Clinical Note Owns its Hierarchy: Multi-Level Hypergraph Neural Networks for Patient-Level Representation Learning

Nayeon Kim[1*], **Yinhua Piao**[2*], and **Sun Kim**[1,2,3,4]
[1] Interdisciplinary Program in Artificial Intelligence, Seoul National University
[2] Department of Computer Science and Engineering, Seoul National University
[3] Institute of Computer Technology, Seoul National University
[4] AIGENDRUG Co., Ltd.
{ny_1031, 2018-27910, sunkim.bioinfo}@snu.ac.kr

Abstract

Leveraging knowledge from electronic health records (EHRs) to predict a patient's condition is essential to the effective delivery of appropriate care. Clinical notes of patient EHRs contain valuable information from healthcare professionals, but have been underused due to their difficult contents and complex hierarchies. Recently, hypergraph-based methods have been proposed for document classifications. Directly adopting existing hypergraph methods on clinical notes cannot sufficiently utilize the hierarchy information of the patient, which can degrade clinical semantic information by (1) *frequent neutral words* and (2) *hierarchies with imbalanced distribution*. Thus, we propose a taxonomy-aware multi-level hypergraph neural network (TM-HGNN), where multi-level hypergraphs assemble useful neutral words with rare keywords via note and taxonomy level hyperedges to retain the clinical semantic information. The constructed patient hypergraphs are fed into hierarchical message passing layers for learning more balanced multi-level knowledge at the note and taxonomy levels. We validate the effectiveness of TM-HGNN by conducting extensive experiments with MIMIC-III dataset on benchmark in-hospital-mortality prediction.[1]

1 Introduction

With improvement in healthcare technologies, electronic health records (EHRs) are being used to monitor intensive care units (ICUs) in hospitals. Since it is crucial to schedule appropriate treatments for patients in ICUs, there are many prognostic models that use EHRs to address related tasks, such as in-hospital mortality prediction. EHRs consist of three types of data; structured, semi-structured, and unstructured. Clinical notes, which are unstructured data, contain valuable comments or summary of the patient's condition written by medical professionals (doctors, nurses, etc.). However, compared to structured data, clinical notes have been underutilized in previous studies due to the difficult-to-understand contents and the complex hierarchies (Figure 1(a)). Transformer-based (Vaswani et al., 2017) methods like ClinicalBERT (Alsentzer et al., 2019; Huang et al., 2019a, 2020) have been proposed to pretrain on large-scale corpus from similar domains, and fine-tune on the clinical notes through transfer learning. While Transformer-based methods can effectively detect distant words compared to other sequence-based methods like convolutional neural networks (Kim, 2014; Zhang et al., 2015) and recurrent neural networks (Mikolov et al., 2010; Tai et al., 2015; Liu et al., 2016), there are still limitations of increasing computational complexity for long clinical notes (Figure 2).

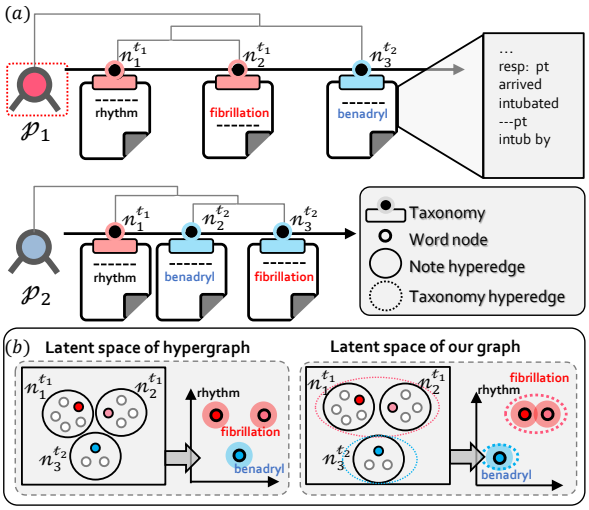

Figure 1: (a) Examples of patient clinical notes with difficult contents (e.g. jargons and abbreviations) and complex structures. Patient p_1 owns notes of radiology taxonomy (pink) and nursing taxonomy (blue). (b) Differences between existing hypergraphs and our proposed multi-level hypergraphs.

Recently, with the remarkable success of the graph neural networks (GNNs) (Kipf and Welling,

*These authors contributed equally to this work.
[1]Our codes and models are publicly available at: https://github.com/ny1031/TM-HGNN

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 5559–5573
July 9-14, 2023

2017; Veličković et al., 2018; Brody et al., 2021), graph-based document classification methods have been proposed (Yao et al., 2019; Huang et al., 2019b) that can capture long range word dependencies and can be adapted to documents with different and irregular lengths. Some methods build word co-occurrence graphs by sliding fixed-size windows to model pairwise interactions between words (Zhang et al., 2020; Piao et al., 2022; Wang et al., 2022). However, the density of the graph increases as the document becomes longer. Besides, there are also some methods apply hypergraph for document classification (Ding et al., 2020; Zhang et al., 2022a), which can alleviate the high density of the document graphs and extract high-order structural information of the documents.

Adopting hypergraphs can reduce burden for managing long documents with irregular lengths, but additional issues remain when dealing with clinical notes: ***(1) Neutral words deteriorate clinical semantic information.*** In long clinical notes, there are many frequently written neutral words (e.g. "*rhythm*") that do not directly represent the patient's condition. Most of the previous methods treat all words equally at the learning stage, which may result in dominance of frequent neutral words, and negligence of rare keywords that are directly related to the patient's condition. Meanwhile, the neutral word can occasionally augment information of rare keywords, depending on the intra-taxonomy context. Taxonomy represents the category of the clinical notes, where implicit semantic meaning of the words can differ. For example, "*rhythm*" occurred with "*fibrillation*" in *ECG* taxonomy can represent serious cardiac disorder of a patient, but when "*rhythm*" is written with "*benadryl*" in *Nursing* taxonomy, it can hardly represent the serious condition. Therefore, assembling intra-taxonomy related words can leverage "*useful*" neutral words with rare keywords to jointly augment the clinical semantic information, which implies the necessity of introducing taxonomy-level hyperedges. (2) ***Imbalanced distribution of multi-level hyperedges.*** There are a small number of taxonomies compared to notes for each patient. As a result, when taxonomy-level and note-level information are learned simultaneously, note-level information can obscure taxonomy-level information. To learn more balanced multi-level information of the clinical notes, an effective way for learning the multi-level hypergraphs with imbalanced distributed hyperedges is required.

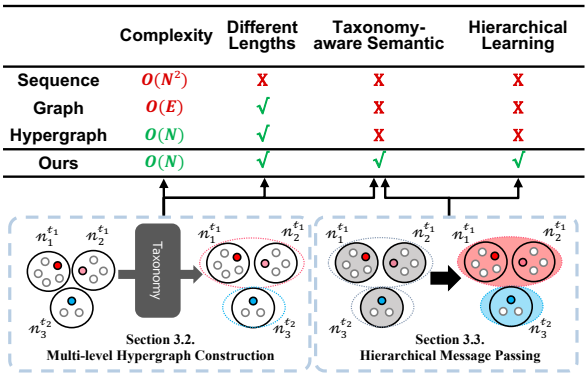

Figure 2: Advantages of the proposed model, compared to sequence, graph and hypergraph based models. N and E denote the number of nodes and edges respectively. We address issues of complexity and different lengths by adopting the hypergraph to represent each patient. Our model retains semantic information by constructing multi-level hypergraph (Section 3.2), and hierarchical message passing layers (Section 3.3) are proposed for balancing multi-level knowledge for patient representation learning.

To address the above issues, we propose TM-HGNN (Taxonomy-aware Multi-level HyperGraph Neural Networks), which can effectively and efficiently utilize the multi-level high-order semantic information for patient representation learning. Specifically, we adopt patient-level hypergraphs to manage highly unstructured and long clinical notes and define multi-level hyperedges, i.e., note-level and taxonomy-level hyperedges. Moreover, we conduct the hierarchical message passing from note-level to taxonomy-level hyperedges using edge-masking. To hierarchically learn word embeddings without mixture of information between note and taxonomy, note and taxonomy hyperedges are disconnected. Note-level word embeddings are learned only with intra-note local information. The following taxonomy-level propagation introduce clinical semantic information by assembling the intra-taxonomy words and separating inter-taxonomy words for better patient-level representation learning. The contributions of this article can be summarized as follows (Figure 2):

- To address issue 1, we construct multi-level hypergraphs for patient-level representation learning, which can assemble "*useful*" neutral word with rare keyword via note and taxonomy level hyperedges to retain the clinical semantic information.

- To address issue 2, we propose hierarchical message passing layers for the constructed graphs with imbalanced hyperedges, which can learn more balanced multi-level knowledge for patient-level representation learning.

- We conduct experiments with MIMIC-III clinical notes on benchmark in-hospital-mortality task. The experimental results demonstrate the effectiveness of our approach.

2 Related Work

2.1 Models for Clinical Data

With the promising potential of managing medical data, four benchmark tasks were proposed by Harutyunyan et al. (2019) for MIMIC-III (Medical Information Mart for Intensive Care-III) (Johnson et al., 2016) clinical dataset. Most of the previous works with MIMIC-III dataset focus on the structured data (e.g. vital signals with time-series) for prognostic prediction tasks (Choi et al., 2016; Shang et al., 2019) or utilize clinical notes combined with time-series data (Khadanga et al., 2019; Deznabi et al., 2021). Recently, there are approaches focused on clinical notes, adopting pre-trained models such as BERT-based (Alsentzer et al., 2019; Huang et al., 2019a; Golmaei and Luo, 2021; Naik et al., 2022) and XLNet-based (Huang et al., 2020) or utilizing contextualized phenotypic features extracted from clinical notes (Zhang et al., 2022b).

2.2 Graph Neural Networks for Document Classification

Graph neural networks (Kipf and Welling, 2017; Veličković et al., 2018; Brody et al., 2021) have achieved remarkable success in various deep learning tasks, including text classification. Initially, transductive graphs have been applied to documents, such as TextGCN (Yao et al., 2019). Transductive models have to be retrained for every renewal of the data, which is inefficient and hard to generalize (Yao et al., 2019; Huang et al., 2019b). For inductive document graph learning, word co-occurrence graphs initialize nodes with word embeddings and exploit pairwise interactions between words. TextING (Zhang et al., 2020) employs the gated graph neural networks for document-level graph learning. Following TextGCN (Yao et al., 2019) which applies graph convolutional networks (GCNs) (Kipf and Welling, 2017) in transductive level corpus graph, InducT-GCN (Wang et al., 2022) applies GCNs in inductive level where unseen documents are allowed to use. TextSSL (Piao et al., 2022) captures both local and global structural information within graphs.

However, the density of word co-occurrence graph increases as the document becomes longer, since the fixed-sized sliding windows are used to capture local pairwise edges. In case of hypergraph neural networks, hyperedges connect multiple number of nodes instead of connecting words to words by edges, which alleviates the high density of the text graphs. HyperGAT (Ding et al., 2020) proposes document-level hypergraphs with hyperedges containing sequential and semantic information. HEGEL (Zhang et al., 2022a) applies Transformer-like (Vaswani et al., 2017) multi-head attention to capture high-order cross-sentence relations for effective summarization of long documents. According to the reduced computational complexity for long documents (Figure 2), we adopt hypergraphs to represent patient-level EHRs with clinical notes. Considering issues of existing hypergraph-based methods (Figure 2), we construct multi-level hypergraphs at note-level and taxonomy-level for each patient. The constructed graphs are fed into hierarchical message passing layers to capture rich hierarchical information of the clinical notes, which can augment semantic information for patient representation learning.

3 Method

3.1 Problem Definition

Our task is to predict in-hospital-mortality for each patient using a set of clinical notes. Given a patient $p \in \mathcal{P}$ with in-hospital-mortality label $y \in \mathcal{Y}$, patient p owns a list of clinical notes $\mathcal{N}_p = [n_1^{t_1}, ..., n_j^{t_k}, ...]$, and each clinical note $n^t \in \mathcal{N}_p$ with taxonomy $t \in \mathcal{T}_p$ contains a sequence of words $\mathcal{W}_{n^t} = [w_1^{n^t}, ..., w_i^{n^t}, ...]$, where j, k and i denote the index of clinical note n, taxonomy t and word w of patient p. The set of taxonomies can be represented by $\mathcal{T} = \{t_1, t_2, ..., t_k, ...\}$.

Our goal is to construct individual multi-level hypergraphs $\mathcal{G}_p$ for each patient p and learn patient-level representation $\mathcal{G}_p$ with the multi-level knowledge by hierarchical message passing layers for in-hospital-mortality prediction task. Since our model is trained by inductive learning, patient p is omitted throughout the paper.

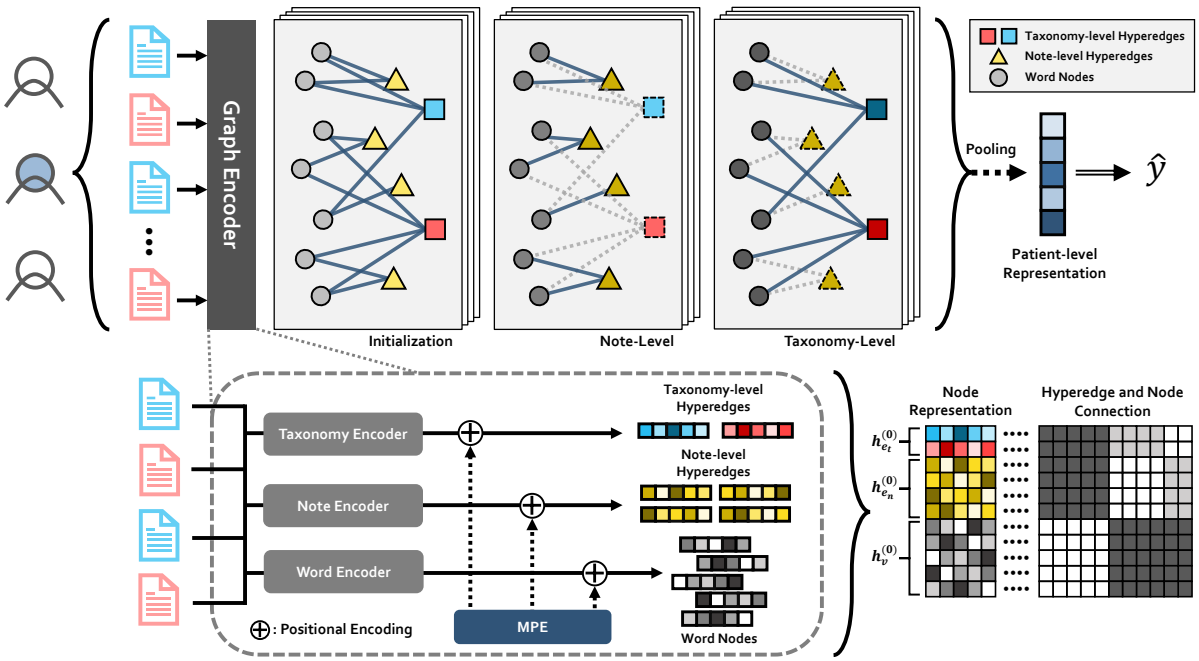

Figure 3: Overview of the proposed TM-HGNN. Taxonomy-aware multi-level hypergraphs are fed into the model for hierarchical message passing. $\hat{y}$ denotes the patient-level prediction.

3.2 Multi-Level Hypergraph Construction

We construct multi-level hypergraphs for patient-level representation learning, which can address the issues that are mentioned in introduction 1. A hypergraph $\mathcal{G}^* = (\mathcal{V}, \mathcal{E})$ consists of a set of nodes $\mathcal{V}$ and hyperedges $\mathcal{E}$ where multiple nodes can be connected to single hyperedge $e \in \mathcal{E}$. A multi-level hypergraph $\mathcal{G} = \{\mathcal{V}, \{\mathcal{E}_\mathcal{N} \cup \mathcal{E}_\mathcal{T}\}\}$ is constructed from patient's clinical notes, where $\mathcal{E}_\mathcal{N}$ and $\mathcal{E}_\mathcal{T}$ denote note-level and taxonomy-level hyperedges, respectively. A word node v exists in note n with the taxonomy of t can be represented by $\{v \in n, n \in t\}$. A note-level hyperedge is denoted as e_n, and a taxonomy-level hyperedge is denoted as e_t.

Multi-level Positional Encoding There are three types of entries in the multi-level hypergraph $\mathcal{G}$, such as word nodes $\mathcal{V}$, note-level hyperedges $\mathcal{E}_\mathcal{N}$ and taxonomy-level hyperedges $\mathcal{E}_\mathcal{T}$. To distinguish these entries, we propose multi-level positional encoding to introduce more domain-specific meta-information to the hypergraph $\mathcal{G}$. The function of multi-level positional encoding $\mathbf{MPE}(\cdot)$ can be defined as:

$$\mathbf{MPE}(x) = [\tau(x), \mathcal{I}_\mathcal{W}(x), \mathcal{I}_\mathcal{N}(x), \mathcal{I}_\mathcal{T}(x)] \quad (1)$$

where entry $x \in \{\mathcal{V}, \mathcal{E}_\mathcal{N}, \mathcal{E}_\mathcal{T}\}$, and function $\tau : x \mapsto \{0, 1, 2\}$ maps entry x to a single type among nodes, note-level and taxonomy-level hyperedges. Functions $\mathcal{I}_\mathcal{W}(\cdot)$, $\mathcal{I}_\mathcal{N}(\cdot)$, and $\mathcal{I}_\mathcal{T}(\cdot)$ maps entry x to positions in the word, note and taxonomy-level, respectively. To initialize embedding of node v, we concatenate embedding $\mathbf{MPE}(v)$ from multi-level position encoding and word2vec (Mikolov et al., 2010) pre-trained embedding $\mathbf{z}_v$. Since shallow word embeddings are widely used to initialize node embeddings in graph-based document representation (Grohe, 2020), we use word2vec (Mikolov et al., 2010) embedding. A word node embedding $\mathbf{h}_v^{(0)}$ is constructed as follows:

$$\mathbf{h}_v^{(0)} = \mathbf{MPE}(v) \oplus \mathbf{z}_v, \quad (2)$$

where $\oplus$ denotes concatenation function.

3.2.1 Hyperedge Construction

To extract multi-level information of patient-level representation using clinical notes, we construct patient hypergraphs with two types of hyperedges, one at the note-level hyperedge $\mathcal{E}_\mathcal{N}$ and the other at the taxonomy-level hyperedge $\mathcal{E}_\mathcal{T}$. A word node v in note n with taxonomy t is assigned to one note-level hyperedge e_n and one taxonomy-level hyperedge e_t, which can be defined as:

$$\mathcal{E}(v) = \{e_n, e_t | v \in n, n \in t\} \quad (3)$$

Note-level Hyperedges We adopt linear embedding function f_n and obtain the index embedding

using $\mathcal{I}_{\mathcal{N}}(n)$. To preserve time-dependent sequential information of clinical note n, we simply add time information $\mathbf{t}(n)$ to the embedding. Then initial embedding of note-level hyperedge $h_{e_n}^{(0)}$ with $\mathbf{MPE}(\cdot)$ can be defined as:

$$\mathbf{h}_{e_n}^{(0)} = \mathbf{MPE}(n) \oplus f_n^{\theta}\big(\mathcal{I}_{\mathcal{N}}(n), \mathbf{t}(n)\big), \tag{4}$$

where $\theta \in \mathbb{R}^{d \times d}$ denotes the parameter matrix of function f_n. Notably, we set the value of word index $\mathcal{I}_{\mathcal{W}}(n)$ as -1 since the note n represents higher level information than word v.

Taxonomy-level Hyperedges Taxonomy-level hyperedges e_t are constructed by taxonomy index $\mathcal{I}_{\mathcal{T}}(t)$ through linear layers f_t concatenated with $\mathbf{MPE}(\cdot)$ function, which can be defined as:

$$\mathbf{h}_{e_t}^{(0)} = \mathbf{MPE}(t) \oplus f_t^{\theta}\big(\mathcal{I}_{\mathcal{T}}(t)\big), \tag{5}$$

where $\theta \in \mathbb{R}^{d \times d}$ denotes the parameter matrix of function f_t. Like note-level hyperedge, we set $\mathcal{I}_{\mathcal{W}}(t)$ and $\mathcal{I}_{\mathcal{N}}(t)$ as -1 since the level of taxonomy t is higher than the levels of note and word.

3.3 Hierarchical Message Passing

To leverage the characteristics of two types of hyperedges, we propose a hierarchical hypergraph convolutional networks, composed of three layers that allow message passing from different types of hyperedges. In general, we define message passing functions for nodes and hyperedges as follows:

$$\mathcal{F}_{\mathcal{W}}(\mathrm{h}, \mathcal{E}, \theta) = \sigma\Bigg(\theta\Big(\sum_{u \in \mathcal{E}(v)} \frac{1}{\sqrt{\hat{d}_v}\sqrt{\hat{d}_u}} \mathrm{h}_u\Big)\Bigg), \tag{6}$$

$$\mathcal{F}_{\tau}(\mathrm{h}, \mathcal{V}^{\tau}, \theta) = \sigma\Bigg(\theta\Big(\sum_{z \in \mathcal{V}^{\tau}(e)} \frac{1}{\sqrt{\hat{d}_e}\sqrt{\hat{d}_z}} \mathrm{h}_z\Big)\Bigg), \tag{7}$$

where $\mathcal{F}_{\mathcal{W}}$ denotes message passing function for word nodes and $\mathcal{F}_{\tau}$ denotes message passing function for hyperedges with type $\tau \in \{1, 2\}$, i.e., note-level hyperedges and taxonomy-level hyperedges, respectively. Function $\mathcal{F}_{\mathcal{W}}$ updates word node embedding h_v by aggregating embeddings of connected hyperedges $\mathcal{E}(v)$. Function $\mathcal{F}_{\tau}$ updates hyperedge embedding h_e by aggregating embeddings of connected word nodes $\mathcal{V}^{\tau}(e)$. σ is the nonlinear activation function such as ReLU, $\theta \in \mathbb{R}^{d \times d}$ is the weight matrix with dimension d which can be differently assinged and learned at multiple levels.

Then we can leverage these defined functions to conduct hierarchical message passing learning at the note level and at the taxonomy level.

	Statistics
# of patients	17,927
# of ICU stays	21,013
# of in-hospital survival	18,231
# of in-hospital mortality	2,679
# of notes per ICU stay	13.29 (7.84)
# of words per ICU stay	1,385.62 (1,079.57)
# of words per note	104.25 (66.82)
# of words per taxonomy	474.75 (531.42)

Table 1: Statistics of the MIMIC-III clinical notes. Averaged numbers are reported with standard deviation.

Initialization Layer Due to the complex structure of the clinical notes, the initial multi-level hypergraph constructed for each patient has a large variance. To prevent falling into local optima in advance, we first use an initialization layer to pre-train the entries of hypergraphs by learning the entire patient graph structure. In this layer, message passing functions are applied to all word nodes $v \in \mathcal{V}$ and hyperedges $e \in \mathcal{E}_{\mathcal{I}} = \{\mathcal{E}_{\mathcal{N}} \cup \mathcal{E}_{\mathcal{T}}\}$. Thus, embeddings of node v, hyperedges e_n and e_t at both levels can be defined as:

$$h_I(v) = \mathcal{F}_{\mathcal{W}}\big(h_v^{(0)}, \mathcal{E}_{\mathcal{I}}(v), \theta_I\big), \tag{8}$$

$$h_I(e_n) = \mathcal{F}_{\tau}\big(h_{e_n}^{(0)}, \mathcal{V}^{\tau}(e_n), \theta_I\big), \tau = 1 \tag{9}$$

$$h_I(e_t) = \mathcal{F}_{\tau}\big(h_{e_t}^{(0)}, \mathcal{V}^{\tau}(e_t), \theta_I\big), \tau = 2 \tag{10}$$

Note-level Message Passing Layer Then we apply note-level message passing layer on hypergraphs with only word nodes $v \in \mathcal{V}$ and note-level hyperedges $e_n \in \mathcal{E}_{\mathcal{N}}$, and the taxonomy-level hyperedges are masked during message passing. In this layer, the word nodes can only interact with note-level hyperedges, which can learn the intra-note local information.

$$h_N(v) = \mathcal{F}_{\mathcal{W}}\big(h_I(v), \mathcal{E}_{\mathcal{N}}(v), \theta_N\big), \tag{11}$$

$$h_N(e_n) = \mathcal{F}_{\tau}\big(h_I(e_n), \mathcal{V}^{\tau}(e_n), \theta_N\big), \tau = 1, \tag{12}$$

$$h_N(e_t) = h_I(e_t) \tag{13}$$

Taxonomy-level Message Passing Layer The last layer is the taxonomy-level message passing layer, where all word nodes $v \in \mathcal{V}$ and taxonomy-level hyperedges $e_t \in \mathcal{E}_{\mathcal{T}}$ can be updated. In this layer, we block the hyperedges at the note level. The node representations with note-level information are fused with taxonomy information

via taxonomy-level hyperedges, which can assemble the intra-taxonomy related words to augment semantic information.

$$h_T(v) = \mathcal{F}_{\mathcal{W}}\big(h_N(v), \mathcal{E}_{\mathcal{T}}(v), \theta_T\big), \tag{14}$$

$$h_T(e_n) = h_N(e_n), \tag{15}$$

$$h_T(e_t) = \mathcal{F}_{\tau}\big(h_N(e_t), \mathcal{V}^{\tau}(e_t), \theta_T\big), \tau = 2 \tag{16}$$

3.3.1 Patient-Level Hypergraph Classification

After all aforementioned hierarchical message passing layers, node and hyperedge embeddings $h_T(v), h_T(e_n), h_T(e_t) \in \mathbf{H}_T$ follow mean-pooling operation which summarizes patient-level embedding z, which is finally fed into sigmoid operation as follows:

$$\hat{y} = \text{sigmoid}(z) \tag{17}$$

where $\hat{y}$ denotes the probability of the predicted label for in-hospital-mortality of the patient. The loss function for patient-level classification is defined as the binary cross-entropy loss:

$$\mathcal{L} = -\left(y \times \log \hat{y} + (1-y) \times \log(1-\hat{y})\right) \tag{18}$$

where y denotes the true label for in-hospital-mortality. The proposed network, TM-HGNN, can be trained by minimizing the loss function.

4 Experimental Settings

4.1 Dataset

We use clinical notes from the Medical Information Mart for Intensive Care III (MIMIC-III) (Johnson et al., 2016) dataset, which are written within 48 hours from the ICU admission. For quantitative evaluation, we follow Harutyunyan et al.'s (2019) benchmark setup for data pre-processing and train/test splits, then randomly divide 20% of train set as validation set. All patients without any notes are dropped during the data preparation. To prevent overfitting into exceptionally long clinical notes for a single patient, we set the maximum number of notes per patient into 30 from the first admission. Table 1 shows the statistics of preprocessed MIMIC-III clinical note dataset for our experiments. We select top six taxonomies for experiments, since the number of notes assigned to each taxonomy differs in a wide range (Appendix B Table 3). In addition, we select two chronic diseases, hypertension and diabetes, to compare prediction results for patients with each disease.

4.2 Compared Methods

In our experiments, the compared baseline methods for end-to-end training are as follows:

- Word-based methods: word2vec (Mikolov et al., 2013) with multi-layer perceptron classifier, and FastText (Joulin et al., 2017).
- Sequence-based methods: TextCNN (Kim, 2014), Bi-LSTM (Hochreiter and Schmidhuber, 1997), and Bi-LSTM with additional attention layer (Zhou et al., 2016).
- Graph-based methods: TextING (Zhang et al., 2020), InducT-GCN (Wang et al., 2022), and HyperGAT (Ding et al., 2020). In particular, HyperGAT represents hypergraph-based method, and the other graph-based methods employ word co-occurrence graphs.

4.3 Implementation Details

TM-HGNN is implemented by PyTorch (Paszke et al., 2019) and optimized with Adam (Kingma and Ba, 2015) optimizer with learning rate 0.001 and dropout rate 0.3. We set hidden dimension d of each layer to 64 and batch size to 32 by searching parameters. We train models for 100 epochs with early-stopping strategy, where the epoch of 30 shows the best results. All experiments are trained on a single NVIDIA GeForce RTX 3080 GPU.

5 Results

Since the dataset has imbalanced class labels for in-hospital mortality as shown in Table 1, we use AUPRC (Area Under the Precision-Recall Curve) and AUROC (Area Under the Receiver Operating Characteristic Curve) for precise evaluation. It is suggested by Davis and Goadrich (2006) to use AUPRC for imbalanced class problems.

5.1 Classification Performance

Table 2 shows performance comparisons of TM-HGNN and baseline methods. Sequence-based methods outperform word-based methods, which indicates capturing local dependencies between neighboring words benefits patient document classification. Moreover, all graph-based methods outperform sequence-based and word-based methods. This demonstrates ignoring sequential information of words is not detrimental to clinical notes. Furthermore, hypergraphs are more effective than previous word co-occurrence graphs, indicating that

Categories	Models	Whole		Hypertension		Diabetes	
		AUPRC	AUROC	AUPRC	AUROC	AUPRC	AUROC
Word-based	Word2vec + MLP	13.49 ± 1.68	56.65 ± 5.12	16.82 ± 1.78	53.56 ± 4.20	18.15 ± 1.42	51.94 ± 3.40
	FastText	17.06 ± 0.08	62.37 ± 0.11	25.56 ± 0.28	62.39 ± 0.18	31.33 ± 0.33	67.59 ± 0.20
Sequence-based	Bi-LSTM	17.67 ± 4.19	58.75 ± 5.78	21.75 ± 5.25	57.39 ± 6.11	27.52 ± 7.57	61.86 ± 8.38
	Bi-LSTM w/ Att.	17.96 ± 0.61	62.63 ± 1.31	26.05 ± 1.80	63.24 ± 1.57	33.01 ± 3.53	68.89 ± 1.58
	TextCNN	20.34 ± 0.67	68.25 ± 0.54	27.10 ± 1.82	66.10 ± 1.20	36.89 ± 2.54	71.83 ± 1.69
Graph-based	TextING	34.50 ± 7.79	78.20 ± 4.27	36.63 ± 8.30	80.12 ± 4.05	36.13 ± 8.66	80.28 ± 3.84
	InducT-GCN	43.03 ± 1.96	82.23 ± 0.72	41.06 ± 2.95	85.56 ± 1.24	40.59 ± 3.07	84.42 ± 1.45
HyperGraph-based	HyperGAT	44.42 ± 1.96	84.00 ± 0.84	42.32 ± 1.78	86.41 ± 1.01	40.08 ± 2.45	85.03 ± 1.20
	T-HGNN (Ours)	45.85 ± 1.91	84.29 ± 0.31	43.53 ± 2.01	87.07 ± 0.64	40.47 ± 2.29	85.48 ± 0.92
	TM-HGNN (Ours)	**48.74 ± 0.60**	**84.89 ± 0.42**	**47.27 ± 1.21**	**87.75 ± 0.54**	**42.22 ± 1.25**	**85.86 ± 0.73**

Table 2: Classification performance comparison on patient-level clinical tasks, evaluated with AUPRC and AUROC in percentages. We report averaged results with standard deviation over 10 random seeds. Values in boldface denote the best results.

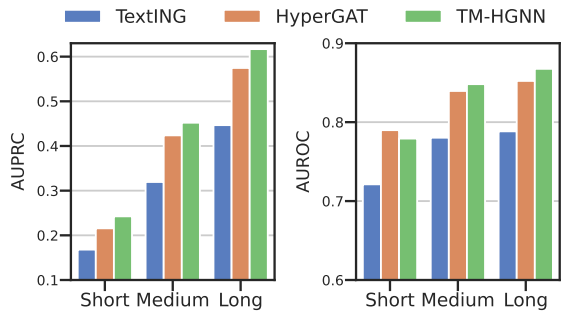

Figure 4: Prediction results of TextING, HyperGAT, and TM-HGNN for three patient-level clinical note groups divided by length (short, medium, and long). AUPRC and AUROC are used for evaluation.

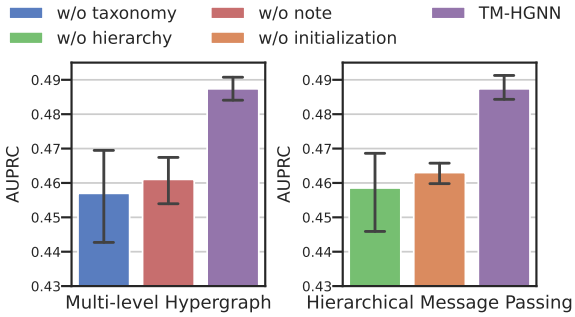

Figure 5: Performance results of ablation studies. The effectiveness of the multi-level hypergraph and hierarchical message passing in the proposed model TM-HGNN are validated respectively.

it is crucial to extract high-order relations within clinical notes. In particular, as TM-HGNN outperforms HyperGAT (Ding et al., 2020), exploiting taxonomy-level semantic information which represents the medical context of the notes aids precise prediction in patient-level. Another advantage of our model, which captures multi-level high order relations from note-level and taxonomy-level with hierarchy, can be verified by the results in Table 2 where TM-HGNN outperforms T-HGNN. T-HGNN indicates the variant of TM-HGNN, which considers note-level and taxonomy-level hyperedges homogeneous. Likewise, results from hypertension and diabetes patient groups show similar tendencies in overall.

5.2 Robustness to Lengths

To evaluate the performance dependencies to lengths, we divide clinical notes in patient-level into three groups by lengths, which are short, medium, and long (Appendix B, Figure 8). For test set, the number of patients is 645, 1,707, and 856 for short, medium, and long group each, and the percentage of mortality is 6.98%, 10.72%, and 15.89% for each group, which implies patients in critical condition during ICU stays are more likely to have long clinical notes. Figure 4 shows performance comparisons for three divided groups with TextING (Zhang et al., 2020) which utilizes word co-occurrence graph, HyperGAT (Ding et al., 2020), a ordinary hypergraph based approach, and our multi-level hypergraph approach (TM-HGNN). All three models were more effective to longer clinical notes, which demonstrates graph based models are robust to long document in general. Among the three models, our proposed TM-HGNN mostly performs the best and HyperGAT (Ding et al., 2020) follows, and then TextING (Zhang et al., 2020). The results demonstrate that our TM-HGNN, which exploits taxonomy-level semantic information, is most effective for clinical notes regardless of the lengths, compared to other graph-based approaches.

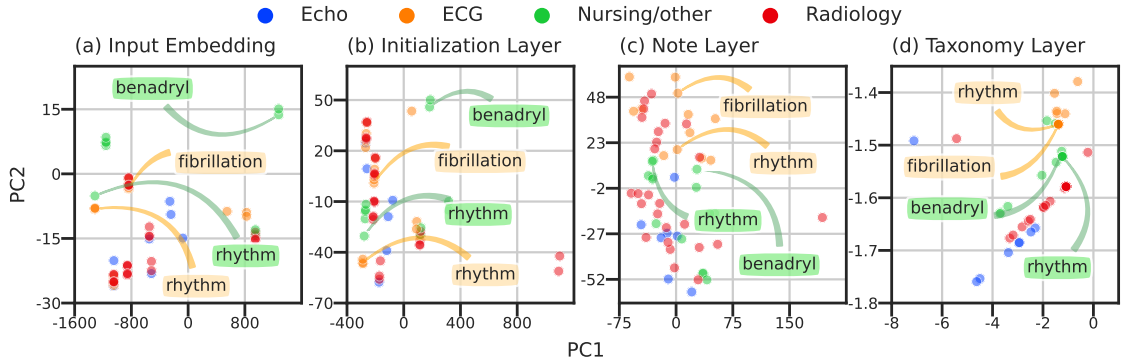

Figure 6: PCA results of learned node representations from each layer of TM-HGNN, for patient case HADM_ID=147702. "*Rhythm*" and "*fibrillation*" from ECG, "*rhythm*" and "*benadryl*" from Nursing/other taxonomy are highlighted. (a) Input word node embeddings. (b) Initialized node embeddings from the first layer. (c) After second layer, note-level message passing. (d) Final node embeddings from TM-HGNN, after taxonomy-level message passing. Word node embeddings are aligned with the same taxonomy words.

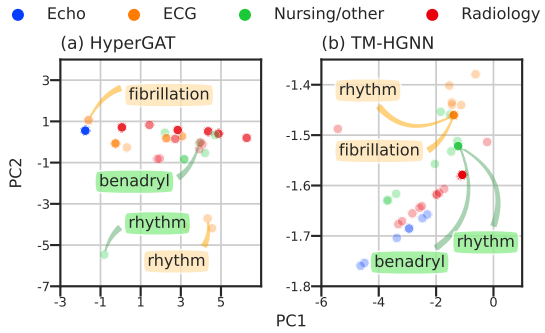

Figure 7: PCA results of learned node representations from HyperGAT (a) and TM-HGNN (b). "*Rhythm*" and "*fibrillation*" from ECG, "*rhythm*" and "*benadryl*" from Nursing/other taxonomy are highlighted.

5.3 Ablation Study

Effect of Multi-level Hypergraph In order to validate the effect of multi-level hypergraphs, we ignore taxonomy-level and note-level hyperedges respectively. *w/o taxonomy*, which ignores taxonomy-level hyperedges, deteriorates the performance most significantly. *w/o note* shows degraded performance as well. Thus, effectiveness of multi-level hypergraph construction for patient representation learning can be verified (Figure 5).

Effect of Hierarchical Message Passing Figure 5 demonstrates that hierarchical message passing (note-level to taxonomy-level) for multi-level hypergraphs is effective than learning without hierarchies, since *w/o hierarchy* shows inferior performance compared to TM-HGNN. *w/o hierarchy* represents T-HGNN from Table 2, which considers every hyperedge as homogeneous. Degraded performance from *w/o initialization* shows the effectiveness of the initialization layer before hierarchical message passing, which indicates that pretraining on the entire multi-level hypergraphs first benefits the patient-level representation learning.

5.4 Case Study

Hierarchical Message Passing We visualize the learned node representations based on principal component analysis (PCA) (Jolliffe, 2002) results, as hierarchical message passing continues in TM-HGNN. In Figure 6(a), "*rhythm*" from ECG and Nursing/other taxonomy are mapped closely for initial word embeddings, since they are literally same words. As the patient-level hypergraphs are fed into a global-level, note-level, and taxonomy-level convolutional layers in order, words in the same taxonomies assemble, which can be found in Figure 6(b), (c), and (d). As a result, "*rhythm*" of ECG represents different semantic meanings from "*rhythm*" of Nursing/other, as it is learned considerably close to "*fibrillation*" from the same taxonomy.

Importance of Taxonomy-level Semantic Information To investigate the importance of taxonomy-level semantic information extraction, we visualize PCA results of the learned node embeddings from the baseline method and the proposed TM-HGNN. We select patient with hospital admission id (HADM_ID) 147702 for case study since TM-HGNN successfully predicts the true label for in-hospital-mortality, which is pos-

itive, but the other baseline methods show false negative predictions. As in Figure 7, HyperGAT learns "*rhythm*" without taxonomy-level semantic information, since it is not assembled with other words in the same taxonomy. But TM-HGNN separately learns "*rhythm*" from ECG and "*rhythm*" from Nursing/other based on different contexts, which results in same taxonomy words aligned adjacently, such as "*fibrillation*" of ECG and "*benadryl*" of Nursing/other. Therefore, in case of TM-HGNN, frequently used neutral word "*rhythm*" from ECG with a word "*fibrillation*" means an irregular "*rhythm*" of the heart and is closely related to mortality of the patient, but "*rhythm*" from Nursing/other with another nursing term remains more neutral. This phenomenon demonstrates that contextualizing taxonomy to frequent neutral words enables differentiation and reduces ambiguity of the frequent neutral words (e.g. "*rhythm*"), which is crucial to avoid false negative predictions on patient-level representation learning.

6 Conclusion

In this paper, we propose a taxonomy-aware multi-level hypergraph neural networks, TM-HGNN, a novel approach for patient-level clinical note representation learning. We employ hypergraph-based approach and introduce multi-level hyperedges (note and taxonomy-level) to address long and complex information of clinical notes. TM-HGNN aims to extract high-order semantic information from the multi-level patient hypergraphs in hierarchical order, note-level and then taxonomy-level. Clinical note representations can be effectively learned in an end-to-end manner with TM-HGNN, which is validated from extensive experiments.

Limitations

Since our approach, TM-HGNN, aggregates every note during ICU stays for patient representation learning, it is inappropriate for time-series prediction tasks (e.g. vital signs). We look forward to further study that adopts and applies our approach to time-series prediction tasks.

Ethics Statement

In MIMIC-III dataset (Johnson et al., 2016), every patient is deidentified, according to Health Insurance Portability and Accountability Act (HIPAA) standards. The fields of data which can identify the patient, such as patient name and address, are completely removed based on the identifying data list provided in HIPAA. In addition, the dates for ICU stays are shifted for randomly selected patients, preserving the intervals within data collected from each patient. Therefore, the personal information for the patients used in this study is strictly kept private. More detailed information about deidentification of MIMIC-III can be found in Johnson et al. (2016).

Acknowledgements

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [NO.2021-0-01343, Artificial Intelligence Graduate School Program (Seoul National University)] and the Bio & Medical Technology Development Program of the National Research Foundation (NRF) funded by the Ministry of Science & ICT (RS-2023-00257479), and the ICT at Seoul National University provides research facilities for this study.

References

Emily Alsentzer, John R Murphy, Willie Boag, Wei-Hung Weng, Di Jin, Tristan Naumann, WA Redmond, and Matthew BA McDermott. 2019. Publicly available clinical bert embeddings. *NAACL HLT 2019*, page 72.

Shaked Brody, Uri Alon, and Eran Yahav. 2021. How attentive are graph attention networks? In *International Conference on Learning Representations*.

Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. 2016. Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. *Advances in neural information processing systems*, 29.

Jesse Davis and Mark Goadrich. 2006. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, page 233–240, New York, NY, USA. Association for Computing Machinery.

Iman Deznabi, Mohit Iyyer, and Madalina Fiterau. 2021. Predicting in-hospital mortality by combining clinical notes with time-series data. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4026–4031.

Kaize Ding, Jianling Wang, Jundong Li, Dingcheng Li, and Huan Liu. 2020. Be more with less: Hypergraph attention networks for inductive text classification.

In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4927–4936.

Sara Nouri Golmaei and Xiao Luo. 2021. Deepnotegnn: predicting hospital readmission using clinical notes and patient network. In *Proceedings of the 12th ACM Conference on Bioinformatics, Computational Biology, and Health Informatics*, pages 1–9.

Martin Grohe. 2020. Word2vec, node2vec, graph2vec, x2vec: Towards a theory of vector embeddings of structured data. In *Proceedings of the 39th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS'20, page 1–16, New York, NY, USA. Association for Computing Machinery.

Hrayr Harutyunyan, Hrant Khachatrian, David C Kale, Greg Ver Steeg, and Aram Galstyan. 2019. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):1–18.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Kexin Huang, Jaan Altosaar, and Rajesh Ranganath. 2019a. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *arXiv preprint arXiv:1904.05342*.

Kexin Huang, Abhishek Singh, Sitong Chen, Edward Moseley, Chih-Ying Deng, Naomi George, and Charolotta Lindvall. 2020. Clinical xlnet: Modeling sequential clinical notes and predicting prolonged mechanical ventilation. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 94–100.

Lianzhe Huang, Dehong Ma, Sujian Li, Xiaodong Zhang, and Houfeng Wang. 2019b. Text level graph neural network for text classification. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3444–3450.

Alistair EW Johnson, Tom J Pollard, Lu Shen, Li-wei H Lehman, Mengling Feng, Mohammad Ghassemi, Benjamin Moody, Peter Szolovits, Leo Anthony Celi, and Roger G Mark. 2016. Mimic-iii, a freely accessible critical care database. *Scientific data*, 3(1):1–9.

Ian T Jolliffe. 2002. *Principal component analysis*. Wiley.

Armand Joulin, Édouard Grave, Piotr Bojanowski, and Tomáš Mikolov. 2017. Bag of tricks for efficient text classification. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 427–431.

Swaraj Khadanga, Karan Aggarwal, Shafiq Joty, and Jaideep Srivastava. 2019. Using clinical notes with time series data for icu management. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6432–6437.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR (Poster)*.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*.

Yikuan Li, Ramsey M Wehbe, Faraz S Ahmad, Hanyin Wang, and Yuan Luo. 2023. A comparative study of pretrained language models for long clinical text. *Journal of the American Medical Informatics Association*, 30(2):340–347.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2016. Recurrent neural network for text classification with multi-task learning. In *IJCAI*.

Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Aakanksha Naik, Sravanthi Parasa, Sergey Feldman, Lucy Wang, and Tom Hope. 2022. Literature-augmented clinical outcome prediction. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 438–453.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Yinhua Piao, Sangseon Lee, Dohoon Lee, and Sun Kim. 2022. Sparse structure learning via graph neural networks for inductive document classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 11165–11173.

Junyuan Shang, Cao Xiao, Tengfei Ma, Hongyan Li, and Jimeng Sun. 2019. Gamenet: Graph augmented memory networks for recommending medication combination. In *proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 1126–1133.

Kai Sheng Tai, Richard Socher, and Christopher D Manning. 2015. Improved semantic representations from tree-structured long short-term memory networks. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1556–1566.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *International Conference on Learning Representations*.

Kunze Wang, Soyeon Caren Han, and Josiah Poon. 2022. Induct-gcn: Inductive graph convolutional networks for text classification. *arXiv preprint arXiv:2206.00265*.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 7370–7377.

Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2022a. Hegel: Hypergraph transformer for long document summarization. *arXiv preprint arXiv:2210.04126*.

Jingqing Zhang, Luis Daniel Bolanos Trujillo, Ashwani Tanwar, Julia Ive, Vibhor Gupta, and Yike Guo. 2022b. Clinical utility of automatic phenotype annotation in unstructured clinical notes: intensive care unit use. *BMJ Health & Care Informatics*, 29(1):e100519.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28:649–657.

Yufeng Zhang, Xueli Yu, Zeyu Cui, Shu Wu, Zhongzhen Wen, and Liang Wang. 2020. Every document owns its structure: Inductive text classification via graph neural networks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 334–339.

Peng Zhou, Wei Shi, Jun Tian, Zhenyu Qi, Bingchen Li, Hongwei Hao, and Bo Xu. 2016. Attention-based bidirectional long short-term memory networks for relation classification. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers)*, pages 207–212.

A Detailed Statistics of MIMIC-III Clinical Notes

Table 3 shows the number of clinical notes assigned to 15 predefined taxonomies in MIMIC-III dataset. Since the number of notes varies in a wide range for each taxonomy, we select top six taxonomies for experiments: Radiology, ECG, Nursing/other, Echo, Nursing, and Physician.

Figure 8 shows histogram for the number of words per patient-level clinical notes in train set. Since 682, 1,070, and 1,689 are the first, second, and third quantile of the train data, we select 600 and 1,600 as the boundaries to divide test set into 3 groups (short, medium, and long), which is used to validate proposed TM-HGNN's robustness to lengths.

B Node Representations from Other Methods

Figure 9 shows PCA results of learned node representations from three different models. According to Figure 9(a) and 9(b), word co-occurrence graphs (TextING) and homogeneous single-level hypergraphs (HyperGAT) show node representations ambiguous to discriminate by taxonomies, since every taxonomy has been shuffled. In Figure 9(c), node embeddings are aligned adjacently and arranged with similar pattern for the same taxonomies. This verifies the effectiveness of the proposed TM-HGNN which captures intra- and inter-taxonomy semantic word relations for patient-level representation learning. Example words (voltage, lvef, benadryl, and obliteration) which are generally used in each taxonomy are shown in Figure 9 to emphasize that the keywords from each taxonomy are learned adjacently to words similar in context within taxonomies in case of TM-HGNN, but not for other methods.

C Explanation of the Medical Terms

- Fibrillation : Fibrillation refers to rapid and irregular contractions of the muscle fibers, especially from the heart. It can lead to serious heart conditions.

- Benadryl : Brand name for the drug Diphenhydramine, which is an antihistamine. Benadryl is one of the over-the-counter drugs, and generally used for alleviating the allergic symptoms.

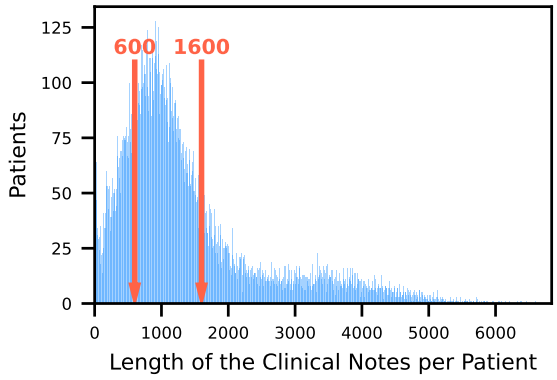

Figure 8: Histogram for the length of patient-level clinical notes in train set. 600 and 1,600 are selected as boundaries to divide clinical notes into three groups (short, medium, and long).

	# of Notes
Radiology	17,466
ECG	16,410
Nursing/other	12,347
Echo	7,935
Nursing	3,562
Physician	3,545
Respiratory	2,024
Nutrition	1,270
General	1,135
Discharge Summary	608
Rehab Services	594
Social Work	424
Case Management	162
Consult	19
Pharmacy	14

Table 3: The number of clinical notes for 15 predefined taxonomies in MIMIC-III dataset.

- Lvef : Abbreviation of left ventricular ejection fraction, which is the ratio of stroke volume to end-diastolic volume. Lvef is known as the central measure for the diagnosis and management of heart failure.
- Obliteration : In Radiology, obliteration refers to the disappearance of the contour of an organ, due to the same x-ray absorption from the adjacent tissue.

D Additional Performance Comparison

We conduct additional experiments using LSTM based on 17 code features selected by Johnson et al. (2016), and Transformer-based ClinicalXLNet (Huang et al., 2020) without pre-training for in-hospital mortality prediction. Table 4 shows that

Models	AUPRC	AUROC
LSTM (code features)	39.86	81.98
ClinicalXLNet (w/o pretrain)	16.77	62.16
TM-HGNN (Ours)	**48.74**	**84.89**

Table 4: Classification performance comparison on patient-level in-hospital-mortality prediction task, evaluated with AUPRC and AUROC in percentages. Values in boldface denote the best results.

Models	AUROC	F1
Clinical-Longformer	0.762	**0.484**
TM-HGNN (Ours)	**0.847**	0.462

Table 5: Classification performance comparison on patient-level acute kidney injury prediction task, evaluated with AUROC and F1 score. Values in boldface denote the best results.

TM-HGNN outperforms approaches using structured data and Transformer-based model without pre-training.

In addition, we train our model on acute kidney injury prediction task (MIMIC-AKI) following Li et al. (2023). Table 5 shows comparative results of our TM-HGNN to Clinical-Longformer (Li et al., 2023) that justify TM-HGNN can effectively utilize high-order semantics from long clinical notes, with much less computational burden compared to long sequence transformer models.

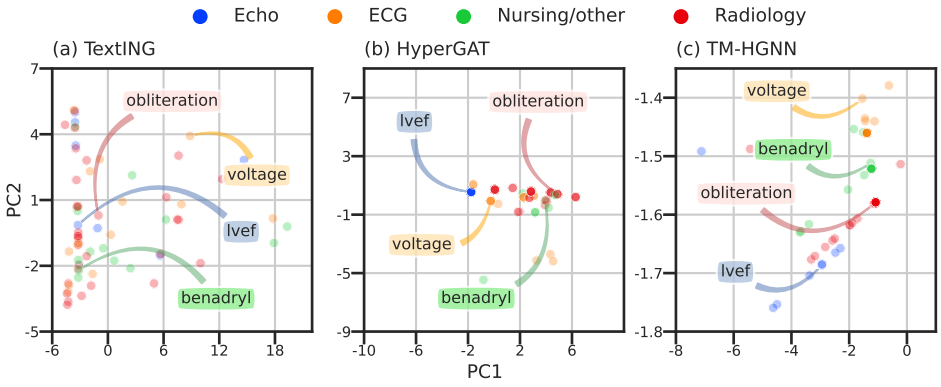

Figure 9: PCA results of learned node representations for patient case HADM_ID=147702, compared with baseline methods. (a) Final node embeddings from TextING. (b) Final node embeddings from HyperGAT. (c) Final node embeddings from the proposed TM-HGNN.

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
Section "Limitations"

☑ A2. Did you discuss any potential risks of your work?
Section "Ethics Statement"

☑ A3. Do the abstract and introduction summarize the paper's main claims?
Section "Introduction"

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

Section 4

☑ B1. Did you cite the creators of artifacts you used?
Section 4.1, Section 4.2

☑ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Section 4.1

☑ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Section 4.2

☑ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Section "Ethics Statement"

☑ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Section 4.1

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Section 4.1

C ☑ Did you run computational experiments?

Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
Section 4.3

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
Section 4.3

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
Section 5.1, Section 5.2, Section 5.3

☑ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Section 3.2, Section 4.1

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
Not applicable. Left blank.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
Not applicable. Left blank.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
Not applicable. Left blank.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
Not applicable. Left blank.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
Not applicable. Left blank.