1 grade.

Model	Depth of the ICD-9 tree				
	0	1	2	3	4
K-BTD	0%	**13%**	**23%**	53%	**11%**
BERT	0%	16%	29%	**42%**	13%
AIC	0%	15%	27%	45%	13%
C-MemNNs	0%	14%	31%	43%	12%

Table 4: Results of error analysis.

We repeat each experiment five times to reduce the effect of randomness, and use the mean value as final results. For all three experiments, the model takes more epochs to converge and the performance on both micro-F1 and macro-F1 of the initial three to five epochs decreases a lot. The results are shown in Figure 3.

We observe that the model performance decreases on all three experiments, and the degree of decline increases from experiment #1 to experiment #3. It's clear that noise to external medical knowledge can decrease model performance. The severer the noise is, the more the model performance drops, which indicates that our model relies heavily on the quality of external medical knowledge. We can safely predict that external knowledge with better quality can bring further improvement to model performance.

4.8 Error Analysis

To give better insight into out model's performance, we calculate the ratio of first-occurred errors that appear at different depth in the ICD-9 tree. We define the depth of the root node to be 0, and the maximum depth is 4. The results are shown in Table 4. For baseline models without tree decoding, we use the ICD-9 tree to locate the depth of the first-occurred error.

It's clear that K-BTD make mistakes in deeper positions in the ICD-9 tree compared with baselines. The deeper the first error occurs in the decoding process, the closer the distance between predictions and ground truth is, which again proves the effectiveness of our model. The reason of low error ratio in depth 4 is that the number of children of nodes in depth 3 is quite small.

5 Conclusion

In this paper, we propose K-BTD, a knowledge-based tree decoding model for automatic emergency diagnosis. To be specific, we utilize the medically significant ICD-9 tree structure and formulate this task as a stepwise top-down decoding procedure. External knowledge is extracted to assist the decoding process, and we have demonstrated its importance in our model. The top-down decoding process enables our model to give different support for each decision, which adds to the interpretability and practicality of our model compared with existing single-step models. Experimental results on real-world data show that our model outperforms baselines in all metrics, which proves the effectiveness and robustness of our model.

In the future, we will focus on increasing the diagnosis accuracy of infrequent diseases.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (grants 61872218, 61721003, 61673241 and 61906105), National Key R&D Program of China (2019YFB1404804), Tsinghua-Fuzhou Institute of Digital Technology, Beijing National Research Center for Information Science and Technology (BNRist), Tsinghua University-Peking Union Medical College Hospital Initiative Scientific Research Program, and Tsinghua University Big Data Research Center. The funders had no roles in study design, data collection and analysis, the decision to publish, and preparation of the manuscript. Besides, the author would like to thank Haoxi Zhong, Yudong Chen and Zhijing Wu for their sincere and helpful suggestions on paper writing.

References

[Beygelzimer *et al.*, 2009] Alina Beygelzimer, John Langford, Yuri Lifshits, Gregory Sorkin, and Alex Strehl. Conditional probability tree estimation analysis and algorithms. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 51–58, 2009.

[Daumé III *et al.*, 2017] Hal Daumé III, Nikos Karampatziakis, John Langford, and Paul Mineiro. Logarithmic time one-against-some. In *Proceedings of the Thirty-Fourth International Conference on Machine Learning*, pages 923–932. JMLR. org, 2017.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Kamkar *et al.*, 2015] Iman Kamkar, Sunil Kumar Gupta, Dinh Phung, and Svetha Venkatesh. Stable feature selection for clinical prediction: Exploiting icd tree structure using tree-lasso. *Journal of biomedical informatics*, 53:277–290, 2015.

[Kim, 2014] Yoon Kim. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1746–1751, 2014.

[Kingma and Ba, 2015] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proceedings of the Third International Conference on Learning Representations*, 2015.

[Li *et al.*, 2018] Min Li, Zhihui Fei, Min Zeng, Fangxiang Wu, Yaohang Li, Yi Pan, and Jianxin Wang. Automated icd-9 coding via a deep learning approach. *IEEE/ACM transactions on computational biology and bioinformatics*, 2018.

[Lipton *et al.*, 2016] Zachary Chase Lipton, David C. Kale, Charles Elkan, and Randall C. Wetzel. Learning to diagnose with LSTM recurrent neural networks. In *4th International Conference on Learning Representations*, 2016.

[Luo *et al.*, 2017] Bingfeng Luo, Yansong Feng, Jianbo Xu, Xiang Zhang, and Dongyan Zhao. Learning to predict charges for criminal cases with legal basis. *arXiv preprint arXiv:1707.09168*, 2017.

[Mullenbach *et al.*, 2018] James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. Explainable prediction of medical codes from clinical text. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Languag Technologies*, pages 1101–1111, 2018.

[Perotte *et al.*, 2013] Adler Perotte, Rimma Pivovarov, Karthik Natarajan, Nicole Weiskopf, Frank Wood, and Noémie Elhadad. Diagnosis code assignment: models and evaluation metrics. *Journal of the American Medical Informatics Association*, 21(2):231–237, 2013.

[Prakash *et al.*, 2017] Aaditya Prakash, Siyuan Zhao, Sadid A Hasan, Vivek Datla, Kathy Lee, Ashequl Qadir, Joey Liu, and Oladimeji Farri. Condensed memory networks for clinical diagnostic inferencing. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

[Sha and Wang, 2017] Ying Sha and May D Wang. Interpretable predictions of clinical outcomes with an attention-based recurrent neural network. In *Proceedings of the 8th ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 233–240, 2017.

[Singh *et al.*, 2018] Gaurav Singh, James Thomas, Iain James Marshall, John Shawe-Taylor, and Byron C. Wallace. Structured multi-label biomedical text tagging via attentive neural tree decoding. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2837–2842, 2018.

[Song *et al.*, 2018] Yan Song, Shuming Shi, Jing Li, and Haisong Zhang. Directional skip-gram: Explicitly distinguishing left and right context for word embeddings. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 175–180, 2018.

[Subotin and Davis, 2016] Michael Subotin and Anthony R Davis. A method for modeling co-occurrence propensity of clinical codes with application to icd-10-pcs auto-coding. *Journal of the American Medical Informatics Association*, 23(5):866–871, 2016.

[Trevena, 2011] Lyndal Trevena. Wikiproject medicine, 2011.

[Wang *et al.*, 2016] Sen Wang, Xiaojun Chang, Xue Li, Guodong Long, Lina Yao, and Quan Z Sheng. Diagnosis code assignment using sparsity-based disease correlation embedding. *IEEE Transactions on Knowledge and Data Engineering*, 28(12):3191–3202, 2016.

[WHO, 1978] WHO. International classification of diseases:[9th] ninth revision, basic tabulation list with alphabetic index. 1978.

[Xiao *et al.*, 2018] 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*, 25(10):1419–1428, 2018.

[Xie and Xing, 2018] Pengtao Xie and Eric Xing. A neural architecture for automated icd coding. In *Proceedings of the Fifty-Sixth Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1066–1076, 2018.