Self-Attention Enhanced Patient Journey Understanding in Healthcare System

Xueping Peng[1](⊠), Guodong Long[1], Tao Shen[1], Sen Wang[2], and Jing Jiang[1]

[1] AAII, FEIT, University of Technology Sydney, Australia
{xueping.peng, guodong.long, jing.jiang}@uts.edu.au,
Tao.Shen@student.uts.edu.au
[2] School of Information Technology and Electrical Engineering, The University of
Queensland, Australia sen.wang@uq.edu.au

Abstract. Understanding patients' journeys in healthcare system is a fundamental prepositive task for a broad range of AI-based healthcare applications. This task aims to learn an informative representation that can comprehensively encode hidden dependencies among medical events and its inner entities, and then the use of encoding outputs can greatly benefit the downstream application-driven tasks. A patient journey is a sequence of electronic health records (EHRs) over time that is organized at multiple levels: patient, visits and medical codes. The key challenge of patient journey understanding is to design an effective encoding mechanism which can properly tackle the aforementioned multi-level structured patient journey data with temporal sequential visits and a set of medical codes. This paper proposes a novel self-attention mechanism that can simultaneously capture the contextual and temporal relationships hidden in patient journeys. A multi-level self-attention network (MusaNet) is specifically designed to learn the representations of patient journeys that is used to be a long sequence of activities. We evaluated the efficacy of our method on two medical application tasks with real-world benchmark datasets. The results have demonstrated the proposed MusaNet produces higher-quality representations than state-of-the-art baseline methods. The source code is available in https://github.com/xueping/MusaNet.

Keywords: Electronic health records · Self-attention Network · Medical Outcome · Healthcare · Bi-directional.

1 Introduction

Healthcare system stores huge volumes of electronic health records (EHR) that contain detailed admission information about patients over a period of time [12, 29]. The data is organised in multi-level structure: the patient journey level, the patient visit level, and the medical code level. A useful example of this structure shown in Fig. 1. In the example, an anonymous patient visits his/her doctor and is admitted to hospital on different days. The diagnoses and procedures performed at each of these visits are recorded as industry-standard medical codes. Each medical code at the lowest level records an independent observation while
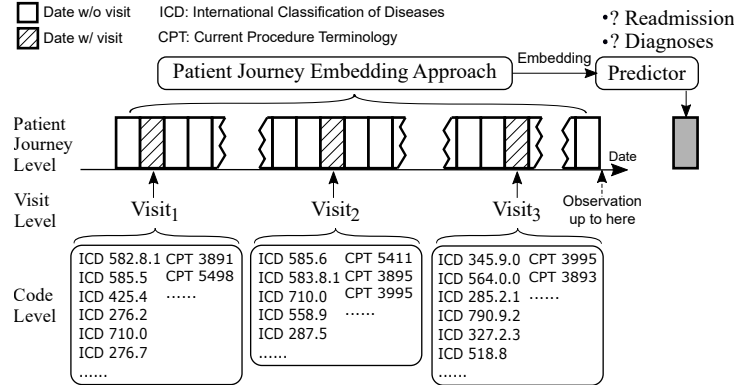
Fig. 1. An example of a patient's medical records in multi-level structure – from top to bottom: patient journey level, individual visit level, and medical code level.

the set of codes at higher levels can depict the medical conditions of a patient at a given time point. At the top level, for each patient, all occurrences of medical events at different time-stamps are chained together as a patient journey, which offers more informative details. Understanding patient activity based on a patient's journey, such as re-admissions and future diagnoses, is a core research task that of significant value for healthcare system. For example, re-admission statistics can be used to measure the quality of care. Diagnoses can be used to more fully understand a patient's problems and medical research [26]. However, researchers have faced many challenges in their attempts to represent patient journeys from EHR data with temporality, high-dimensionality, and irregularity [25].

Recurrent neural network (RNN) has been widely used to model medical events in sequential EHR data for clinical prediction tasks [2, 3, 5, 16, 25, 18]. For instance, Ref. [5, 2] indirectly exploit an RNN to embed the visit sequences into a patient representation by multi-level representation learning to integrate visits and medical codes based on visit sequences and the co-occurrence of medical codes. Other research works have, however, used RNNs directly to model time-ordered patient visits for predicting diagnoses [4, 3, 16, 25, 18]. Convolutional Neural Network (CNN) also has been exploited to represent patient journey. For example, Ref. [22] transforms a record into a sequence of discrete elements separated by coded time gaps, and then employ CNN to detect and combine predictive local clinical motifs to stratify the risk. Ref. [30] utilises CNN in code level to learn visit embedding. These RNN and CNN based models follow ideas of processing sentences [13] in documents from NLP to treat a patient journey as a document and a visit as a sentence, which only has a sequential relationship, while two arbitrary visits in one patient journey may be separated by different time intervals. The interval between two visits, which has been largely disregarded in the existing studies on patient journey representation, can be modelled as auxiliary information fed into the supervised algorithms.

Recently, research has proposed the integration of attention mechanisms and RNNs to model sequential EHR data [4, 16, 26, 24]. Ref. [32] used a sole attention mechanism to construct a sequence-to-sequence model for a neural machine translation task that achieved a state-of-the-art quality score. According to Ref. [27], attention allows for more flexibility in sequence lengths than RNNs, and is more task/data-driven when modelling dependencies. Unlike sequential models, attention is easy to compute. Computation can also be significantly accelerated with distributed/parallel computing schemes. However, to the best of our knowledge, a neural network based entirely on attention has never been designed for analytics with EHR data.

To fill this gap in the literature, and address some of the open issues listed above, we propose a novel attention mechanism called masked self-attention (mSA) for temporal context fusion that uses self-attention to capture contextual information and the temporal dependencies between patient's visits. In addition, we propose an end-to-end neural network, called multi-level self-attention network (MusaNet) to understand medical outcomes using a learned representation of a patient journey based solely on the proposed attention mechanism. MusaNet constructs a multi-level self-attention representation to simultaneously represent visits and patient journeys using attention pooling and mSA layers. It is worth noting that, compared to the existing RNN and CNN-based methods, MusaNet can yield better prediction performance for a long sequence of medical records.

Experiments conducted on two prediction tasks with two public EHR datasets demonstrate that the proposed MusaNet is superior to several state-of-the-art baseline methods.

To summarize, our main contributions are:

- a novel attention mechanism, called mSA, that uses self-attention to capture the contextual information and long-term dependencies between patient's visits;
- an end-to-end neural network called "MusaNet" that predicts medical outcomes using a learned representation of a patient journey based solely on the proposed attention mechanism; and
- an evaluation of the proposed model on two benchmark datasets with two prediction tasks, demonstrating that the MusaNet model is superior to all the compared methods.

2 Background

2.1 Definitions and Notations

Definition 1 (Medical Code). *A medical code is a term or entry to describe a diagnosis, procedure, medication, or laboratory test administered to a patient. A set of medical codes is formally denoted as $X = \{x_1, x_2, \ldots, x_{|X|}\}$, where $|X|$ is the total number of unique medical codes in the EHR data.*

Definition 2 (Visit). *A visit is a hospital stay from admission to discharge with an admission time stamp. A visit is denoted as $V_i = < x_i, t_i >$, where*

$x_i = [x_{i1}, x_{i2}, ..., x_{ik}]$, i is the i-th visit, t_i is the admission time of the i-th visit, k is the total number of medical codes in a visit.

Definition 3 (Patient Journey). *A patient journey consists of a sequence of visits over time, denoted as $J = [V_1, V_2, ..., V_n]$, where n is the total number of visits for the patient.*

Definition 4 (Temporal position). *Temporal position refers to a difference in days between admission time t_i of the i-th visit and admission time t_1 of the first visit in a patient journey, denoted as $p_i = |t_i - t_1|$, where $i = 1, \ldots, n$.*

Definition 5 (Task). *Given a set of patient journeys $\{J_l\}_{l=1,...}$, the task is to predict medical outcomes.*

In the remainder of the paper, a patient's medical data means a stored and chronological sequence of m visits in a patient journey J_i. To reduce clutter, we omit the superscript and subscript (i) indicating i-th patient, when discussing a single patient journey.

Table 1 summarizes notations we will use throughout the paper.

Table 1. Notations for BiteNet.

Notation	Description
X	Set of unique medical codes
$|X|$	The size of unique medical concept
V_i	The i-th visit of the patient
v_i	The representation of t-th visit of the patient
$\boldsymbol{v}$	Sequence of n visit embeddings of the patient
x_i	Set of medical codes in V_i
$\boldsymbol{x}$	Sequence of medical codes, $[x_1, x_2, ..., x_n]$
e_i	Set of medical code embeddings in $\boldsymbol{x}_i$
$\boldsymbol{e}$	A sequence of medical code embeddings, $[e_1, e_2, ..., e_n]$
J	A patient journey consisting of a sequence of visits over time
d	The dimension of medical code embedding

2.2   Medical Code Embedding

The concept of word embedding was introduced to medical analytics by Ref. [19, 20] as a way to learn low-dimensional real-valued distributed vector representations of medical concepts for downstream tasks instead of using discrete medical codes. Hence, medical code embedding is a fundamental processing unit for learning EHRs in a deep neural network. Formally, given a sequence or set of medical concepts $\boldsymbol{c} = [c_1, c_2, ..., c_n] \in \mathbb{R}^{|C| \times n}$, where c_i is a one-hot vector, and n is the sequence length. In the NLP literature, a word embedding method like word2vec [19, 20] converts a sequence of one-hot vectors into a sequence of low-dimensional vectors $\boldsymbol{e} = [e_1, e_2, ..., e_n] \in \mathbb{R}^{d \times n}$, where d is the embedding dimension of e_i.

This process can be formally written as $\boldsymbol{e} = W^{(e)}\boldsymbol{c}$, where $W^{(e)} \in \mathbb{R}^{d \times |C|}$ is the embedding weight matrix, which can be fine-tuned during the training phase.

A visit can be represented by the set of medical codes embedded with real-valued dense vectors. A straightforward approach to learning this representation v_i is to sum the embeddings of medical codes in the visit, which is written as

$$v_i = \sum_{e_{ik} \in e_i} e_{ik}, \tag{1}$$

where e_i is the set of medical code embeddings in the i-th visit, and e_{ik} is the k-th code embedding in e_i. A visit can also be represented as a real-valued dense vector with a more advanced method, such as self-attention and attention pooling discussed below.

2.3   Attention Mechanism

Vanilla Attention. Given a patient journey consisting of a sequence of visits $\boldsymbol{J} = [v_1, v_2, ..., v_m]$ and a vector representation of a query $q \in \mathbb{R}^d$, vanilla attention [1] computes the alignment score between q and each visit v_i using a compatibility function $f(v_i, q)$. A softmax function then transforms the alignment scores $\alpha \in \mathbb{R}^n$ to a probability distribution $p(z|\boldsymbol{J}, q)$, where z is an indicator of which visit is important to q. A large $p(z = i|\boldsymbol{J}, q)$ means that v_i contributes important information to q. This attention process can be formalized as

$$\alpha = [f(v_i, q)]_{i=1}^n, \tag{2}$$
$$p(z|\boldsymbol{J}, q) = softmax(\alpha). \tag{3}$$

The output s is the weighted average of sampling a visit according to its importance, i.e.,

$$s = \sum_{i=1}^n p(z = i|\boldsymbol{J}, q) \cdot v_i. \tag{4}$$

Additive attention is a commonly-used attention mechanism [1] , where a compatibility function $f(\cdot)$ is parameterized by a multi-layer perceptron (MLP), i.e.,

$$f(v_i, q) = w^T \sigma(W^{(1)} v_i + W^{(2)} q + b^{(1)}) + b, \tag{5}$$

where $W^{(1)} \in \mathbb{R}^{d \times d}$, $W^{(2)} \in \mathbb{R}^{d \times d}, w \in \mathbb{R}^d$ are learnable parameters, and $\sigma(\cdot)$ is an activation function.

Ref. [27] recently proposed a multi-dimensional (multi-dim) attention mechanism as a way to further improve an attention module's ability to model contexts through feature-wise alignment scores. So, to model more subtle context and dependency relationships, the score might be large for some features but small for others. Formally, $P_{ki} \triangleq p(z_k = i|\boldsymbol{J}, q)$ denotes the attention probability of the i-th visit on the k-th feature dimension, where the attention score is calculated from a multi-dim compatibility function by replacing the weight vector w in Eq. (5) with a weight matrix. Note that, for simplicity, we ignore the subscript k if it does not cause any confusion. The attention result can, therefore, be written as $s = \sum_{i=1}^n P_{\cdot i} \odot v_i$

Sophisticated Self-attention Mechanism. Information about the contextual and temporal relationships between two individual visits v_i and v_j in the same patient journey $\boldsymbol{J}$ is captured with a self-attention mechanism [32] because a context-aware representation probably leads to better empirical performance [9, 32, 27].

Specifically, the query q in an attention compatibility function, such as a multi-dim compatibility function, is replaced with v_j , i.e.,

$$f(v_i, v_j) = W^T \sigma(W^{(1)} v_i + W^{(2)} v_j + b^{(1)}) + b. \tag{6}$$

Similar to the probability matrix P in multi-dim attention, each input individual visit v_j is associated with a probability matrix P_j such that $P_{ki}^j \triangleq p(z_k = i|\boldsymbol{J}, v_j)$. The output representation for each v_j is

$$s_j = \sum_{i=1}^{n} P_{\cdot i}^j \odot v_i \tag{7}$$

The final output of self-attention is $\boldsymbol{s} = [s_1, s_2, \ldots, s_n]$, each of which is a visit-context embedded representation for each individual visit.

However, a fatal defect in previous self-attention mechanisms for NLP tasks is that they are order-insensitive and therefore, cannot model temporal relationships between the visits in a patient journey. Even if position embedding [32] is used to alleviate this problem, it is designed for consecutive natural language words rather than patient visits with arbitrary intervals.

Attention Pooling. Attention pooling [14, 15] explores the importance of each individual visit within an entire patient journey given a specific task. It works by compressing a sequence of visit embeddings from a patient journey into a single context-aware vector representation for downstream classification or regression. Formally, q is removed from the common compatibility function, which is written as

$$f(v_i) = W^T \sigma(W^{(1)} v_i + b^{(1)}) + b. \tag{8}$$

The multi-dim attention probability matrix P is defined as $P_{ki} \triangleq p(z_k = i|\boldsymbol{J})$. The final output of attention pooling, which is used as sequence encoding, is similar to the self-attention mechanism above, i.e.,

$$s = \sum_{i=1}^{n} P_{\cdot i} \odot v_i \tag{9}$$

3   Proposed Model

This section begins by introducing the masked self-attention module, followed by the MusaNet context fusion module for predicting medical outcomes.
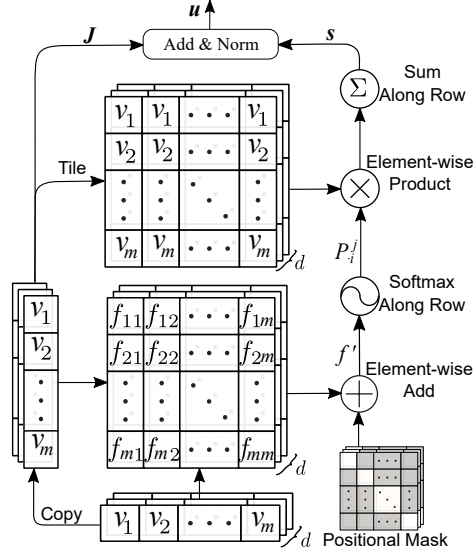
Fig. 2. The masked self-attention mechanism (mSA). f_{ij} is defined as $f(v_i, v_j)$ in Eq. (6). f' is defined as $f'(v_i, v_j)$ in Eq.(10). $P_{\cdot i}^{j}$ is formally defined in Eq.(13), and s is defined in Eq. (7).

3.1   Masked Self-Attention

Because self-attention was originally designed for NLP tasks, it does not consider temporal relationships (e.g., interval between visits) within inputs. Obviously, it is very important when modeling sequential medical events. Inspired by the work of Ref. [27, 28] with masked self-attention, we developed masked self-attention (mSA) to capture the contextual and temporal relationships between visits. The structure of mSA is shown in Fig. 2.

The self-attention mechanism outlined in Eq.(6) is rewritten into a temporal-dependent format:

$$f'(v_i, v_j) = f(v_i, v_j) + M_{ij}^{pos}, \tag{10}$$

where $f(v_i, v_j) = W^T \sigma(W^{(1)} v_i + W^{(2)} v_j + b^{(1)}) + b$ captures the contextual dependency between the i-th visit and the target j-th visit. In this context, masks can be used to encode temporal order information into an attention output. Our approach also incorporates three masks, such as, forward M^{fw}, backward M^{bw} and diagonal mask M_{ij}^{diag}, i.e.,

$$M_{ij}^{fw} = \begin{cases} 0, & i < j \\ -\infty, & otherwise \end{cases} \tag{11}$$

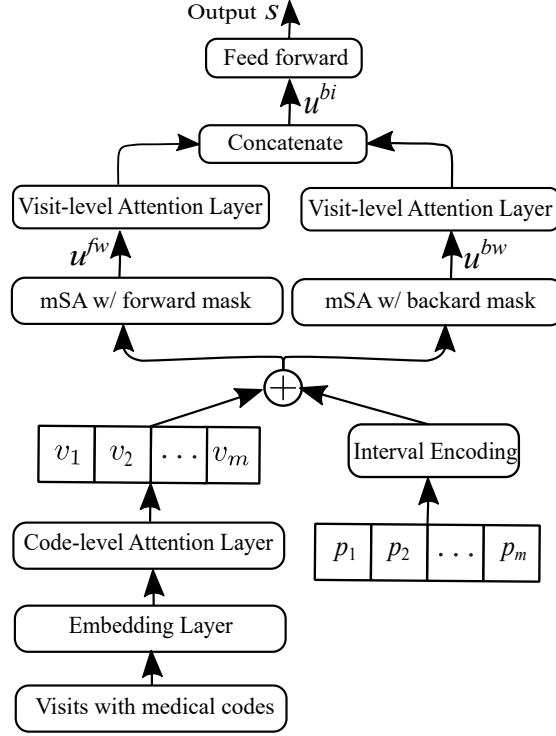$$M_{ij}^{bw} = \begin{cases} 0, & i > j \\ -\infty, & otherwise \end{cases} \tag{12}$$

Fig. 3. The multi-level self-attention network (MusaNet) framework. The inputs are m sequential visits with medical codes. m is the maximal number of visits in individual patient journeys over the dataset, and padding is applied if the number of visits in a patient journey is less than m.

The forward mask M^{fw}, only attends later visits j to earlier visits i, and vice versa with the backward one. Here, let $M_{ij}^{pos} \in \{M_{ij}^{fw}, M_{ij}^{bw}\}$.

Given an input patient journey $\boldsymbol{J}$ and a positional mask M, $f'(v_i, v_j)$ is computed according to Eq. (10) followed by the standard procedure of self-attention to compute the probability matrix P^j for each v_j as follows:

$$P_{\cdot i}^j \triangleq [p(z_k = i | \boldsymbol{J}, v_j)]_{k=1}^d, \tag{13}$$

The output $\boldsymbol{s}$ is computed as Eq.(7). Then, layer normalization ($Norm$) and rectified linear unit ($ReLU$) is used to generate the output $\boldsymbol{u}$ as follows,

$$\boldsymbol{u} = \text{Norm}(\text{ReLU}(\boldsymbol{J} + \boldsymbol{s})). \tag{14}$$

3.2   Multi-level Self-attention Network

We propose a patient journey embedding model called the "Multi-lelvel Self-Attention Network (MusaNet)" with mSA as its major components. The architecture of MusaNet is shown in Fig. 3. In MusaNet, the embedding layer

is applied to the input medical codes of visits, and its output is processed by the code-level attention layer to generate visit embeddings $[v_1, v_2, ..., v_m]$ given in Eq.(9). The visit interval encodings are then added to the visit embeddings, which is followed by two parameter-untied mSA blocks with forward mask M^{fw} in Eq.(11) and M^{bw} in Eq.(12), respectively. Their outputs are denoted by $u^{fw}, u^{bw} \in \mathbb{R}^{d \times m}$. The visit-level attention layers in Eq.(9) are applied to the outputs followed by the concatenation layer to generate output $u^{bi} \in \mathbb{R}^{2d}$. Lastly, a feed-forward layer consisting of dense and softmax layers is employed to generate a categorical distribution over targeted categories.

Interval Encoding. Although mSA incorporates information on the order of visits in a patient journey, the relative time interval between visits is an important factor in longitudinal studies. We include information on the visit intervals $p = [p_1, p_2, ..., p_m]$ in the sequence. In particular, we add interval encodings to the visit embeddings of the sequence. The encoding is performed by mapping interval t to the same lookup table during both training and prediction. The d-dimensional interval encoding is then added to the visit embedding with the same dimension.

4 Experiments

4.1 Data Description

Dataset. MIMIC-III [11] is an open-source, large-scale, de-identified dataset of ICU patients and their EHRs. The dataset consists of 46k+ ICU patients with 58k+ visits collected over 11 years. In this paper, we consider two sub-datasets: Dx and Dx&Tx, where Dx is a dataset which only includes diagnosis codes for each visit, and Dx&Tx is another dataset which includes diagnosis and procedure codes for each visit.

Data pre-processing. We chose patients who made at least two visits. All infrequent diagnoses codes were removed and the threshold was empirically set to 5. In summary, we extract 7,499 patients with an average of 2.66 visits per patient; the average number of diagnoses and procedures each visit are 13 and 4, respectively.

4.2 Experiment Setup

Baseline models. We compare the performance of our proposed model against several baseline models[3]:

---

[3] GRAM [3] and KAME [18] and MMORE [31] are not baselines as they use external knowledge to learn the medical code representations.

- *RNN* and *BRNN*, We directly embed visit information into the vector representation v_t by summation of embedded medical cods in the visit, and then feed this embedding to the GRU. The hidden state h_t produced by the GRU is used to predict the $(t+1)$-th visit information.
- *RETAIN* [4], which learns the medical concept embeddings and performs heart failure prediction via the reversed RNN with the attention mechanism.
- *Deepr* [22]: which is a multilayered architecture based on convolutional neural networks (CNNs) that learn to extract features from medical records and predict future risk.
- *Dipole* [16], which uses bidirectional recurrent neural networks and three attention mechanisms (location-based, general, concatenation-based) to predict patient visit information.
- *SAnD* [30]: which employs a masked, self-attention mechanism, and uses positional encoding and dense interpolation strategies to incorporate temporal order to generate a sequence-level prediction.
- *Transformer Encoder* (TransEnc) [32]: which only considers Transformer with one encoder layer to replace mSA in MusaNet.

Validation Tasks. The two tasks we selected to evaluate the performance of our proposed model are to predict re-admission and future diagnosis [26].

- **Re-admission (Readm)** is a standard measure of the quality of care. We predicted unplanned admissions within 30 days following a discharge from an indexed visit. A visit is considered a "re-admission" if admission date is within thirty days after discharge of an eligible indexed visit [26].
- **Diagnoses** reflect the model's understanding of a patient's problems. In the experiments, we aim to predict diagnosis categories instead of the real diagnosis codes, which are the nodes in the second hierarchy of the ICD9 codes[4].

Evaluation metrics. The two evaluation metrics used are:

- **PR-AUC:** (Area under Precision-Recall Curve), to evaluate the likelihood of re-admission. PR-AUC is considered to be a better measure for imbalanced data like ours [7].
- **precision@k:** which is defined as the correct medical codes in top k divided by $\min(k, |y_t|)$, where $|y_t|$ is the number of category labels in the $(t+1)$-th visit [18]. We report the average values of precision@k in the experiments. We vary k from 5 to 30. The greater the value, the better the performance.

Implementation. We implement all the approaches with Tensorflow 2.0. For training models, we use RMSprop with a minbatch of 32 patients and 10 epochs. The drop-out rate is 0.1 for all the approaches. The data split ratio is 0.8:0.1:0.1 for training, validation and testing sets. In order to fairly compare the performance, we set the same embedding dimension $d = 128$ for all the baselines and the proposed model.

---

[4] http://www.icd9data.com

Table 2. Prediction performance comparison of future re-admission and diagnoses (Dx is for diagnosis, and Tx is for procedure).

Data	Model	Readm (PR-AUC)	Diagnosis Precision@k			
			k=5	k=10	k=20	k=30
Dx	RNN	0.3021	0.6330	0.5874	0.6977	0.8068
	BRNN	0.3119	0.6362	0.5925	0.7014	0.8128
	RETAIN	0.3014	0.6498	0.5948	0.6999	0.8102
	Deepr	0.2999	0.6434	0.5865	0.6981	0.8113
	Dipole	0.2841	0.6484	0.5997	0.7034	0.8121
	SAnD	0.2979	0.6179	0.5709	0.6805	0.7959
	TransEnc	0.3116	0.6482	0.5980	0.7037	0.8139
	MusaNet	**0.3261**	**0.6507**	**0.6069**	**0.7104**	**0.8227**
Dx & Tx	RNN	0.3216	0.6317	0.5857	0.6973	0.8093
	BRNN	0.3270	0.6402	0.5961	0.7088	0.8138
	RETAIN	0.3161	0.6497	0.6021	0.7061	0.8148
	Deepr	0.3142	0.6391	0.5947	0.7004	0.8125
	Dipole	0.2899	0.6515	0.6097	0.7121	0.8149
	SAnD	0.2996	0.6242	0.5774	0.6878	0.8004
	TransEnc	0.3137	0.6559	0.6086	0.7105	0.8144
	MusaNet	**0.3344**	**0.6618**	**0.6127**	**0.7146**	**0.8242**

4.3   Results

Overall Performance. Tab. 2 reports the results of the two prediction tasks on the two datasets - future re-admissions and diagnoses. The results show that MusaNet outperforms all the baselines. This demonstrates that the superiority of our framework results from the explicit consideration of inherent hierarchy of EHRs, and the contextual and temporal dependencies which are incorporated into the representations. Furthmore, we note that the performance obtained by the models remains approximately the same (RETAIN, RNN) or even drops by up to 0.43% (Deepr) after adding the procedure to the training data over Precision@5. The underlying reason may be that the future diagnosis prediction is less sensitive to the procedures, thus the relationships among procedures cannot be well captured when predicting few diagnoses. Furthermore, by using attention mechanisms, Dipole, SAnD, TransEnc, and our MusaNet model achieve a marginal improvement when comparing the performance of using Dx&Tx for training and that of Dx. This implies that attention could play an important role in the learned representations of the procedures.

Robustness for lengthy sequence of visits. We conducted a set of experiments to evaluate the robustness of MusaNet by varying the length of sequential visits of a patient from 6 to 16, which are considered to be long patient journeys in the medical domain. Fig. 4 shows the results. Overall, the model MusaNet outperforms the other models as the length of a patient's sequential visits increases in diagnoses prediction both Dx and Dx&Tx. In particular, the performance of
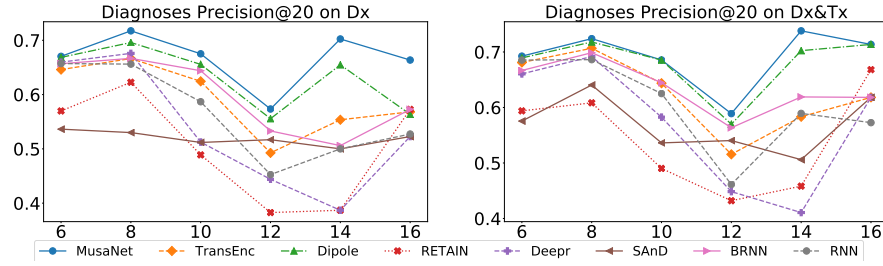
Fig. 4. Robustness comparison regarding precision@20 on Dx and Dx&Tx. Length of visit sequence varies from 6 to 16.

Table 3. Ablation Performance Comparison.

Data	Ablation	Readm (PR-AUC)	Diagnosis Precision@k			
			k=5	k=10	k=20	k=30
Dx	Attention	0.3170	0.6371	0.5921	0.7044	0.8142
	PosMask	0.3033	0.6287	0.5821	0.7044	0.8125
	Interval	0.3132	0.6318	0.5907	0.6977	0.8080
	MusaNet	**0.3261**	**0.6507**	**0.6069**	**0.7104**	**0.8227**
Dx&Tx	Attention	0.3278	0.6459	0.5997	0.7030	0.8104
	PosMask	0.3237	0.6457	0.5957	0.7097	0.8174
	Interval	0.3290	0.6485	0.6050	0.7141	0.8189
	MusaNet	**0.3344**	**0.6618**	**0.6127**	**0.7146**	**0.8242**

Dipole is comparative to our MusaNet and follows a similar trend with an increase in the length of sequential visits. From these results, a conclusion can be drawn that the positional masks in the mSA module and interval encoding in our framework play a vital role in capturing the lengthy temporal dependencies between patient's sequential visits.

Ablation Study. We performed a detailed ablation study to examine the contributions of each of the model's components to the prediction tasks. There are three components, which are (Attention) the two attention pooling layers to learn the visits from the embedded medical codes and learn the patient journey from the embedded visits, (PosMask) the position mask in mSA module, and (Interval) the interval encodings to be added to the learned visit embeddings.

- **Attention:** replace each of the two attention pooling layers with a simple summation layer;
- **PosMask:** remove the forward and backward position mask in the mSA module;
- **Interval:** remove the interval encodings module;
- **MusaNet:** our proposed model in the paper.

From Tab. 3, we see that the full complement of MusaNet achieved superior accuracy to the ablated models. Specifically, we note that the position mask from

Fig. 5. Importance of visits for patient 1 (Left), and importance of diagnoses in visit 1 (Right).
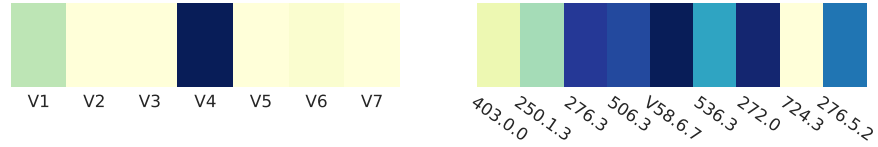

Fig. 6. Importance of visits for patient 2 (Left), and importance of diagnoses in visit 4 (Right).

the mSA module (PosMask) contributes the highest accuracy to re-admission prediction over Dx, which gives us confidence in using the position mask to learn the patient journey representations without sufficient data. Moreover, it is clear that the attention pooling component provides valuable information with the learned embeddings of the patient journey for the performance of diagnoses prediction over Dx&Tx. Specifically, MusaNet predicted re-admissions by 2.83% more accurately on Dx and by 1.07% more accurately on Dx&Tx. Similarly, MusaNet predicted diagnoses precision@5 by 2.2% more accurately on Dx and diagnoses precision@30 by 1.4% more accurately on Dx&Tx.

Case Study: Visualization and Explainability. One aspect of this method is that it hierarchically compresses medical codes into visits and visits into patient journeys. At each level of aggregation, the model decides the importance of the lower-level entries on the upper-level entry, which makes the model explainable. To showcase this feature, we visualized two patient journeys. These examples come from the re-admission prediction task with the Dx dataset. From the importance distribution of the patient visits, we analyzed the most important visits to future re-admission. For example, Visit 1 in Fig. 5 was the most influential factor on Patient 1's re-admission. After zooming on Visit 1, we found a vital insight in Diagnosis 3 (ICD 585.6), i.e., end-stage renal disease, which would obviously cause frequent and repeated re-admissions to hospital. As shown in Fig. 6, Patient 2 had 7 visits. Visit 4 contributed most to the re-admissions. Again, the diagnoses reveal the cause: long-term use of insulin (ICD V58.6.7) and pure hypercholesterolemia (ICD 272.0).

5   Related Work

5.1   Medical Concept Embedding

Borrowing ideas from word representation models [10, 20, 19, 33, 23], researchers in the healthcare domain have recently explored the possibility of creating representations of medical concepts. Much of this research has focused on the Skip-gram model. For example, Minarro-Gimnez et al. [21] directly applied Skip-gram to learn representations of medical text, and Vine et al. [8] did the same for UMLS medical concepts. Choi et al. [6] went a step further and used the Skip-gram model to learn medical concept embeddings from different data sources, including medical journals, medical claims, and clinical narratives. In other work [2], Choi et al. developed the Med2Vec model based on Skip-gram to learn concept-level and visit-level representations simultaneously. The shortcoming of all these models is that they view EHRs as documents in the NLP sense, which means that temporal information is ignored.

5.2   Patient Journey Embedding

Applying deep learning to healthcare analytical tasks has recently attracted enormous interest in healthcare informatics. RNN has been widely used to model medical events in sequential EHR data for clinical prediction tasks [2, 3, 5, 16, 25, 18]. Choi et al. [2, 5] indirectly exploit an RNN to embed the visit sequences into a patient representation by multi-level representation learning to integrate visits and medical codes based on visit sequences and the co-occurrence of medical codes. Other research works have, however, used RNNs directly to model time-ordered patient visits for predicting diagnoses [4, 3, 16, 25, 18, 17, 34]. CNN has been exploited to represent a patient journey in other way. For example, Nguyen et al. [22] transform a record into a sequence of discrete elements separated by coded time gaps, and then employ CNN to detect and combine predictive local clinical motifs to stratify the risk. These RNN- and CNN-based models follow ideas of processing sentences [13] in documents from NLP to treat a patient journey as a document and a visit as a sentence, which only has a sequential relationship, while two arbitrary visits in one patient journey may be separated by different time intervals. Attention-based neural networks have been exploited successfully in healthcare tasks to model sequential EHR data [4, 16, 26, 30] and have been shown to improve predictive performance.

6   Conclusion

In this paper, we proposed a novel multi-level self-attention network (**MusaNet**) to encode patient journeys. The model framework comprises a masked self-attention module (mSA) for capturing the contextual information and the temporal relationships, attention poolings for tackling the multi-level structure of EHR data, and interval encoding for encoding temporal information of visits. As demonstrated by experiment results, MusaNet produces better representations that is validated using two downstream healthcare tasks.

7   Acknowledgments

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. arXiv:1409.0473 (2014)
2. Choi, E., Bahadori, M.T., Searles, E., Coffey, C., Thompson, M., Bost, J., Tejedor-Sojo, J., Sun, J.: Multi-layer representation learning for medical concepts. In: SIGKDD. pp. 1495–1504. ACM (2016)
3. Choi, E., Bahadori, M.T., Song, L., Stewart, W.F., Sun, J.: Gram: graph-based attention model for healthcare representation learning. In: SIGKDD. pp. 787–795. ACM (2017)
4. Choi, E., Bahadori, M.T., Sun, J., Kulas, J., Schuetz, A., Stewart, W.: Retain: An interpretable predictive model for healthcare using reverse time attention mechanism. In: NeurIPS. pp. 3504–3512 (2016)
5. Choi, E., Xiao, C., Stewart, W., Sun, J.: Mime: Multilevel medical embedding of electronic health records for predictive healthcare. In: NeurIPS. pp. 4552–4562 (2018)
6. Choi, Y., Chiu, C.Y.I., Sontag, D.: Learning low-dimensional representations of medical concepts. AMIA Summits on Translational Science Proceedings **2016**, 41 (2016)
7. Davis, J., Goadrich, M.: The relationship between precision-recall and roc curves. In: ICML. pp. 233–240. ACM (2006)
8. De Vine, L., Zuccon, G., Koopman, B., Sitbon, L., Bruza, P.: Medical semantic similarity with a neural language model. In: CIKM. pp. 1819–1822. ACM (2014)
9. Hu, M., Peng, Y., Huang, Z., Qiu, X., Wei, F., Zhou, M.: Reinforced mnemonic reader for machine reading comprehension. arXiv:1705.02798 (2017)
10. Jha, K., Wang, Y., Xun, G., Zhang, A.: Interpretable word embeddings for medical domain. In: 2018 IEEE International Conference on Data Mining (ICDM). pp. 1061–1066. IEEE (2018)
11. Johnson, A.E., Pollard, T.J., Shen, L., Li-wei, H.L., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Celi, L.A., Mark, R.G.: Mimic-iii, a freely accessible critical care database. Scientific data **3**, 160035 (2016)
12. Kho, A.N., Yu, J., Bryan, M.S., Gladfelter, C., Gordon, H.S., Grannis, S.J., et al.: Privacy-preserving record linkage to identify fragmented electronic medical records in the all of us research program. In: Machine Learning and Knowledge Discovery in Databases, ECML PKDD. Communications in Computer and Information Science, vol. 1168, pp. 79–87. Springer (2019)
13. Kim, Y.: Convolutional neural networks for sentence classification. In: EMNLP. pp. 1746–1751 (Oct 2014)
14. Lin, Z., Feng, M., Santos, C.N.d., Yu, M., Xiang, B., Zhou, B., Bengio, Y.: A structured self-attentive sentence embedding. arXiv:1703.03130 (2017)
15. Liu, Y., Sun, C., Lin, L., Wang, X.: Learning natural language inference using bidirectional lstm model and inner-attention. arXiv:1605.09090 (2016)
16. Ma, F., Chitta, R., Zhou, J., You, Q., Sun, T., Gao, J.: Dipole: Diagnosis prediction in healthcare via attention-based bidirectional recurrent neural networks. In: SIGKDD. pp. 1903–1911. ACM (2017)

17. Ma, F., Gao, J., Suo, Q., You, Q., Zhou, J., Zhang, A.: Risk prediction on electronic health records with prior medical knowledge. In: SIGKDD. pp. 1910–1919. ACM (Jul 2018)
18. Ma, F., You, Q., Xiao, H., Chitta, R., Zhou, J., Gao, J.: KAME: Knowledge-based attention model for diagnosis prediction in healthcare. In: CIKM. pp. 743–752. ACM (Oct 2018)
19. Mikolov, T., Chen, K., Corrado, G., Dean, J.: Efficient estimation of word representations in vector space. arXiv:1301.3781 (2013)
20. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NeurIPS. pp. 3111–3119 (2013)
21. Minarro-Giménez, J.A., Marin-Alonso, O., Samwald, M.: Exploring the application of deep learning techniques on medical text corpora. Stud Health Technol Inform **205**, 584–588 (2014)
22. Nguyen, P., Tran, T., Wickramasinghe, N., Venkatesh, S.: Deepr: A convolutional net for medical records. arXiv:1607.07519 (2016)
23. Peng, X., Long, G., Pan, S., Jiang, J., Niu, Z.: Attentive dual embedding for understanding medical concepts in electronic health records. In: 2019 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2019)
24. Peng, X., Long, G., Shen, T., Wang, S., Jiang, J., Blumenstein, M.: Temporal self-attention network for medical concept embedding. 2019 IEEE International Conference on Data Mining (ICDM) pp. 498–507 (2019)
25. Qiao, Z., Zhao, S., Xiao, C., Li, X., Qin, Y., Wang, F.: Pairwise-ranking based collaborative recurrent neural networks for clinical event prediction. In: IJCAI. pp. 3520–3526 (2018)
26. Rajkomar, A., Oren, E., Chen, K., Dai, A.M., Hajaj, N., Hardt, M., Liu, P.J., Liu, X., Marcus, J., Sun, M., et al.: Scalable and accurate deep learning with electronic health records. npj Digital Medicine **1**(1), 18 (2018)
27. Shen, T., Zhou, T., Long, G., Jiang, J., Pan, S., Zhang, C.: Disan: Directional self-attention network for rnn/cnn-free language understanding. In: AAAI (2018)
28. Shen, T., Zhou, T., Long, G., Jiang, J., Zhang, C.: Bi-directional block self-attention for fast and memory-efficient sequence modeling. arXiv:1804.00857 (2018)
29. Shickel, B., Tighe, P.J., Bihorac, A., Rashidi, P.: Deep ehr: a survey of recent advances in deep learning techniques for electronic health record (ehr) analysis. IEEE J Biomed Health Inform **22**(5), 1589–1604 (2018)
30. Song, H., Rajan, D., Thiagarajan, J.J., Spanias, A.: Attend and diagnose: Clinical time series analysis using attention models. In: AAAI (2018)
31. Song, L., Cheong, C.W., Yin, K., Cheung, W.K., Cm, B.: Medical concept embedding with multiple ontological representations. In: IJCAI. pp. 4613–4619 (2019)
32. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: NeurIPS. pp. 5998–6008 (2017)
33. Wang, S., Li, X., Chang, X., Yao, L., Sheng, Q.Z., Long, G.: Learning multiple diagnosis codes for icu patients with local disease correlation mining. ACM Transactions on Knowledge Discovery from Data (TKDD) **11**(3), 1–21 (2017)
34. Zhang, Q., Wu, J., Zhang, P., Long, G., Zhang, C.: Salient subsequence learning for time series clustering. IEEE transactions on pattern analysis and machine intelligence **41**(9), 2193–2207 (2018)