Deep Learning for Healthcare

Memory networks

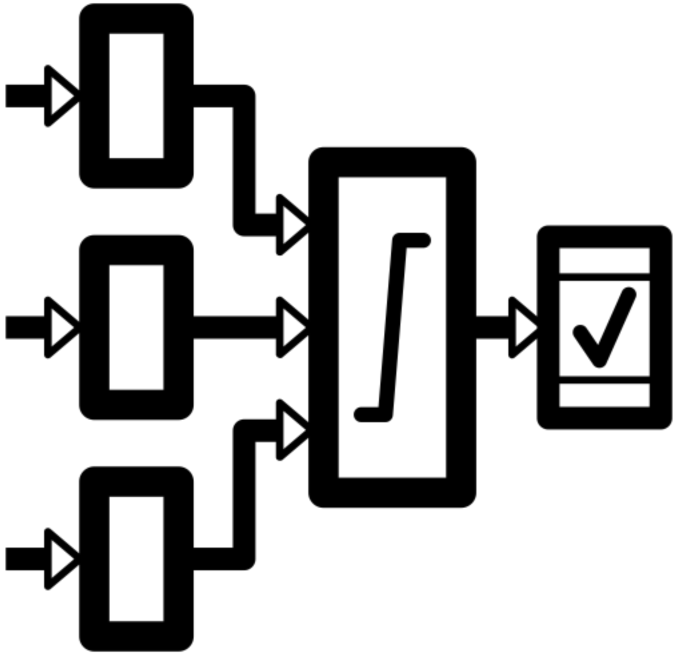
ILLINOIS

Outline

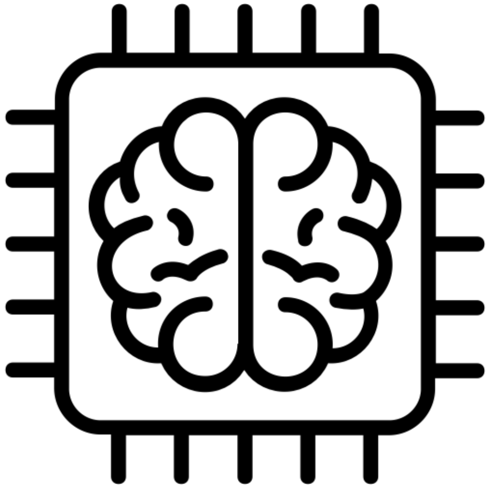
- Memory networks
  - Original
  - End to end
- Self attention
  - Transformer
  - BERT
- Case studies
  - Doctor2Vec
  - Medication recommendation
    - GAMENET
    - Pretraining of graph augmented transformer

ILLINOIS

Original memory network



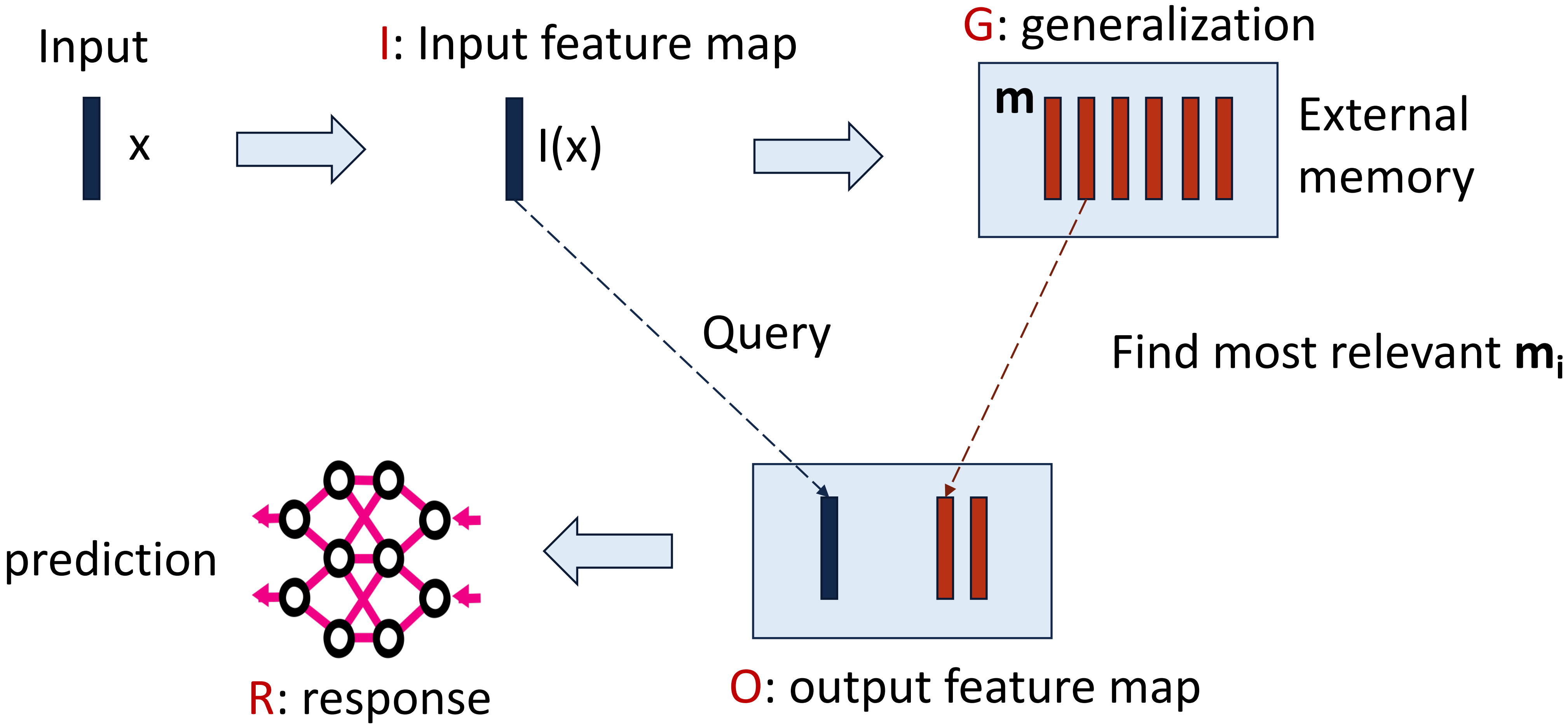
Deep neural network



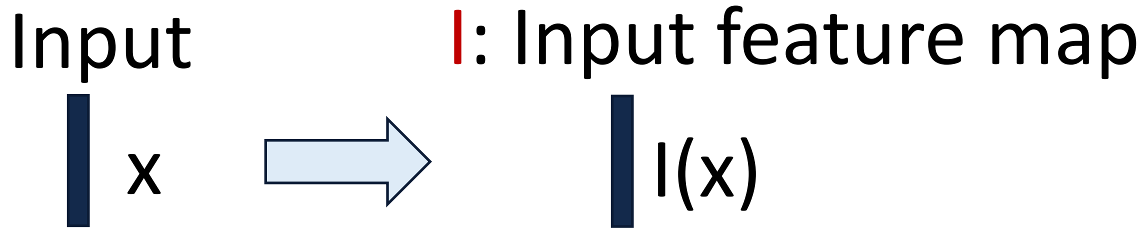
Memory components

Weston, Jason, Sumit Chopra, and Antoine Bordes. “Memory Networks.” ICLR 2015

Overview of memory networks

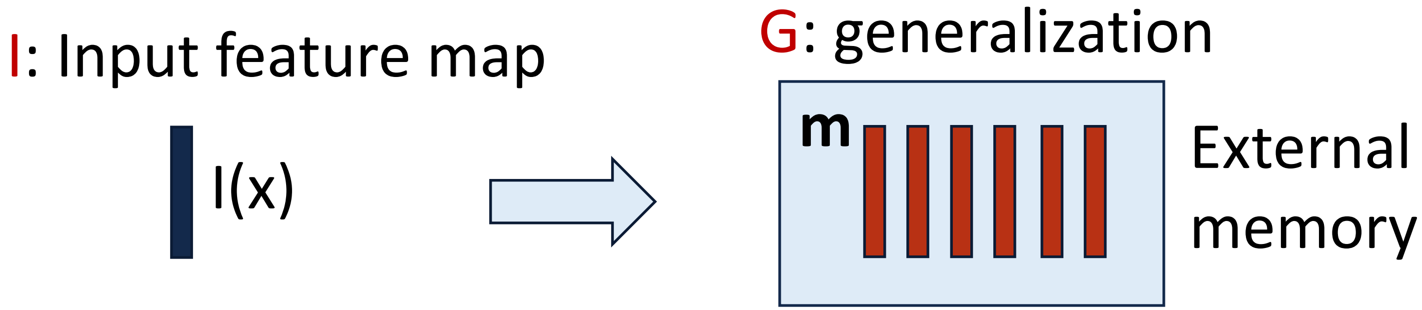


I: Input feature map



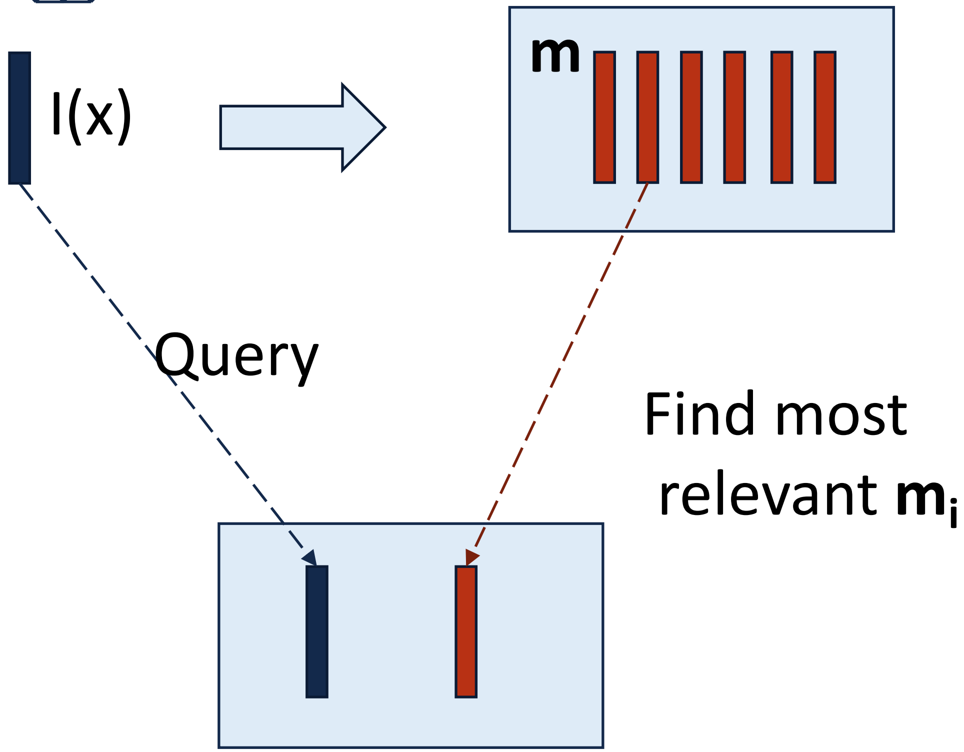
- Convert input x to an internal feature representation I(x)
- Options:
  - Simple bad of words – one hot encoding
  - RNN if the inputs are sequences

G: generalization



- Updates old memories given new input *x*
- $\mathbf{m}_i = G(\mathbf{m}_i, I(x), \mathbf{m})$ for all i.
- Simplest version is to store *I(x)* in a slot of memory
  - $\mathbf{m}_{H(x)} = I(x)$ *where H(x) is the hash function for finding the memory location to store.*

O: output feature map



O: output feature map

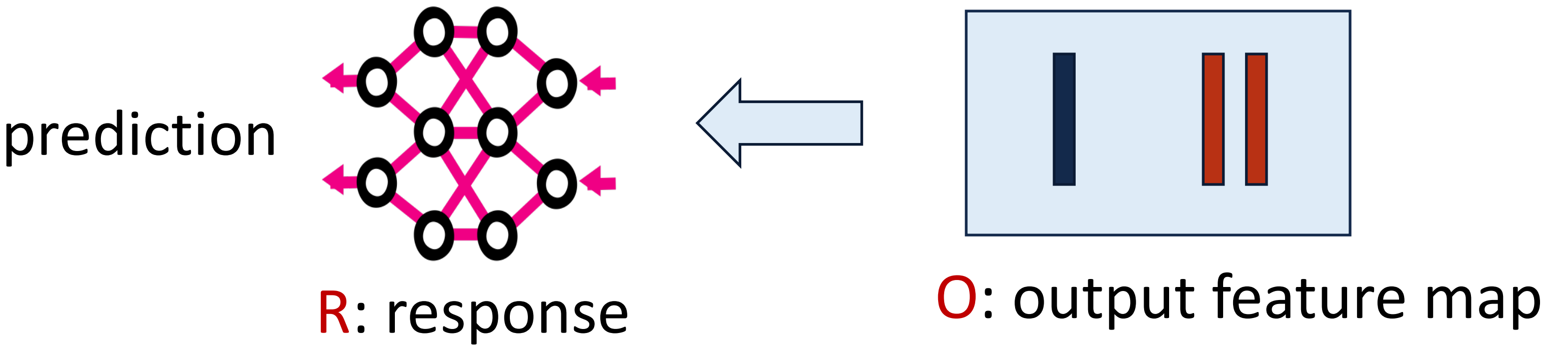
- Compute output feature $o = O(I(x), \mathbf{m})$
- *E.g.*, find the most relevant memory $\mathbf{m_i}$

- For $k = 1$, the most relevant memory is:
- $o_1 = O_1(\mathbf{I(x)}, \mathbf{m}) = \text{argmax}\ (s_O(\mathbf{I(x)}, \mathbf{m_i}))$ for all i

- Final output $o = [\mathbf{I(x)}, \mathbf{m_{o1}}]$

ILLINOIS

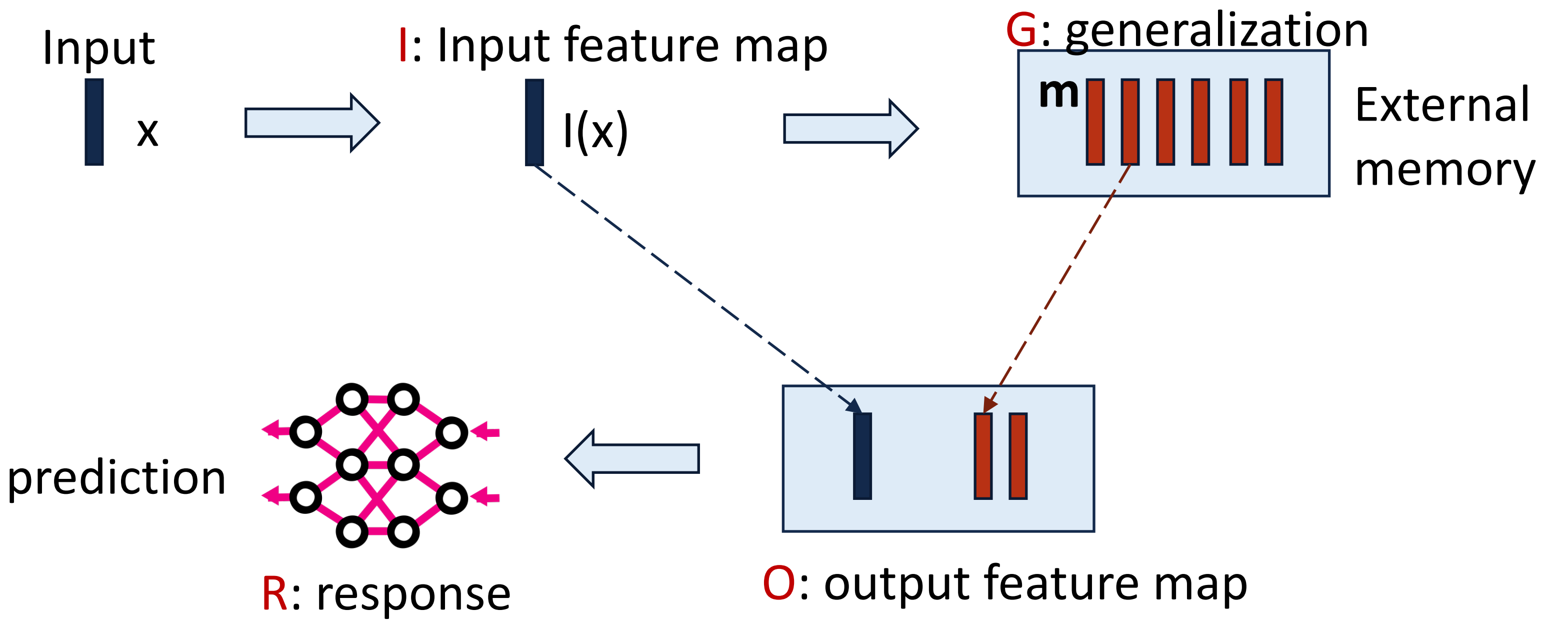
R: response



- Map output to the final response r = R(o)
- E.g,
  - Softmax for classiciation
  - RNN for sequence generation

Summary of memory network

- It bring a memory component into a deep neural network
- Limitation: not end-to-end trained because of argmax op for finding the optimal memory slot

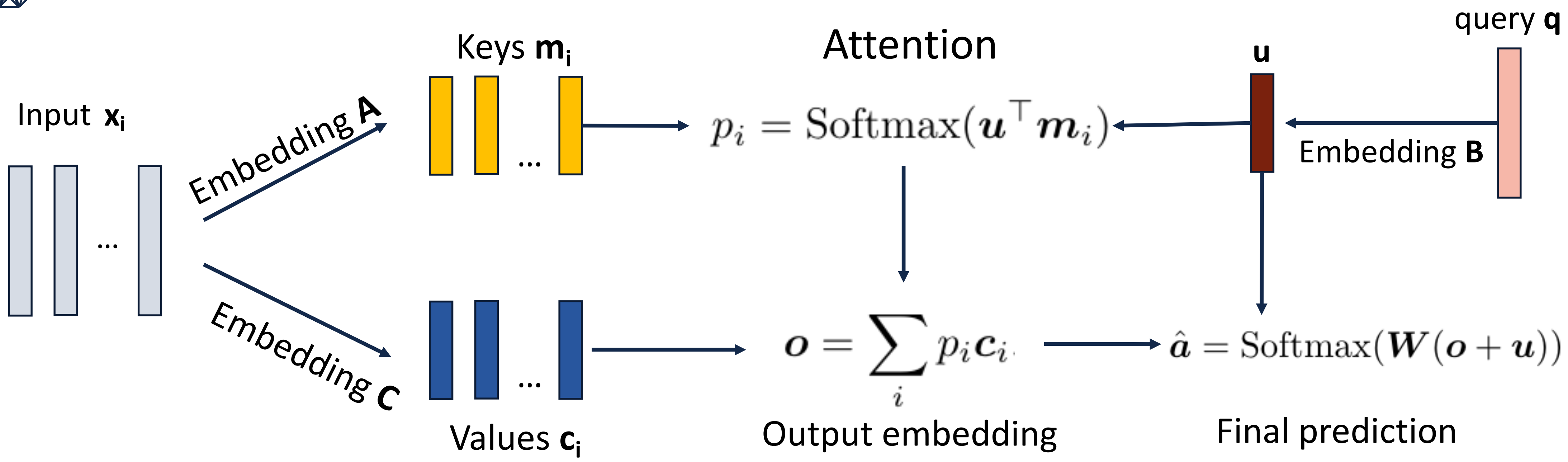


End-to-end memory network

- **Motivation**: Original memory network cannot be trained end-to-end
- **Insight**: replace the argmax in the original memory network with softmax with attention
- **General idea**
  - Store input $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ in the memory component
  - Compute attention weights between a query $\mathbf{q}$ and all $\mathbf{x}_i$

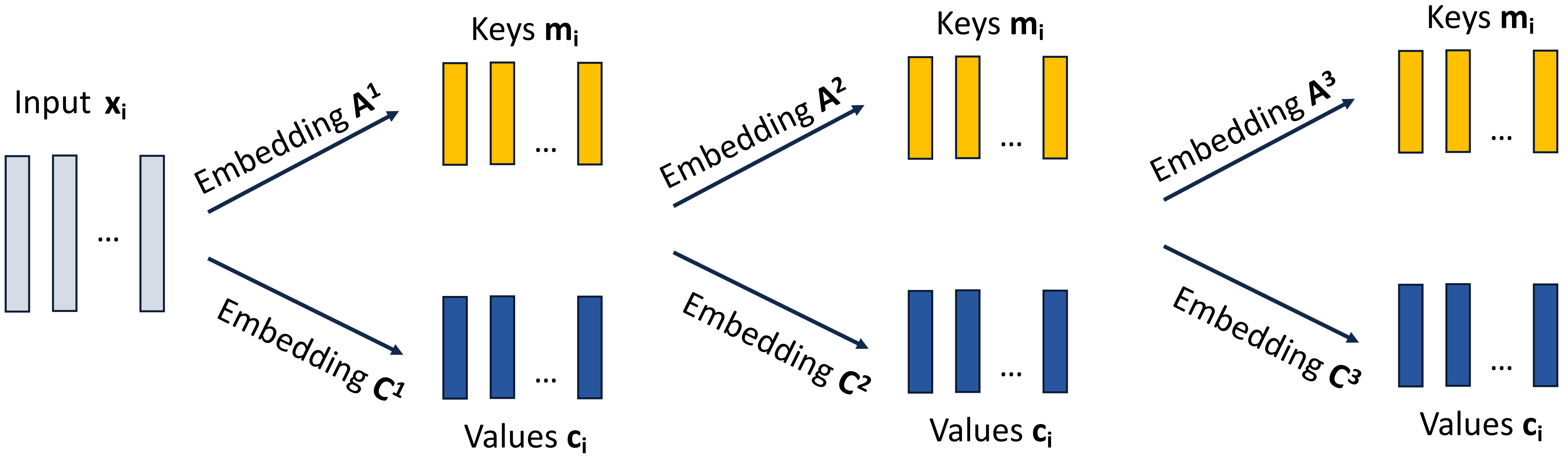
Sukhbaatar, Sainbayar, Arthur Szlam, Jason Weston, and Rob Fergus. 2015.
"End-To-End Memory Networks." *arXiv [cs.NE]*. arXiv. http://arxiv.org/abs/1503.08895.

End-to-end memory network



- Model parameters are
  - Embedding matrices A, B, C, W

Multi-layer end-to-end memory network



- Parameter sharing strategies:
  - Adjacent: $\mathbf{A}^{k+1} = \mathbf{C}^k$
  - Layer-wise: $\mathbf{A}^1 = \mathbf{A}^2 = \ldots = \mathbf{A}^k$, $\mathbf{C}^1 = \mathbf{C}^2 = \ldots = \mathbf{C}^k$

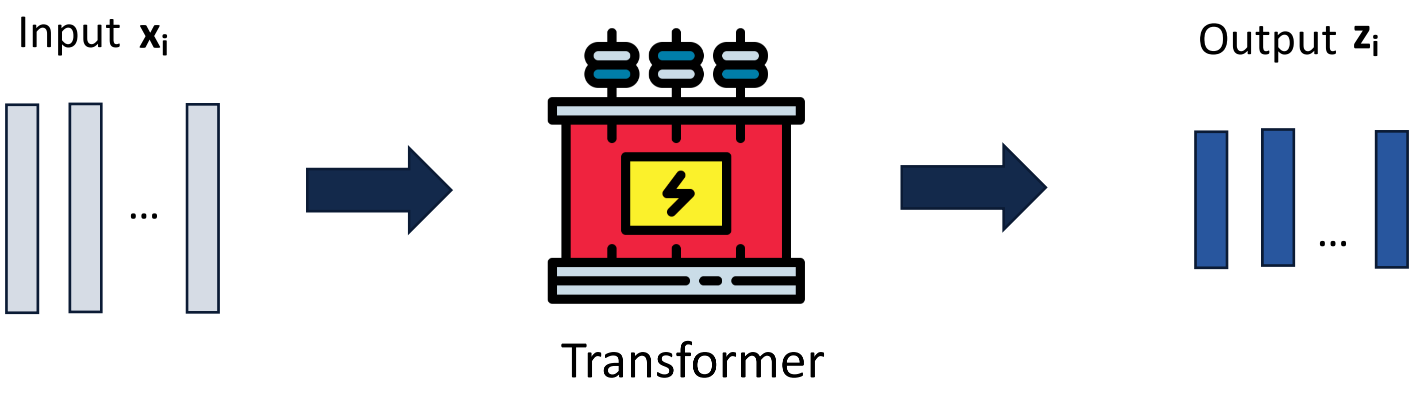
Summary: End-to-end memory network

- A variant of memory network that can be trained end-to-end

- Key ideas:
  - Use softmax attention to replace argmax operation
  - Use question answer (QA) template to model memory network
  - Allow end-to-end training

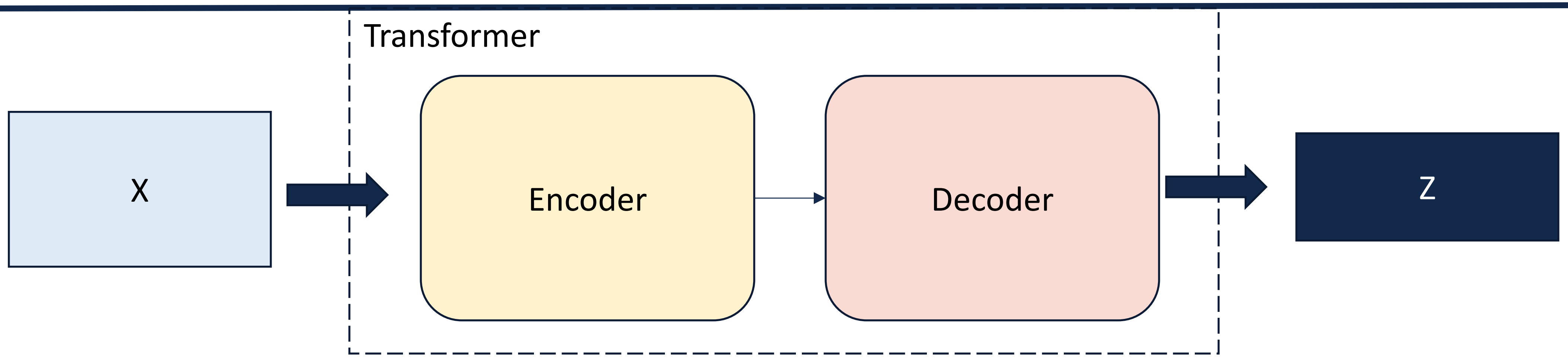
Transformer

- **Transformer** is an effective embedding method for sequential data using **self-attention** strategy.
- Question: *Can we make seq2seq model with attention train faster?*



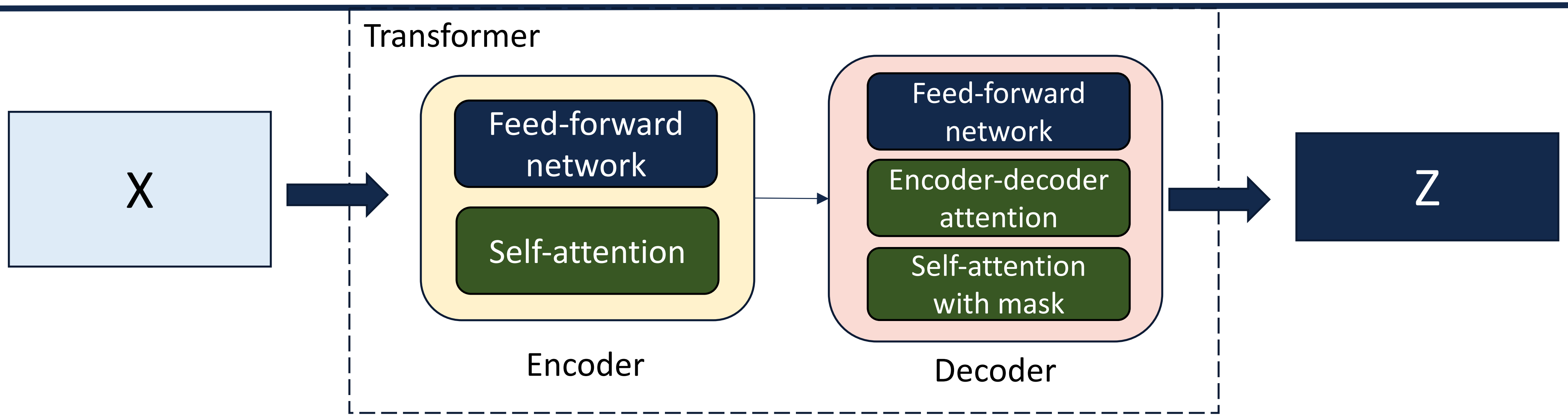
Vaswani, Ashish, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. "Attention Is All You Need." *arXiv [cs.CL]*. arXiv. http://arxiv.org/abs/1706.03762.

High-level view of transformer



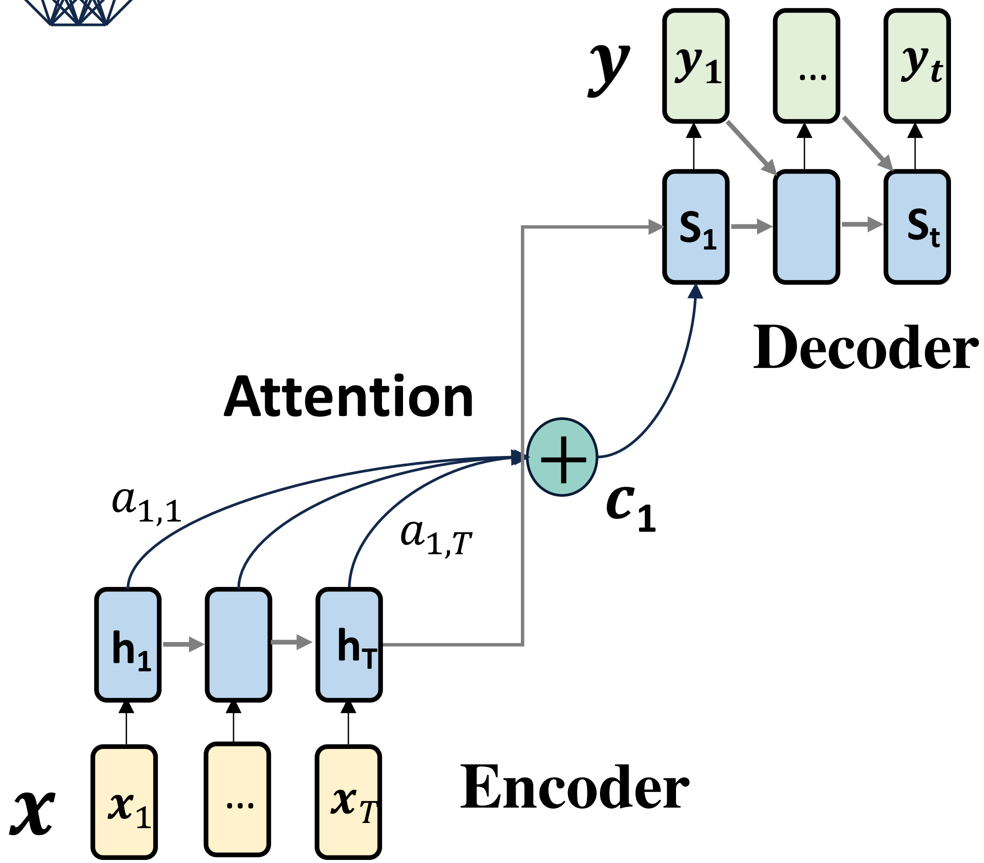
- An encoder-decoder model (sounds familiar, right?)

High-level view of transformer



- An encoder-decoder model with self-attention modules

Review: Attention on RNN Model

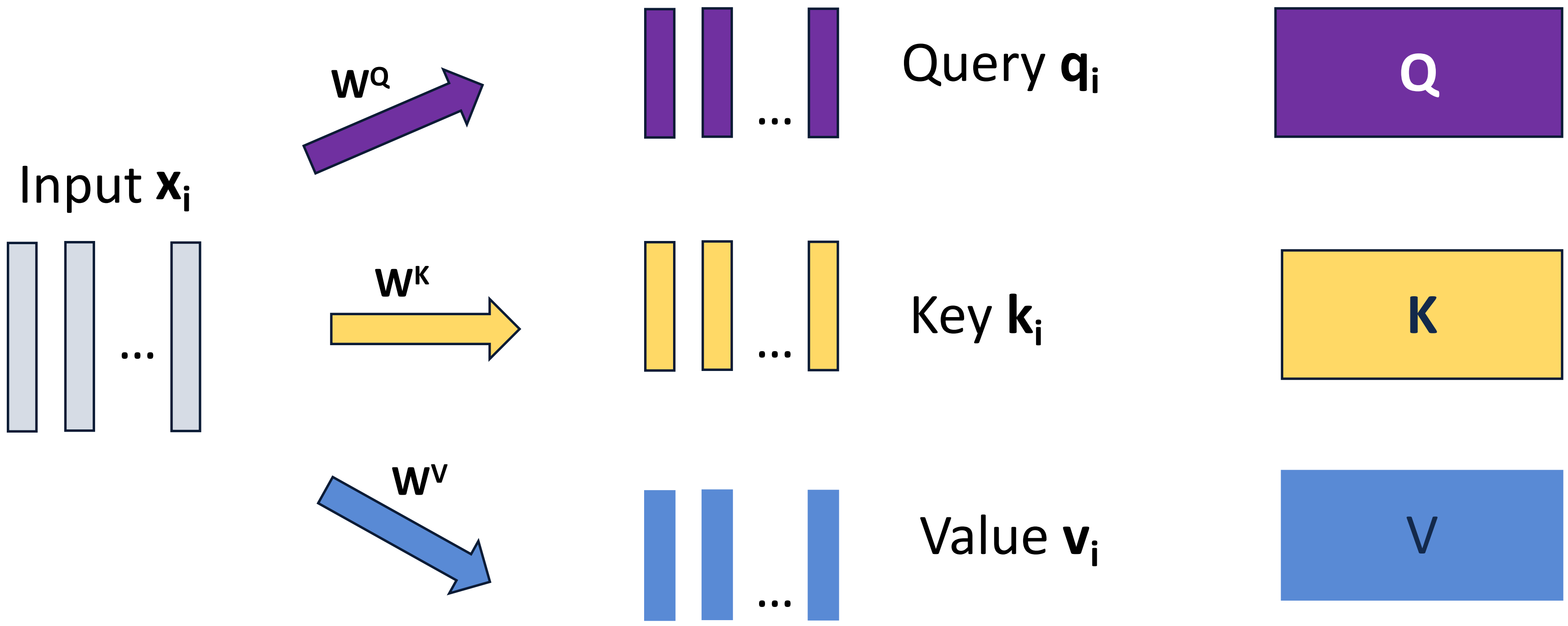


RNN cannot be trained in parallel due to its recurrence dependency

Remove RNN, and put attentions directly on input $\mathbf{x}_i$

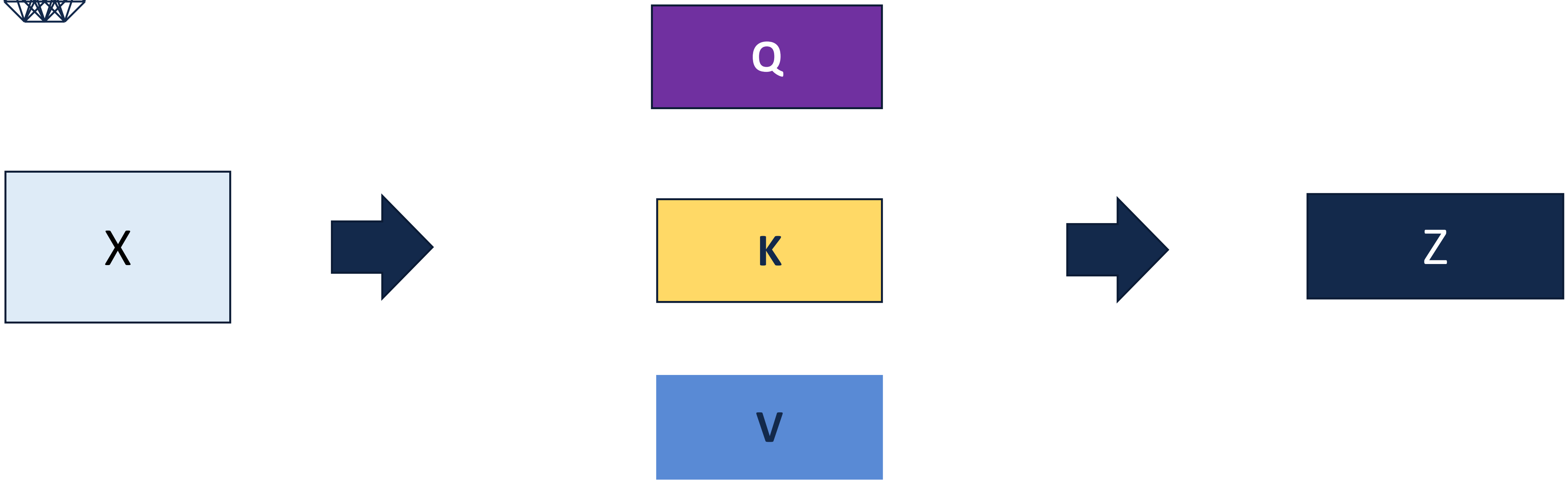
ILLINOIS

Self-attention



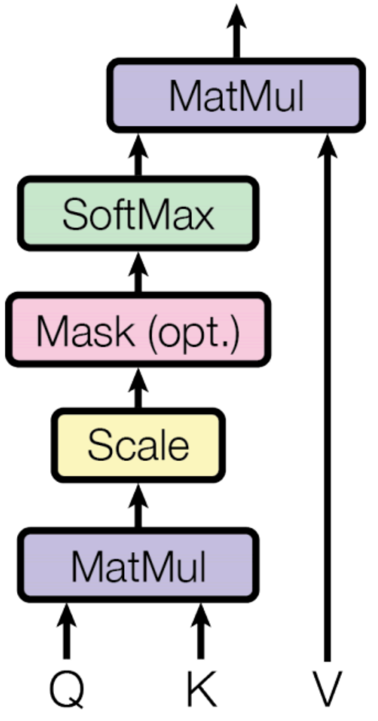
- Attention among input themselves
- Use a query retrieval strategy
- Define 3 embeddings of input x: $\mathbf{W}^Q$, $\mathbf{W}^K$, $\mathbf{W}^V$

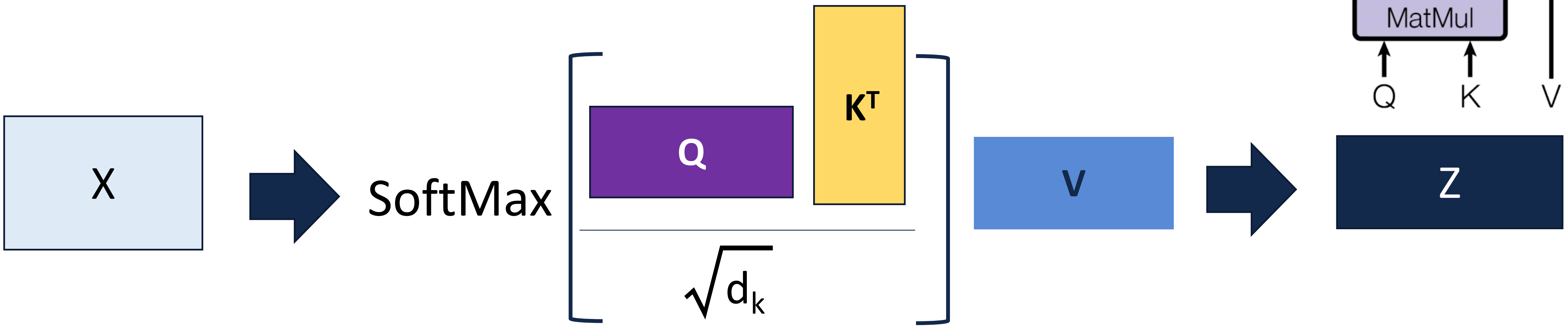
Self-attention (cont.)



- Three versions of embeddings can be learned in parallel to speed up learning

Self-attention (cont.)



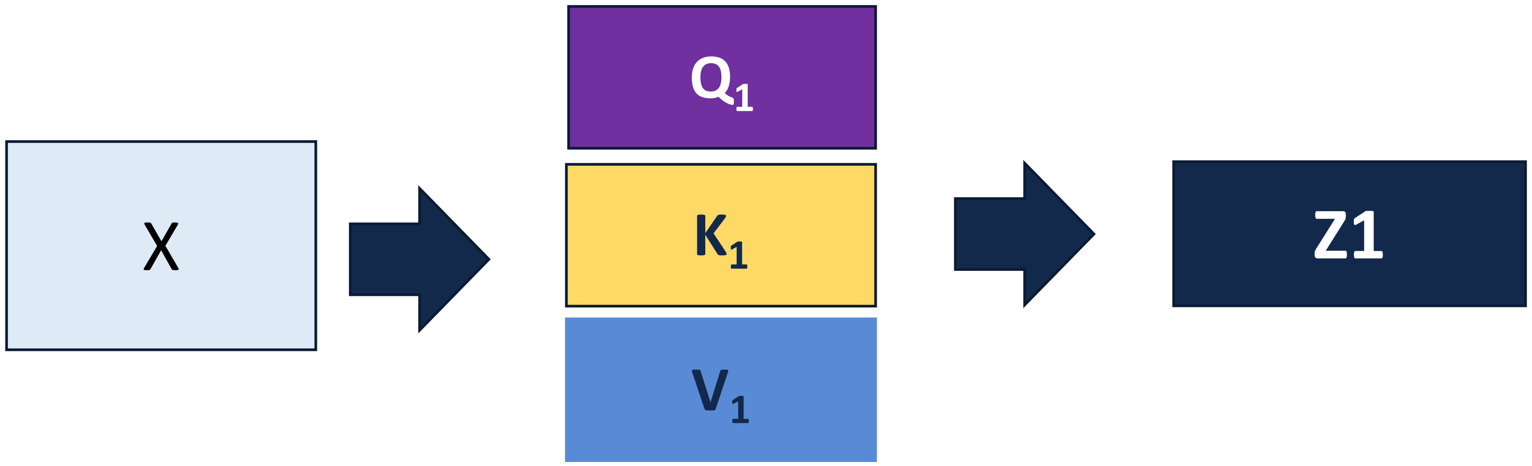


- Scaled dot product is computed as the similarity score vector with **Q K**
- Based on similarity score, retrieve/recombine value matrix **V**

Image credit: Vaswani et al. 2017. "Attention Is All You Need." http://arxiv.org/abs/1706.03762.

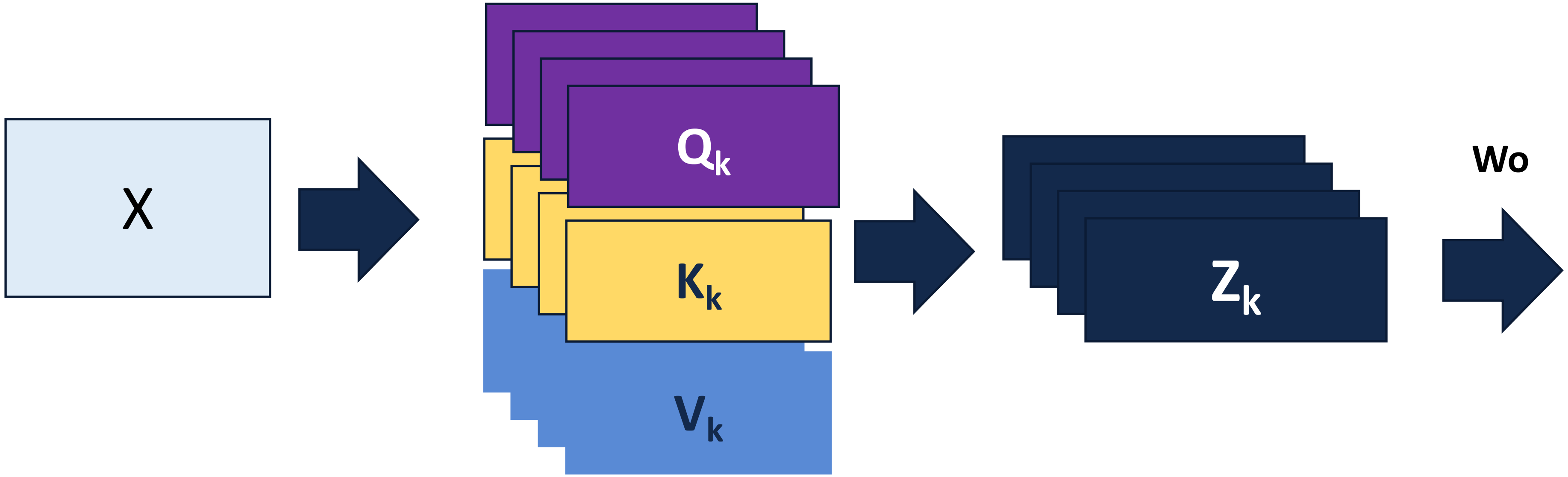
ILLINOIS

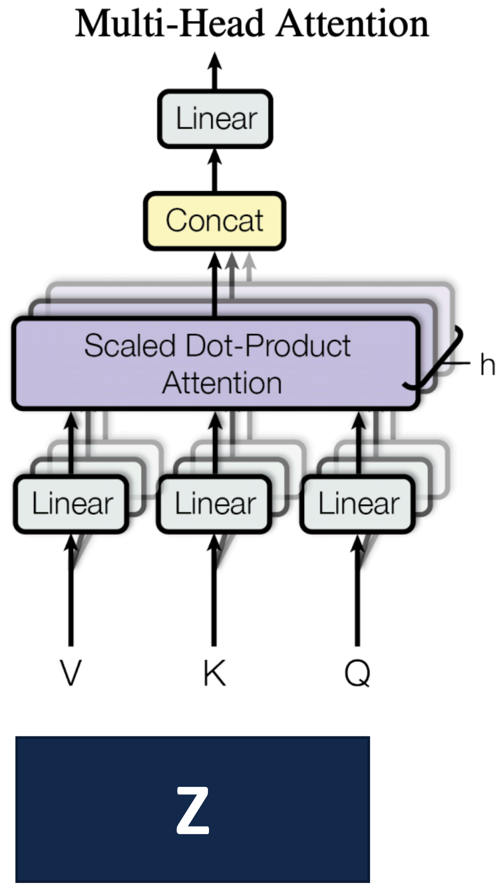
Multi-head attention



- Repeat the same attention procedures multiple times with different initialization

Multi-head attention

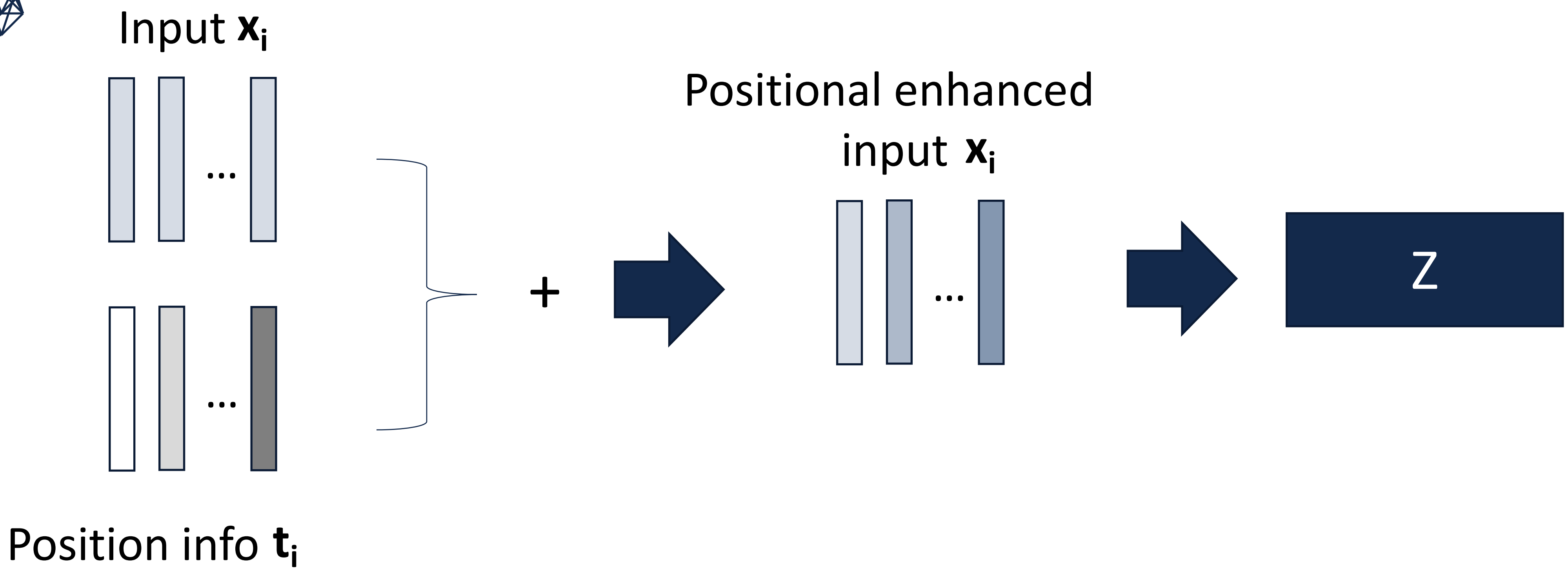




- Repeat the same attention procedures multiple times with different initialization
- Recombine at the end Z = **[Z1, Z2, …, Zk] Wo**

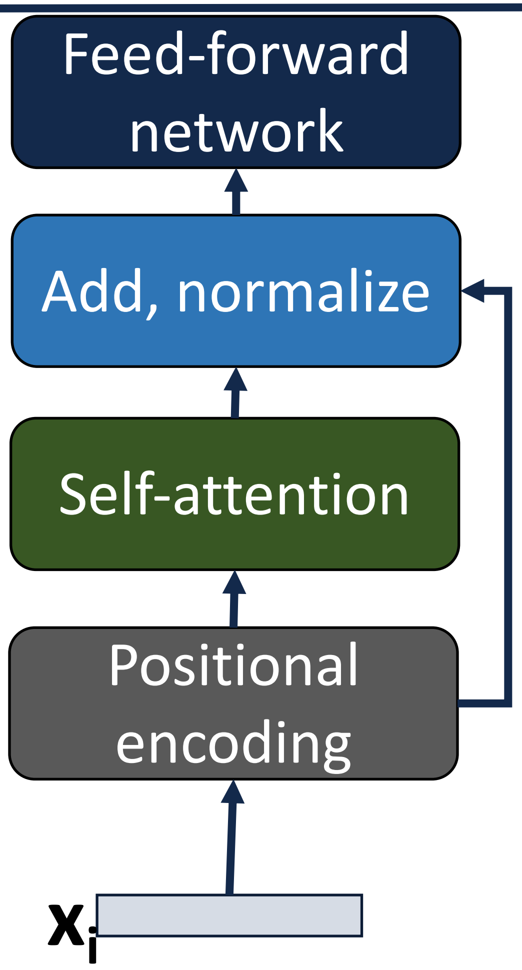
Image credit: Vaswani et al. 2017. "Attention Is All You Need." http://arxiv.org/abs/1706.03762.

Other idea: Positional encoding



- For sequential data, we can add some position specific information to the input

Other idea: Residual connection



- Add the input to self-attention output
- Use layer normalization
  - subtract mean and rescale by standard deviation across all neurons

$$\mathbf{h}^t = f\left[\frac{\mathbf{g}}{\sigma^t} \odot \left(\mathbf{a}^t - \mu^t\right) + \mathbf{b}\right]$$

$$\mu^t = \frac{1}{H}\sum_{i=1}^{H} a_i^t \qquad \sigma^t = \sqrt{\frac{1}{H}\sum_{i=1}^{H}\left(a_i^t - \mu^t\right)^2}$$

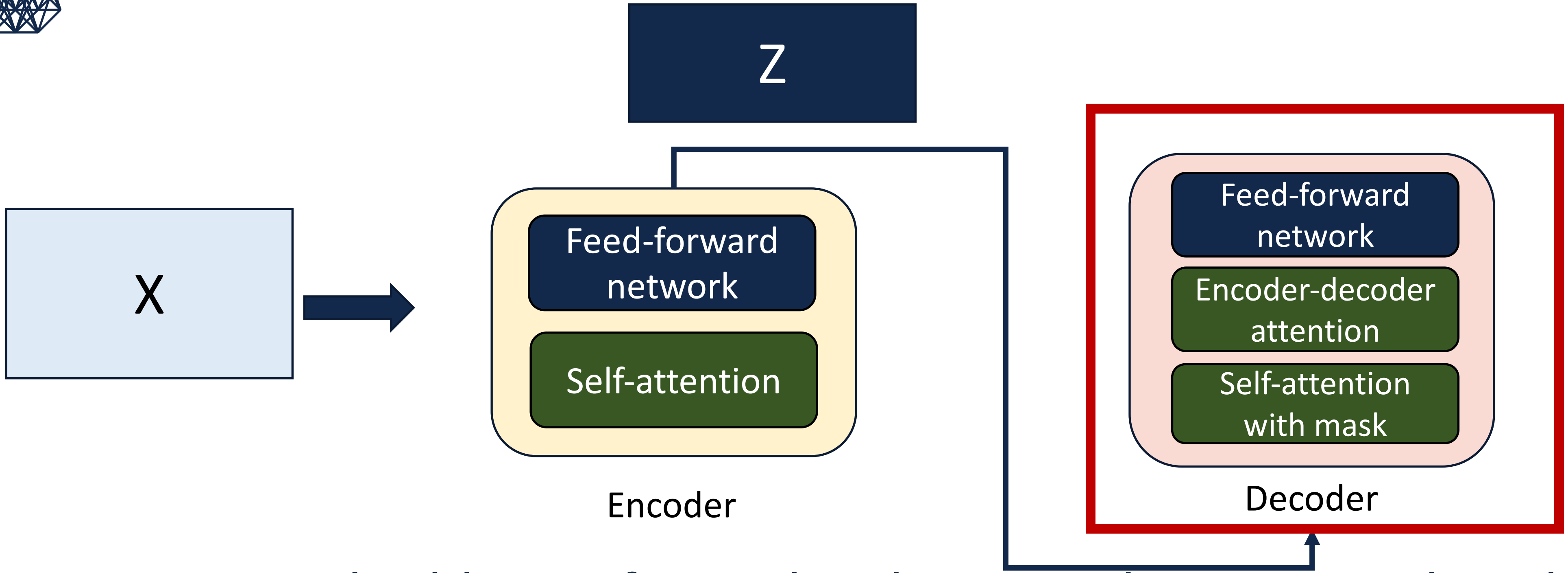
Mean

Standard deviation

Ba, Jimmy Lei, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. “Layer Normalization.” *arXiv [stat.ML]*. arXiv. http://arxiv.org/abs/1607.06450.

ILLINOIS

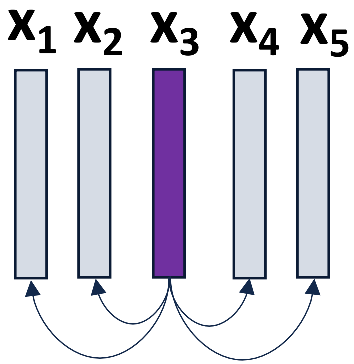
Decoder of transformer



- Output embeddings of encoders become the input to decoder
- Decoder components: variant of self-attention:
  - Self-attention with future mask
  - Encoder-decoder attention

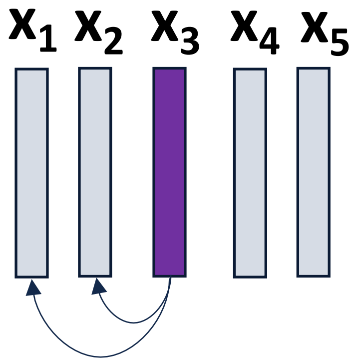
Self-attention with future mask

Standard self-attention

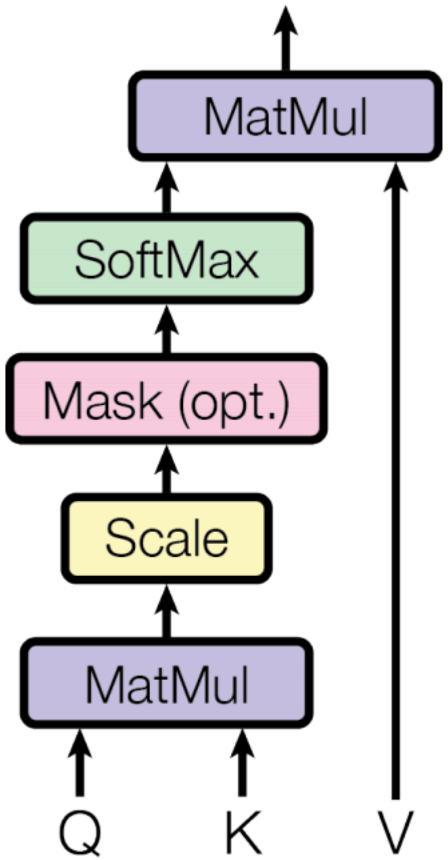


Attention of $\mathbf{x_3}$ to everyone

Attention with future mask



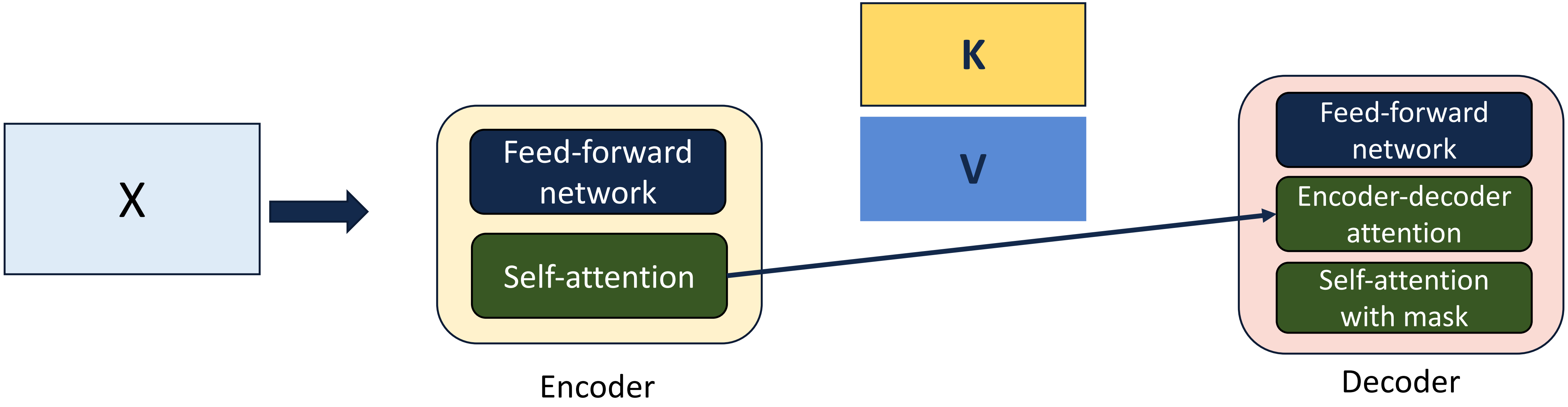
Attention of $\mathbf{x_3}$ only
on previous timestamps $\mathbf{x_1}$, $\mathbf{x_2}$



- Attention mask will be put –inf on the future timestamps

Image credit: Vaswani et al. 2017. "Attention Is All You Need." http://arxiv.org/abs/1706.03762.

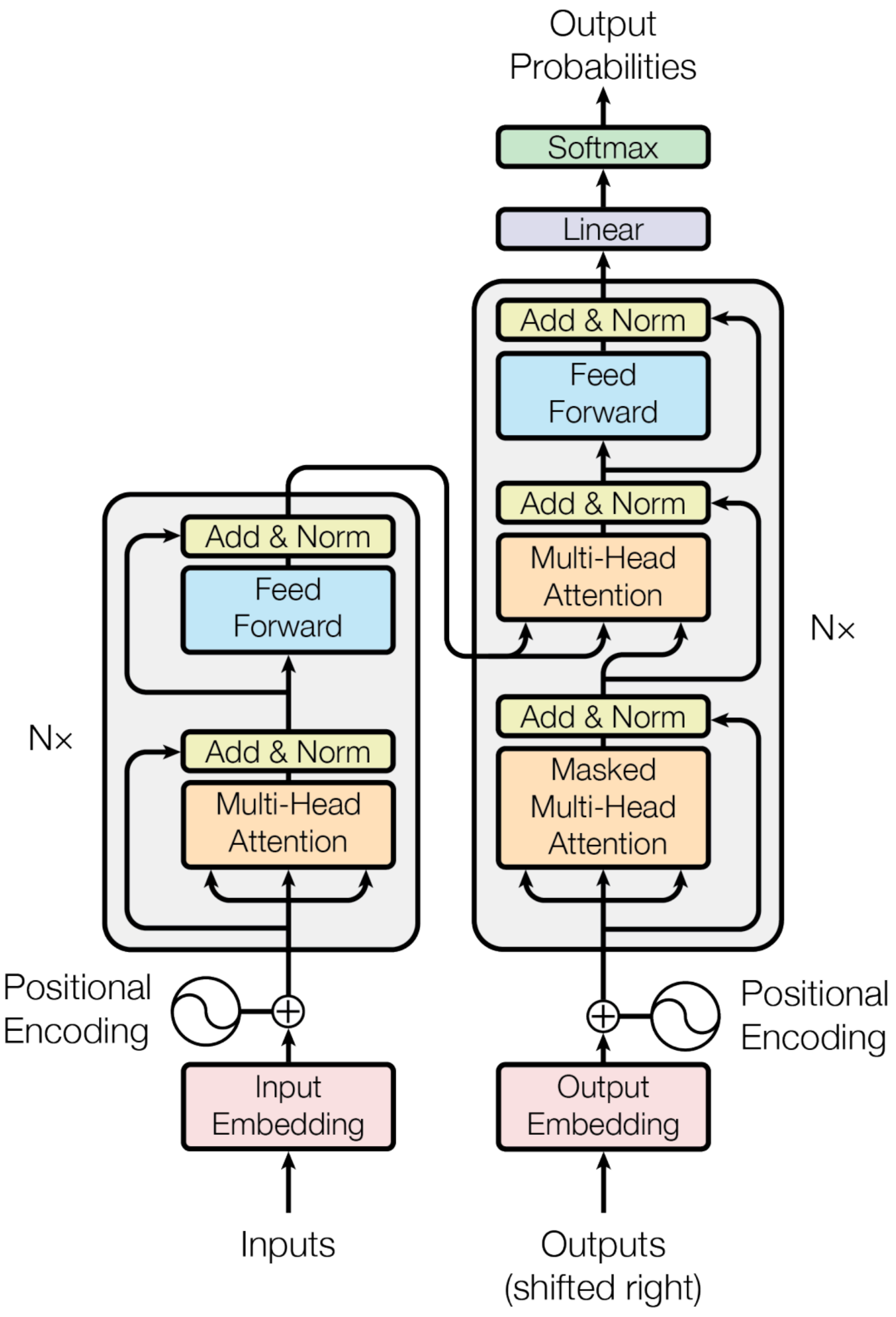
Encoder-decoder attention



- Keys K and Values V from the encoder are reused in encoder-decoder attention

Summary of Transformer

- Key idea is self-attention
  - Remove sequential dependency (No RNN module)
  - Maximize parallelism (attention with key-value pairs)
  - Takes 3 embedding matrices **K, V, Q**, which are all transformation from input **X**.
- Other important ideas:
  - Multi-head attention
  - Positional encoding
  - Residual connection



Vaswani et al. 2017. “Attention Is All You Need.” http://arxiv.org/abs/1706.03762.

BERT: Pre-training of Deep Bidirectional Transformers

- BERT model is an application of Transformer for NLP
- Key ideas

Context specific embedding

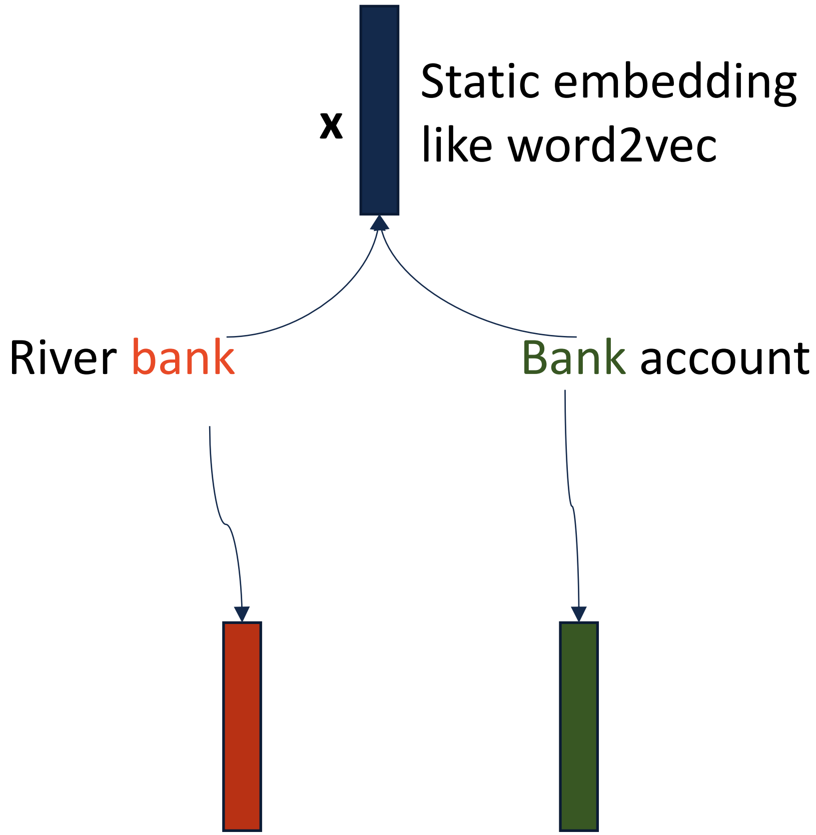
Masked language model

Devlin, Jacob, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. "BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding." *arXiv [cs.CL]*. arXiv. http://arxiv.org/abs/1810.04805.

Context specific embedding

- Previous word embeddings (e.g., Word2Vec):
  - Static embedding for each word

- BERT computes dynamic embedding for each word



Masked language model

- Existing deep learning language models:
  - RNN: Embeddings are trained from left to right
  - Bidirectional RNN: Concatenation of 2 RNNs (left to right and right to left)
- BERT masks x% input words, and try to predict what they are

Nobody ever won a chess game by resigning

Nobody ever ? a chess ? by resigning

ILLINOIS

Doctor2Vec:
Dynamic Doctor Representation Learning for Clinical Trial Recruitment

Biswal, Siddharth, Cao Xiao, Lucas M. Glass, Elizabeth Milkovits, and Jimeng Sun. 2020. “Doctor2Vec: Dynamic Doctor Representation Learning for Clinical Trial Recruitment.” AAAI

ILLINOIS

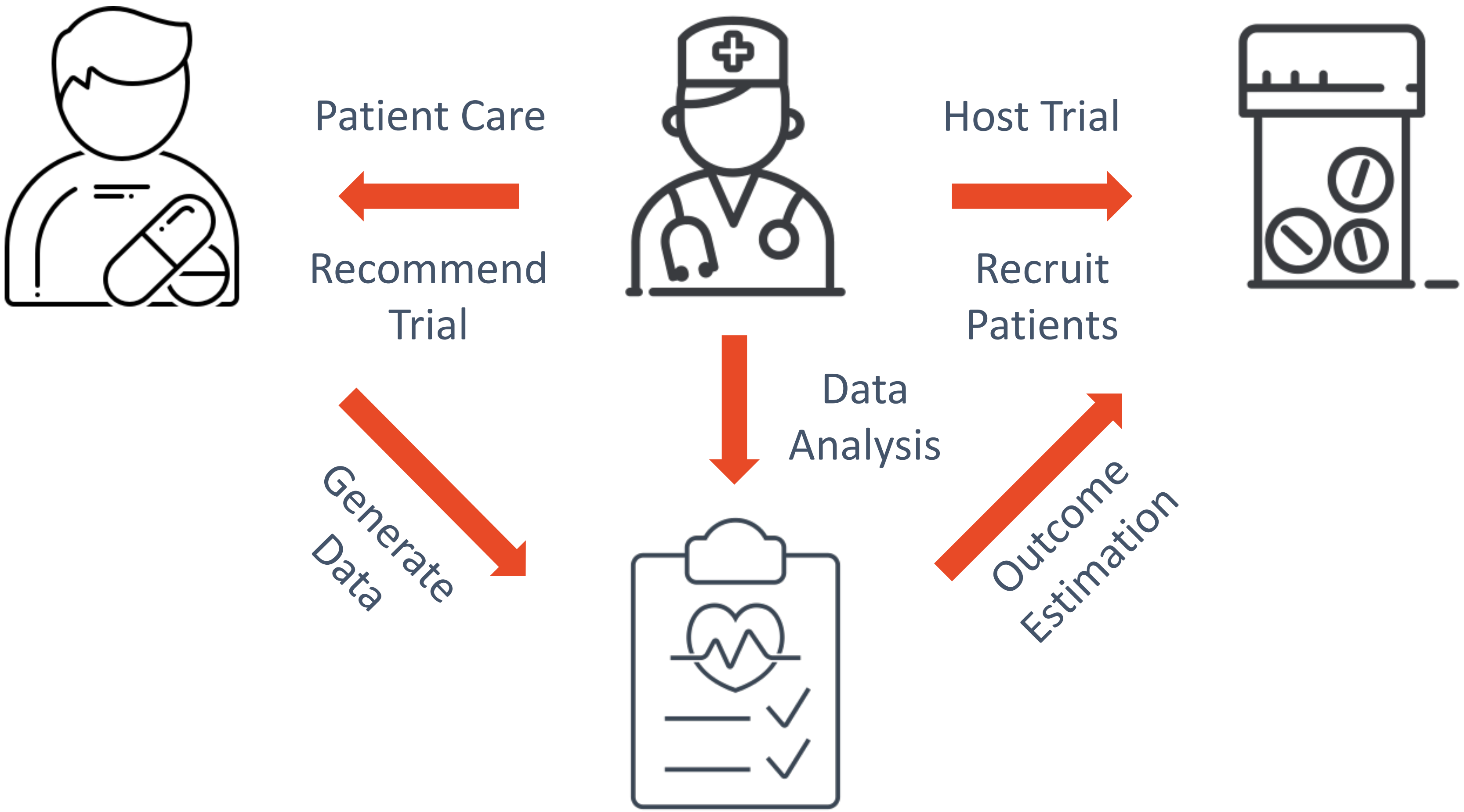
Agenda

- Background
- Doctor2Vec
- Experiments

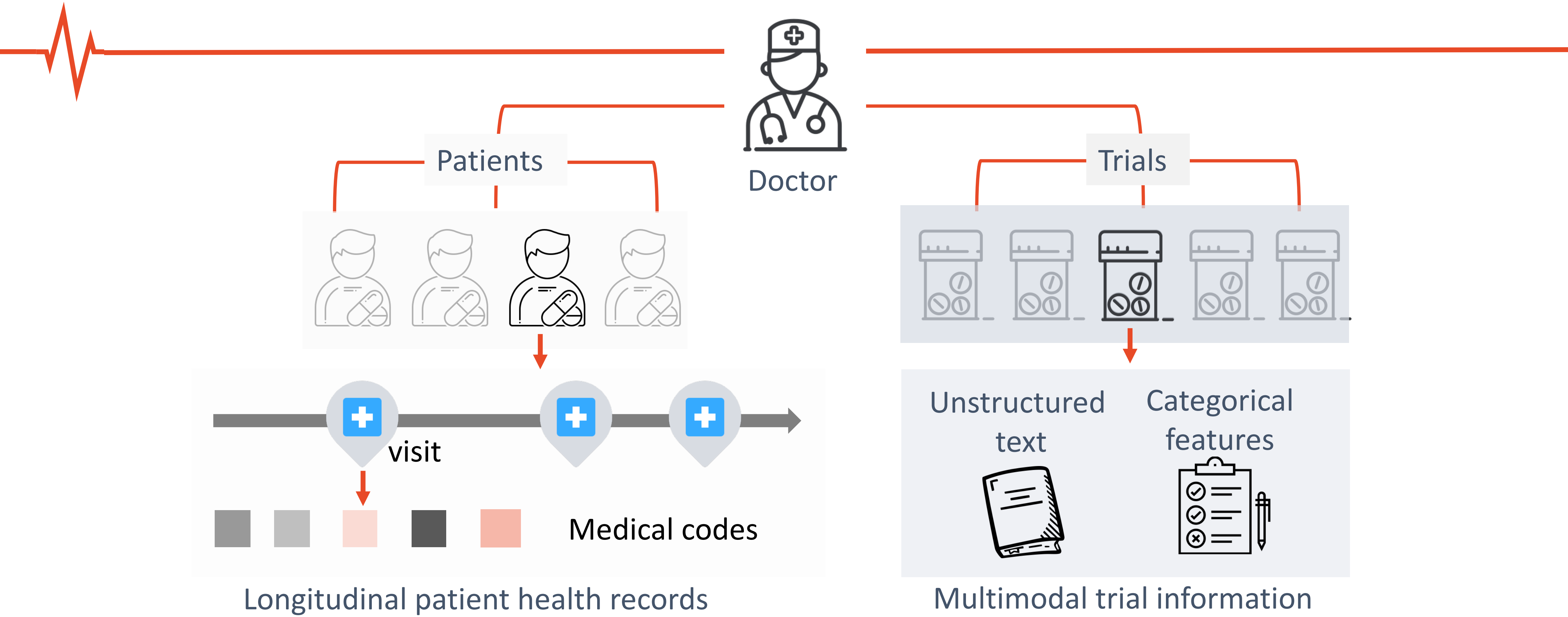
Agenda

- Background
- Doctor2Vec
- Experiments

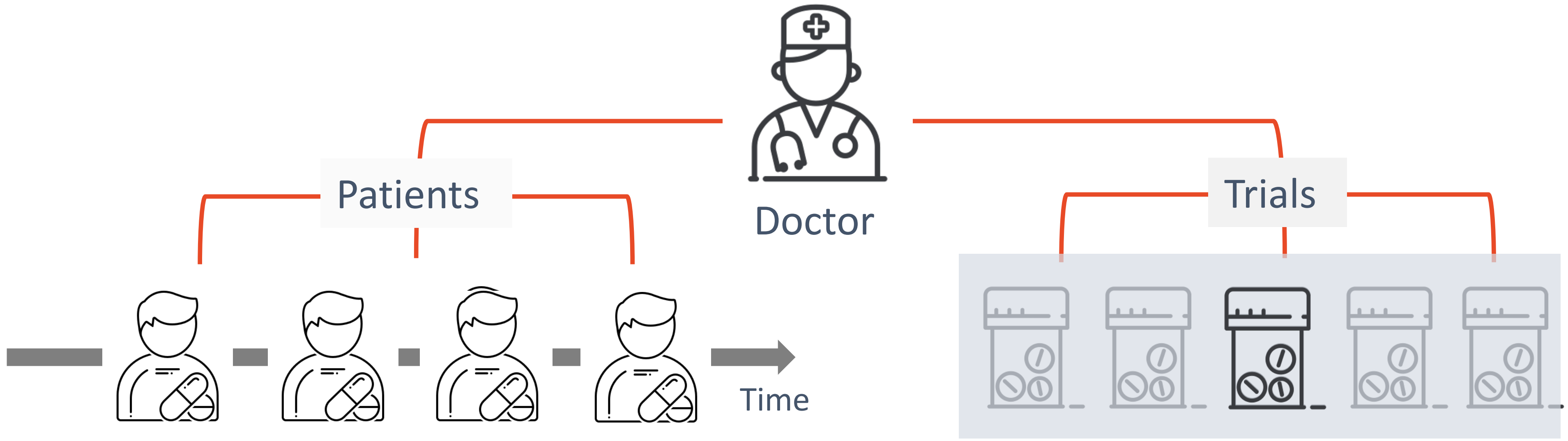
Doctors Play Pivotal Roles in Healthcare



Doctor Representation Learning



Challenges of Doctor Representation Learning



1 Existing works do not capture the time-evolving patterns of doctors experience/expertise.

2 Existing works learn a static doctor representation, rather than a dynamic one based on the corresponding trial.

Our Contribution

Challenges

1. Existing works do not capture the time-evolving patterns of doctors experience/expertise.

2. Existing works learn a static doctor representation, rather than a dynamic one based on the corresponding trial.

Solutions

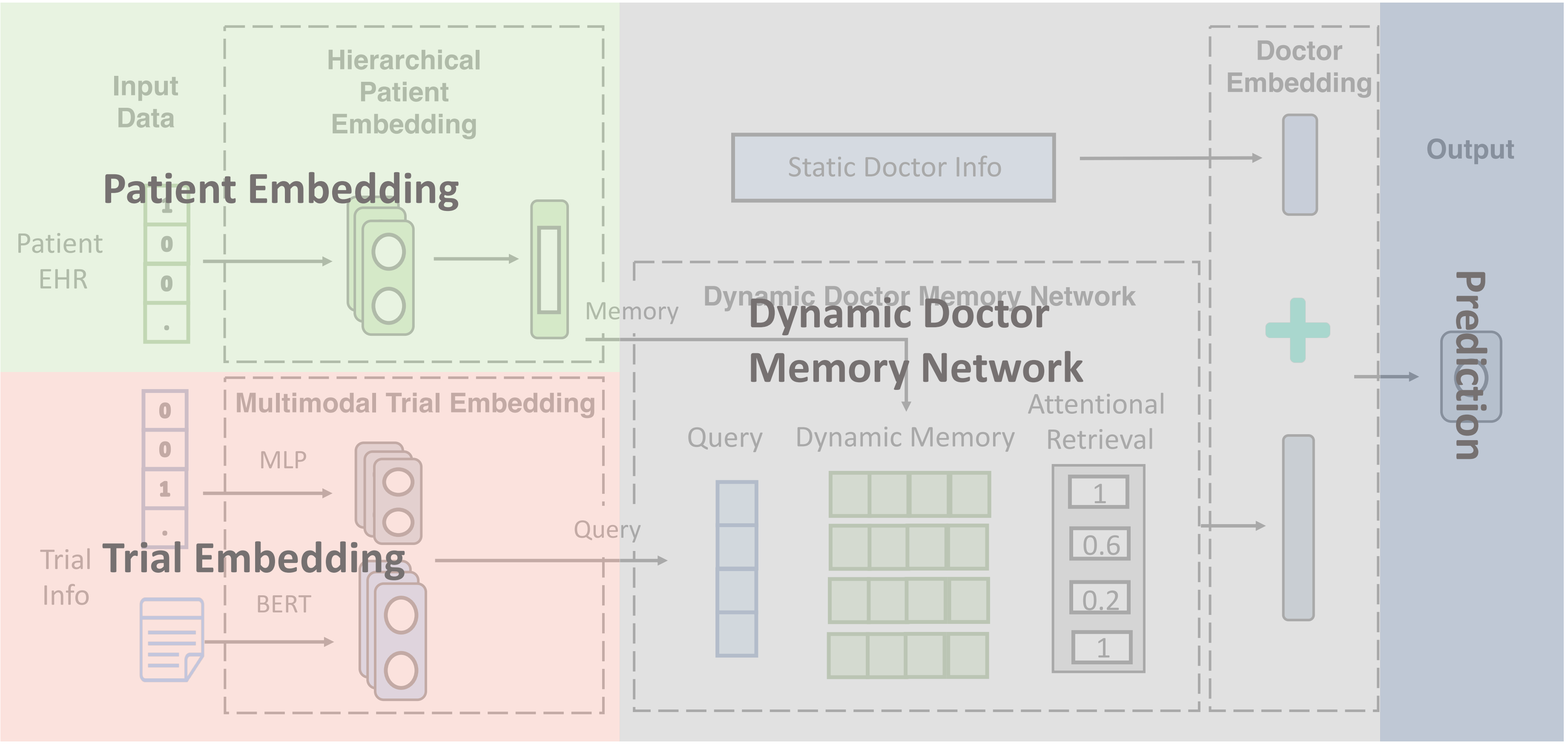
1. Patient embedding as a memory for dynamic doctor experience encoding.

2. Trial embedding as a query for improved doctor embedding.

Agenda

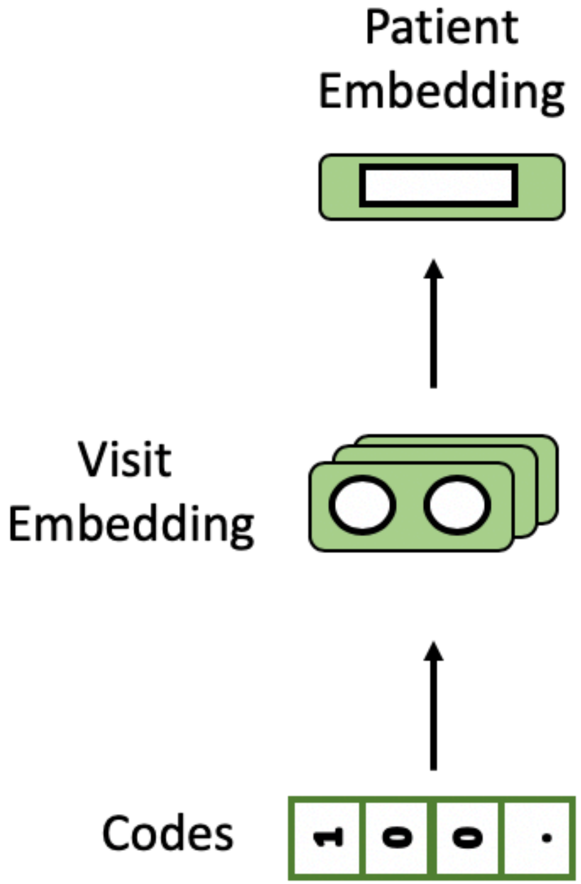
- Background
- Doctor2Vec
- Experiments

Doctor2Vec: **Overview**



ILLINOIS

Doctor2Vec: Hierarchical Patient Embedding



$$\mathbf{I}(\mathbf{k}) = \sum \alpha_{\mathbf{t}} \cdot \mathbf{h}_{\mathbf{t}}$$

Memory for memory networks

$$\mathbf{g}_1, \mathbf{g}_2, \ldots \mathbf{g}_{\mathbf{t}} = \text{bi-LSTM}(\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_{\mathbf{t}})$$

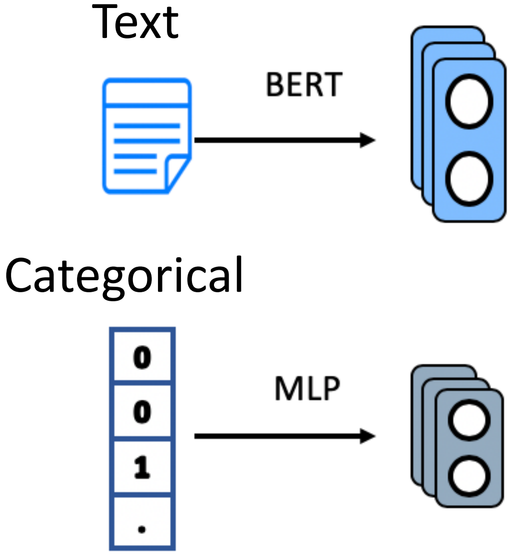
$$\mathbf{e}_{\mathbf{t}} = \mathbf{w}_{\alpha}^{\mathbf{T}} * \mathbf{g}_{\mathbf{t}} + \mathbf{b}_{\alpha}$$

$$\alpha_1, \alpha_2, ..\alpha_{\mathbf{t}} = \text{softmax}(\mathbf{e}_1, \mathbf{e}_2, ..\mathbf{e}_{\mathbf{t}})$$

$$\mathbf{h}_{\mathbf{t}}(\mathbf{k}) = \mathbf{W}_{\mathbf{emb}} * \mathbf{v}_{\mathbf{t}}(\mathbf{k})$$

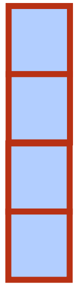
Doctor2Vec: Multimodal Trial Embedding

Pretrained BERT using MIMIC text data to extract embeddings for each word.



$$\mathbf{Q}(\mathbf{text})_{\mathbf{emb}} = \mathrm{BERT}(\mathbf{Q}(\mathbf{text})^{(l)})$$

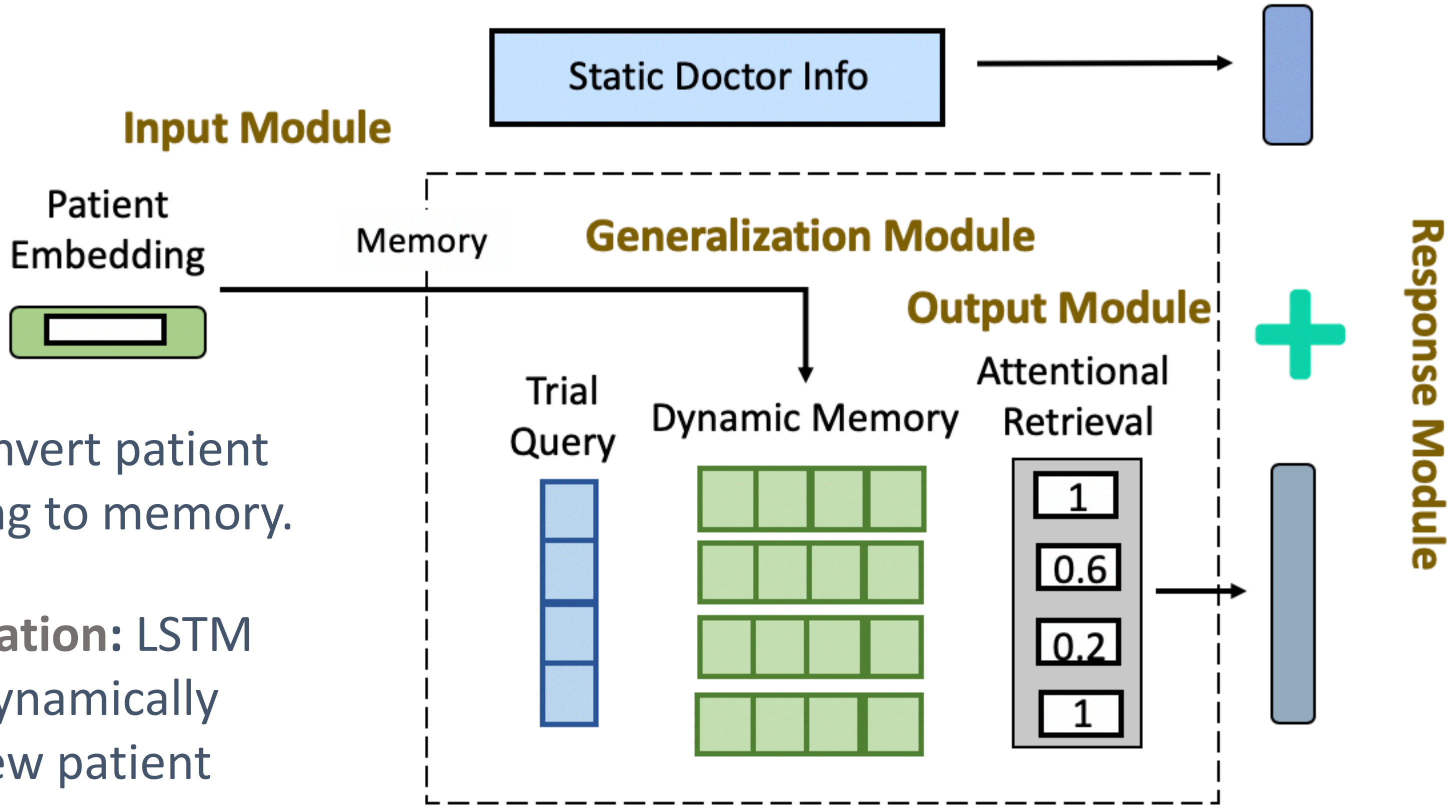
$$\mathbf{Q}(\mathbf{cat})_{\mathbf{emb}} = \mathbf{W}_{\mathbf{c}}(\mathbf{Q}(\mathbf{cat})^{(\mathbf{l})}) + \mathbf{b}_{\mathbf{c}}$$



Query for memory networks

$$\mathbf{Q}_{\mathbf{emb}}(\mathbf{l}) = (\mathbf{W}_{\mathbf{ci}}\mathbf{Q}(\mathbf{cat})_{\mathbf{emb}} + \mathbf{b}_{\mathbf{ci}}) \odot (\mathbf{W}_{\mathbf{ti}}\mathbf{Q}(\mathbf{text})_{\mathbf{emb}} + \mathbf{b}_{\mathbf{ti}})$$

Doctor2Vec: Dynamic Doctor Memory Network



Input: Convert patient embedding to memory.

Generalization: LSTM layer to dynamically update new patient memory.

Output: Attentional memory retrieval to learn doctor embedding only based on trial-specific memory.

Response: Concat. to get doctor embedding

ILLINOIS

Doctor2Vec: Prediction

Output predicted enrollment rate between 0 and 1

O

$$\mathbf{Y} = \mathrm{Softmax}([\mathbf{Doc}_{\mathbf{emb}}; \mathbf{Q}_{\mathbf{emb}}(\mathbf{l}); \mathbf{Doc}_{\mathbf{static}}])$$

Regression Task

Category	[0,0.2]	(0.2,0.4]	(0.4,0.6]	(0.6,0.8)	(0.8,1.0]
Distribution	12%	33%	37%	12%	6%

Classification Task

Agenda

- Background
- Doctor2Vec
- Experiments

ILLINOIS

Design of Experiments

Q1: Does Doctor2Vec have better performance in predicting clinical trial enrollment to support site selection?

Q2: Can Doctor2Vec embedding perform in transfer learning setting for trials across countries or across diseases?

Data

a) IQVIA trial data about trials formed during 2014 and 2019 across 28 countries.
b) clinical trial description from clinicaltrials.gov, matched with IQVIA trial data on NCT ID
c) IQVIA claims data

Table 2: Data Statistics

# of clinical trials	2609
# of doctors	25894
# of doctor-trial pair(samples)	102487
# of patients	430,239
Avg # of Dx codes per visit	4.23
Max # of Dx codes per visit	56
Avg # of Procedure codes per visit	1.23
Max # of Procedure codes per visit	18
Avg # of Med codes per visit	9.36

Metrics

Classification Task	The precision recall area under curve (PR-AUC) is the area under the PR curve. A good metric for data imbalanced setting. The higher the better.
Regression Task	The coefficient of determination (R-squared) is the square of the correlation between predicted scores and actual scores. The higher the better.

Baseline

- **Median Enrollment (Median):** considers the median enrollment rate for each therapeutic area as estimated rate for all trials in that area.
- **Logistic Regression (LR):** Combine all features and then apply LR.
- **Random Forest (RF):** Combine all features and then apply RF.
- **AdaBoost:** Combine all features and then apply AdaBoost.
- **Multi-layer Perceptron (MLP):** Convert codes to count vectors, convert categorical information of clinical trials to multi-hot vectors and obtain TF-IDF features from text information of clinical trials. Then apply MLP.
- **Long Short-Term Memory Networks (LSTM):** process all temporal data using LSTM and then concatenate with other features.
- **DeepMatch:** Features for the doctors are obtained from the top 50 most frequent medical codes and passed through an MLP layer to obtain an embedding vector.

Results

	PR-AUC	R^2 Score
Median	0.571 ± 0.014	0.54 ± 0.072
LR	0.672 ± 0.041	0.314 ± 0.082
RF	0.731 ± 0.034	0.618 ± 0.034
AdaBoost	0.747 ± 0.002	0.684 ± 0.146
MLP	0.761 ± 0.019	0.762 ± 0.049
LSTM	0.792 ± 0.034	0.780 ± 0.621
DeepMatch	0.735 ± 0.068	0.821 ± 0.073
Doctor2Vec	$\mathbf{0.861 \pm 0.021}$	$\mathbf{0.841 \pm 0.072}$

Doctor2Vec has 8.7% relative improvement in PR-AUC over the best baseline LSTM.

- ✓ LSTM > MLP > Other non temporal models, due to better model the temporal information.
- ✓ DeepMatch models achieved much lower PR-AUC since the model leverages the 50 most frequent codes for embedding, thus miss important but non-frequent information.
- ✓ DeepMatch in the regression settings tends to perform better than MLP and LSTM, due to the skewed data distribution.

Transfer Learning Results

Transfer to a less populated or newly explored country

Train model on 1443 clinical trials in the United states during the time 2014-2019 and test on 47 clinical trials in South Africa during the time 2014-2019.

	PR-AUC	R^2 Score
Median	0.524 ± 0.032	0.420 ± 0.039
LR	0.601 ± 0.023	0.279 ± 0.014
RF	0.661 ± 0.038	0.552 ± 0.048
AdaBoost	0.672 ± 0.01	0.581 ± 0.039
LSTM	0.758 ± 0.013	0.721 ± 0.025
DeepMatch	0.703 ± 0.087	0.756 ± 0.031
Doctor2Vec	$\mathbf{0.862 \pm 0.003}$	$\mathbf{0.817 \pm 0.025}$

Doctor2Vec *achieved 13.7% better PR-AUC then LSTM and 8.1% R2 then DeepMatch.*

Transfer to rare or low prevalence diseases

Test on 38 clinical trials for drugs about idiopathic pulmonary fibrosis (IPF) and inflammatory bowel disease(IBD. Train on 2569 clinical trials from the rest of the available diseases.

	PR-AUC	R^2 Score
Median	0.413 ± 0.013	0.387 ± 0.001
LR	0.521 ± 0.021	0.225 ± 0.028
RF	0.610 ± 0.019	0.517 ± 0.032
AdaBoost	0.623 ± 0.002	0.548 ± 0.046
LSTM	0.725 ± 0.002	0.623 ± 0.038
DeepMatch	0.638 ± 0.021	0.678 ± 0.049
Doctor2Vec	$\mathbf{0.784 \pm 0.032}$	$\mathbf{0.716 \pm 0.014}$

Doctor2Vec *achieved 8.1% better PR-AUC then LSTM and 5.2% R2 then DeepMatch.*

Case Study

Ground Truth

Trial: Phase I trial for Gemcitabine plus Cisplatin (a combination of cancer therapy)

Doctor: a doctor in USA who has worked in internal medicine during past 3 years. The doctor has a broader coverage of diseases.

Enrollment Rate: 0.71

Prediction

Best Baseline LSTM: 0.57
Consider these diseases homogeneously when measuring the match between the doctor and the trial.

Doctor2Vec: 0.69
Focus more on the patients who had cancer diagnosis instead of all patients, thus is more accurate.

Summary: Doctor2Vec

- Doctor representation using memory network to encode both dynamic information and static information
- Strong performance in clinical trial recruitment applications

GAMENet: Graph Augmented MEmory Networks for Recommending Medication Combination

Shang, Junyuan, Cao Xiao, Tengfei Ma, Hongyan Li, and Jimeng Sun. 2018. “GAMENet: Graph Augmented MEmory Networks for Recommending Medication Combination.” *AAAI*

Medication Errors & Adverse Drug-drug Interactions

10 percent of all U.S. deaths are now due to medical error

3rd highest cause of death in the U.S. is medical error

Adverse drug-drug interactions affects **15 percent** U.S. population.

Cost more than **$177 billion** per year in disease management

https://www.hopkinsmedicine.org/news/media/releases/study_suggests_medical_errors_now_third_leading_cause_of_death_in_the_us

Task: Recommend Medication Combinations

Visit 1

Dx, Proc
Rx

……

Visit t-1

Dx, Proc
Rx

Patient history

+

Visit t

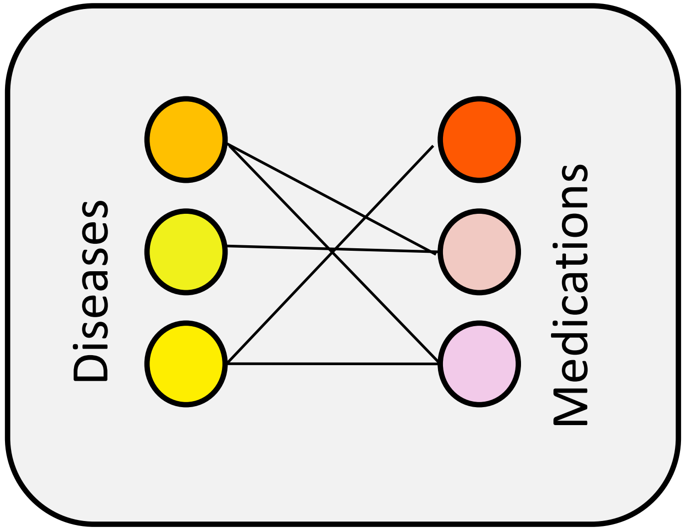
Dx, Proc

Diagnosis + Procedure
from current visit

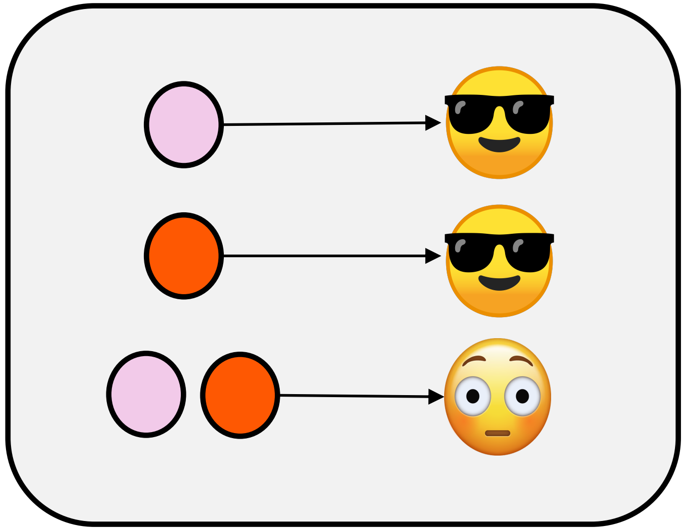
Medication combination
at visit t?

Challenges for Medication Recommendation

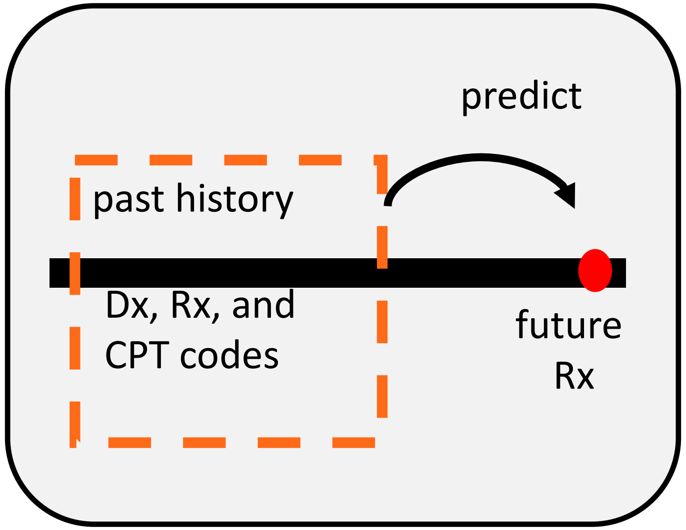
Complex Dependency



Drug-drug Interaction



Patient history



Graph Augmented Memory Networks (GAMENet)

Patient Representation

Graph Augmented Memory Network

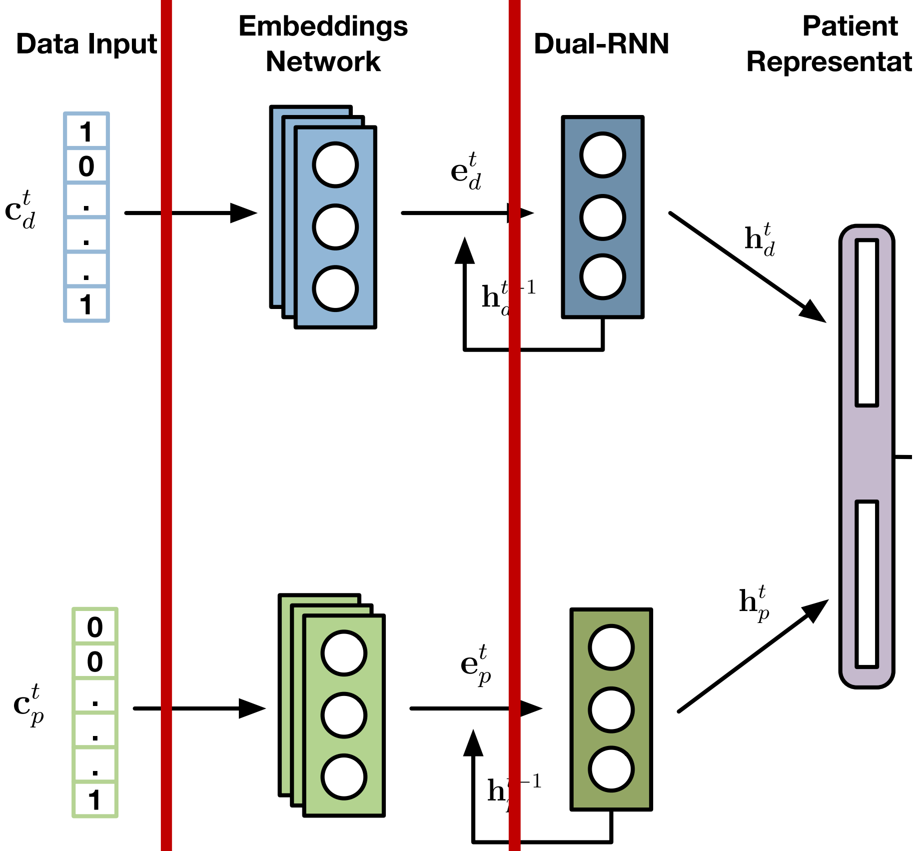
Patient Representation Module

INPUT

Visit codes $\boldsymbol{c}_*^t$

Embedding

$$\boldsymbol{e}_*^t = \boldsymbol{W}_{*,e}\boldsymbol{c}_*^t$$



OUTPUT

Patient Representation

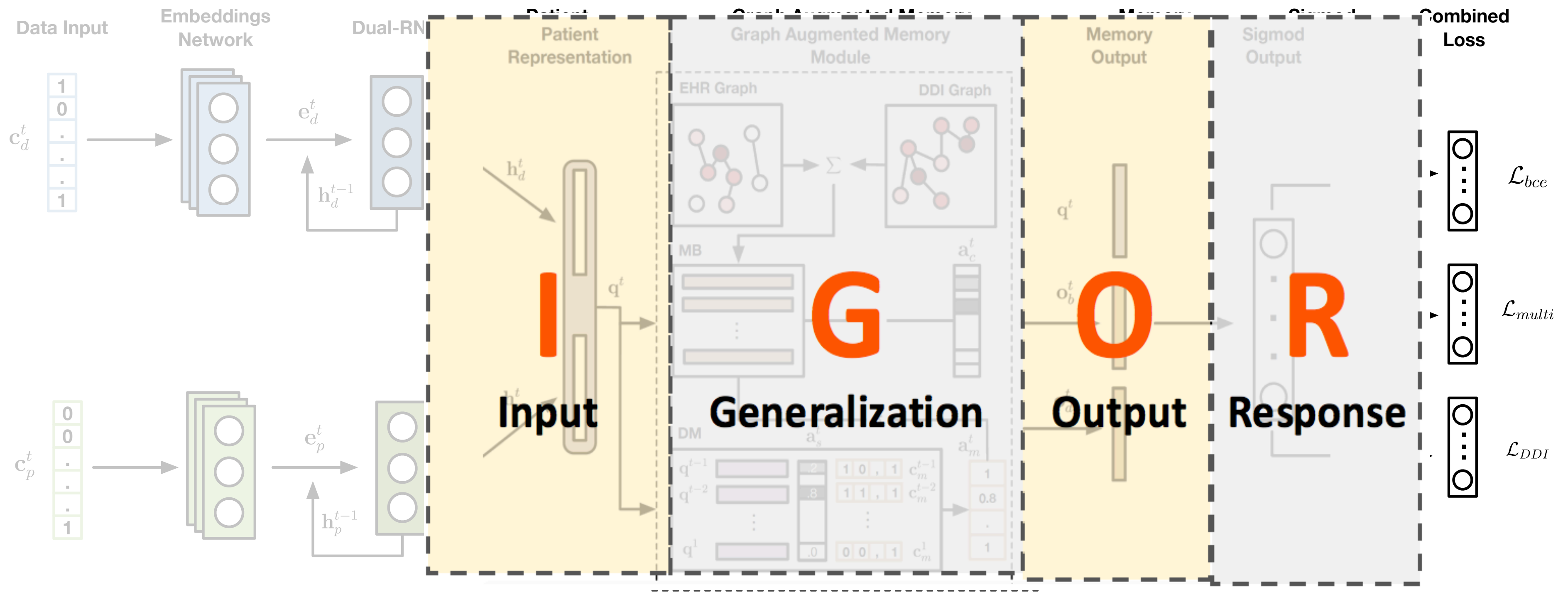
$$[\boldsymbol{h}_d^t, \boldsymbol{h}_p^t]$$

$\boldsymbol{h}_d^t = RNN_d(\boldsymbol{e}_d^1, \cdots, \boldsymbol{e}_d^t)$ (diagnosis)

$\boldsymbol{h}_p^t = RNN_p(\boldsymbol{e}_p^1, \cdots, \boldsymbol{e}_p^t)$ (procedure)

ILLINOIS

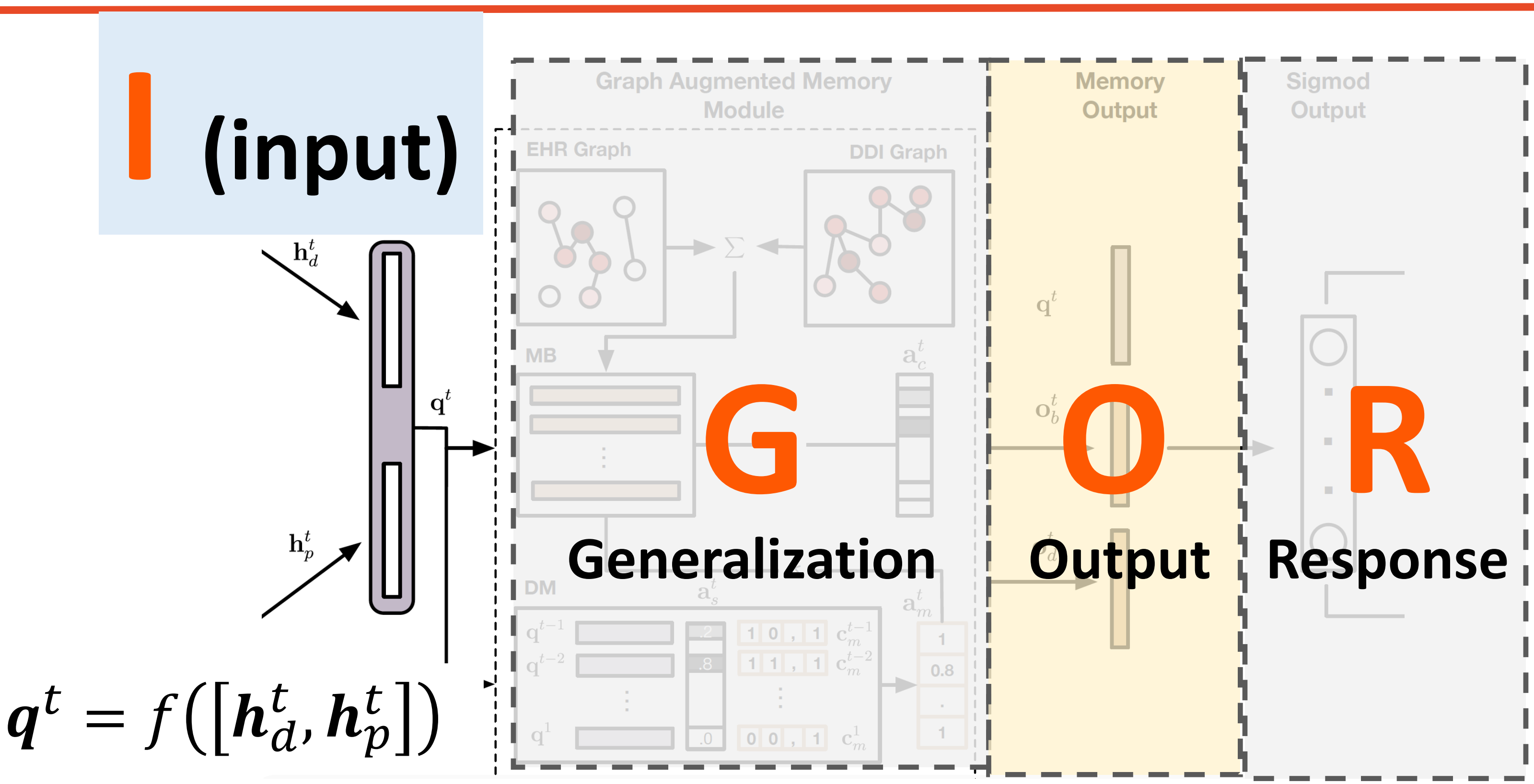
Graph Augmented Memory Module (I, G, O, R)



Graph augmented memory network that comprises of memory components **I, G, O, R.**

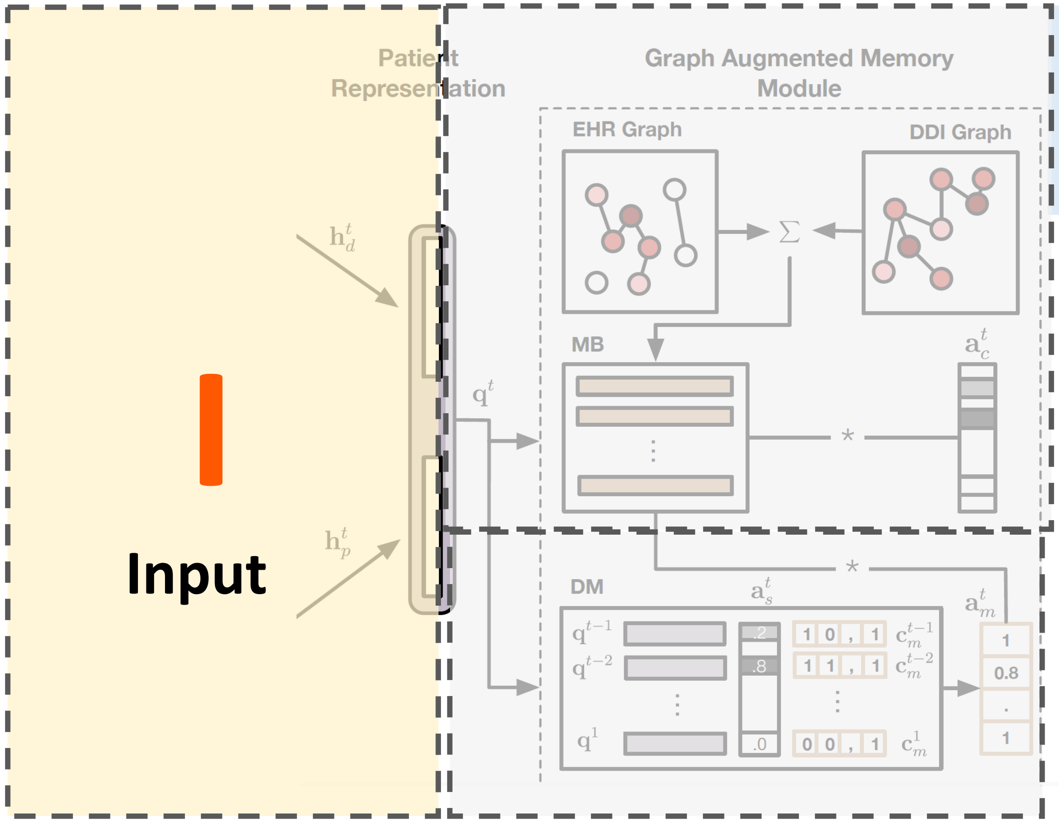
Weston et al., Memory Networks, ICLR 2015

Graph Augmented Memory Module (I, G, O, R)



Medical embedding h_d^t, h_p^t generates patient query q^t.

Graph Augmented Memory Module (I, G, O, R)



G (generalization)

Memory Bank (MB)

Integrate static knowledge graphs

Dynamic Memory (DM)

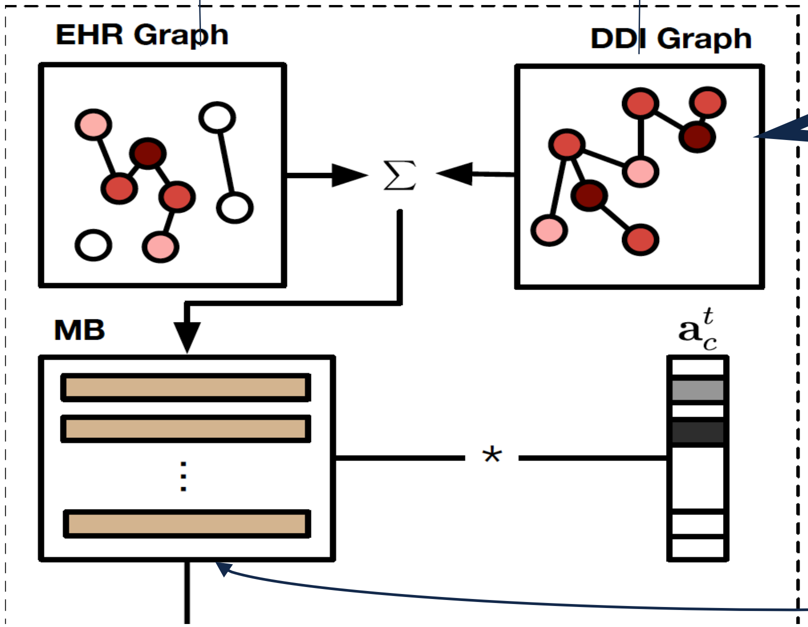
Incorporate patient medication history

Static Memory

Memory Bank (MB)

Co-prescription patterns

Drug-drug interaction knowledge



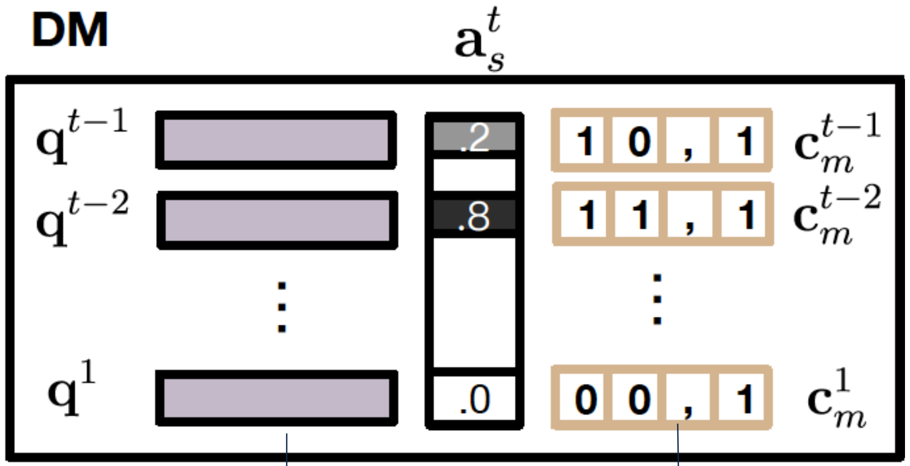
Graph Convolutional Networks

$$\widetilde{A}_* = \widetilde{D}^{-\frac{1}{2}}(A_* + I)\,\widetilde{D}^{-\frac{1}{2}}$$
$$Z_1 = \widetilde{A}_e \tanh(\widetilde{A}_e W_{e1}) W_1$$
$$Z_2 = \widetilde{A}_d \tanh(\widetilde{A}_d W_{e2}) W_2$$
$$M_b = Z_1 - \beta Z_2$$

Output medication embedding $\boldsymbol{M}_b$.

Query $$\boldsymbol{o}_b^t = \boldsymbol{M}_b^{\mathsf{T}} \overbrace{\mathrm{Softmax}(\boldsymbol{M}_b \boldsymbol{q}^t)}^{\boldsymbol{a}_c^t}$$

Dynamic Memory (DM)

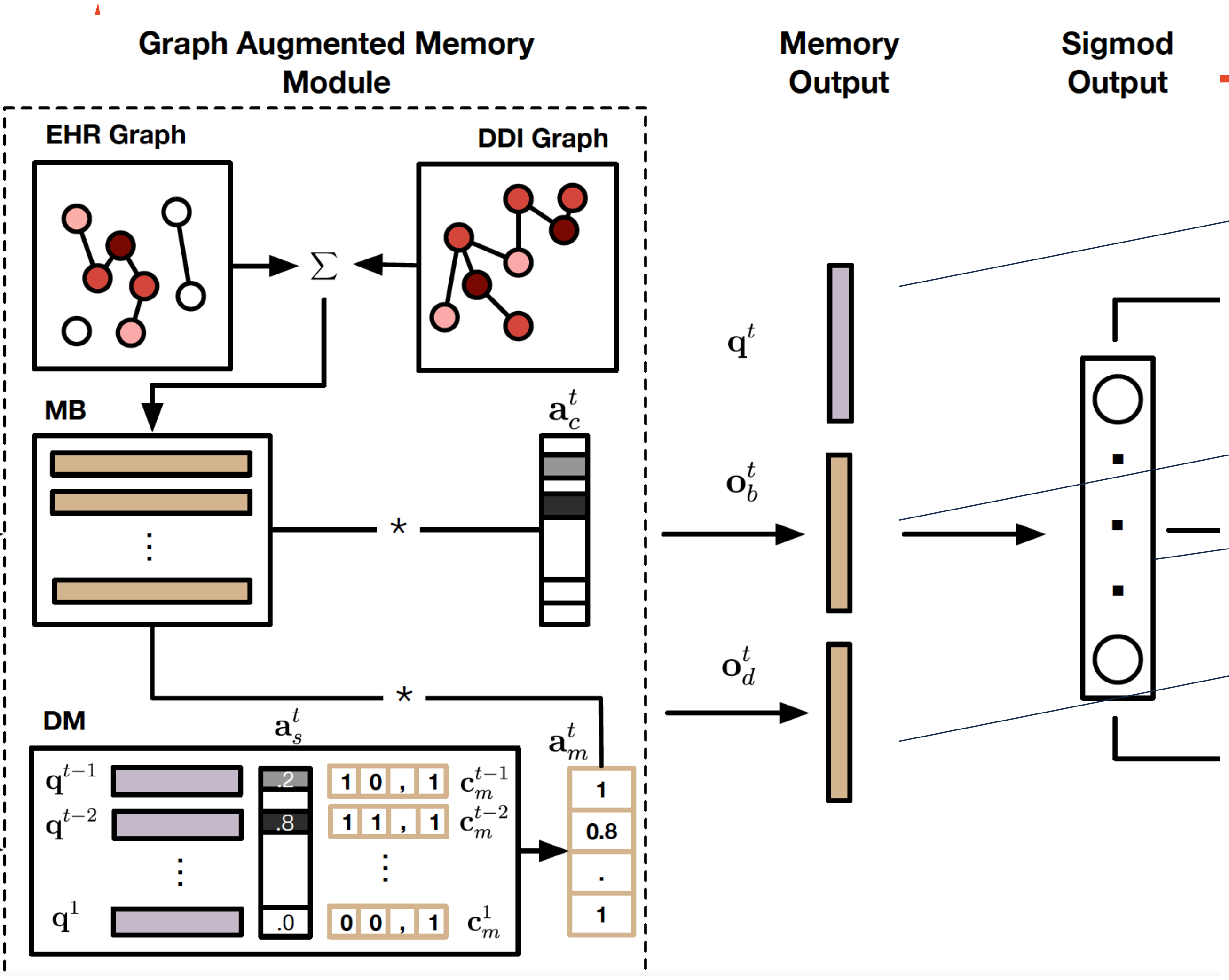


Keys
Patient query $M_{d,k}^t = [\boldsymbol{q}^1, \cdots, \boldsymbol{q}^{t-1}]$

Values
medication history $M_{d,v}^t = [\boldsymbol{c}_m^1, \cdots, \boldsymbol{c}_m^{t-1}]$

Query $\boldsymbol{o}_d^t = \boldsymbol{M}_b^\intercal \overbrace{(\boldsymbol{M}_{d,v}^t)^\intercal \underbrace{\mathrm{Softmax}(\boldsymbol{M}_{d,k}^t \boldsymbol{q}^t)}_{\boldsymbol{a}_s^t}}^{\boldsymbol{a}_m^t}$

Output and Response Module (I, G, O, R)



Patient query

Output from static memory

Output from dynamic memory

Medication prediction
$\hat{y}_t = \sigma([\boldsymbol{q}^t, \boldsymbol{o}_b^t, \boldsymbol{o}_d^t])$

Experiments

Patient Record

Table 2: Statistics of the Data

# patients	6,350
# clinical events	15,016
# diagnosis	1,958
# procedure	1,426
# medication	145
avg # of visits	2.36
avg # of diagnosis	10.51
avg # of procedure	3.84
avg # of medication	8.80
# medication in DDI knowledge base	123
# DDI types in knowledge base	40

MIMIC-III *https://mimic.physionet.org/*

Gold-standard DDI Knowledge

TWOSIDES

Top-40 severe DDIs

http://tatonettilab.org/resources

- ✓ Patient more than one visit.
- ✓ Medication during the first 24 hours.

Results

Method	DDI rate change	Jaccard	F1
Nearest	+1.80%	0.3911	0.5465
LR	+1.16%	0.4075	0.5658
Leap	**-31.53%**	0.3844	0.5410
RETAIN	+2.57%	0.4168	0.5781
DMNC	+22.14%	0.4343	0.5934
GAMENet	**-3.60%**	**0.4509**	**0.6081**

Accurate prediction with fewer DDI

ILLINOIS

Variants of GAMENet

Method	DDI rate change	Jaccard	F1	PRAUC
DDI only	-4.44%	0.4304	0.5894	0.6736
EHR only	-1.12%	0.4257	0.5850	0.6665
Dynamic memory only	4.52%	0.4431	0.6047	0.6891
GAMENet	**-3.60%**	**0.4509**	**0.6081**	0.6904

Static memory

Pre-training of Graph Augmented Transformers for Medication Recommendation

Shang, Junyuan, Tengfei Ma, Cao Xiao, and Jimeng Sun. 2019.
"Pre-Training of Graph Augmented Transformers for Medication Recommendation." *IJCAI*

Outline

✓ Background

✓ Graph Augmented Transformers (G-BERT)

✓ Experiments

ILLINOIS

Outline

- ✓ Background
- ✓ Graph Augmented Transformers (G-BERT)
- ✓ Experiments

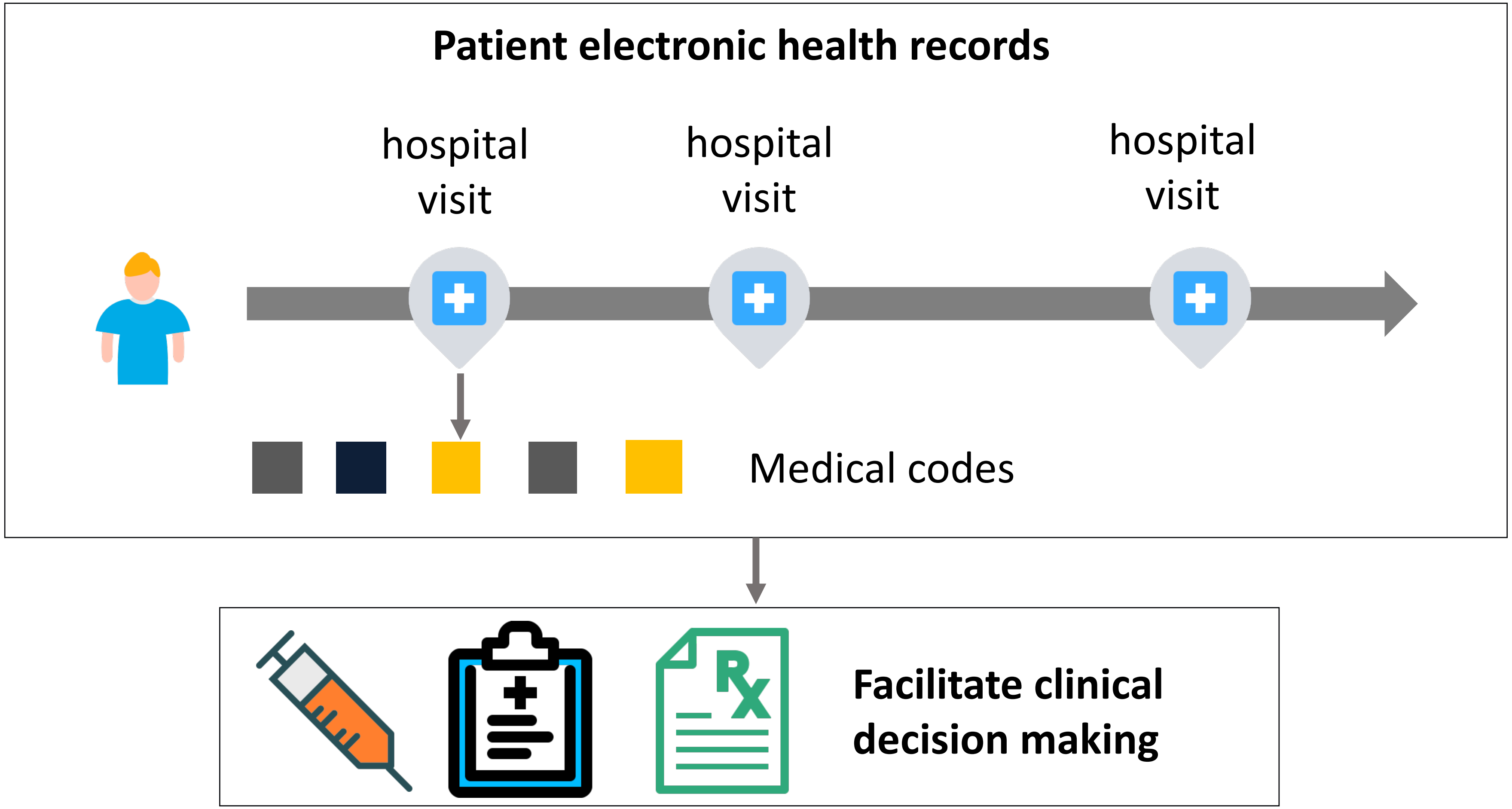
Background

Medical error contributes to 10 percent of all U.S. deaths, and ranks 3rd among all causes of death.

Medical staff shortage problem affect people live in rural areas (~20% of U.S. population).

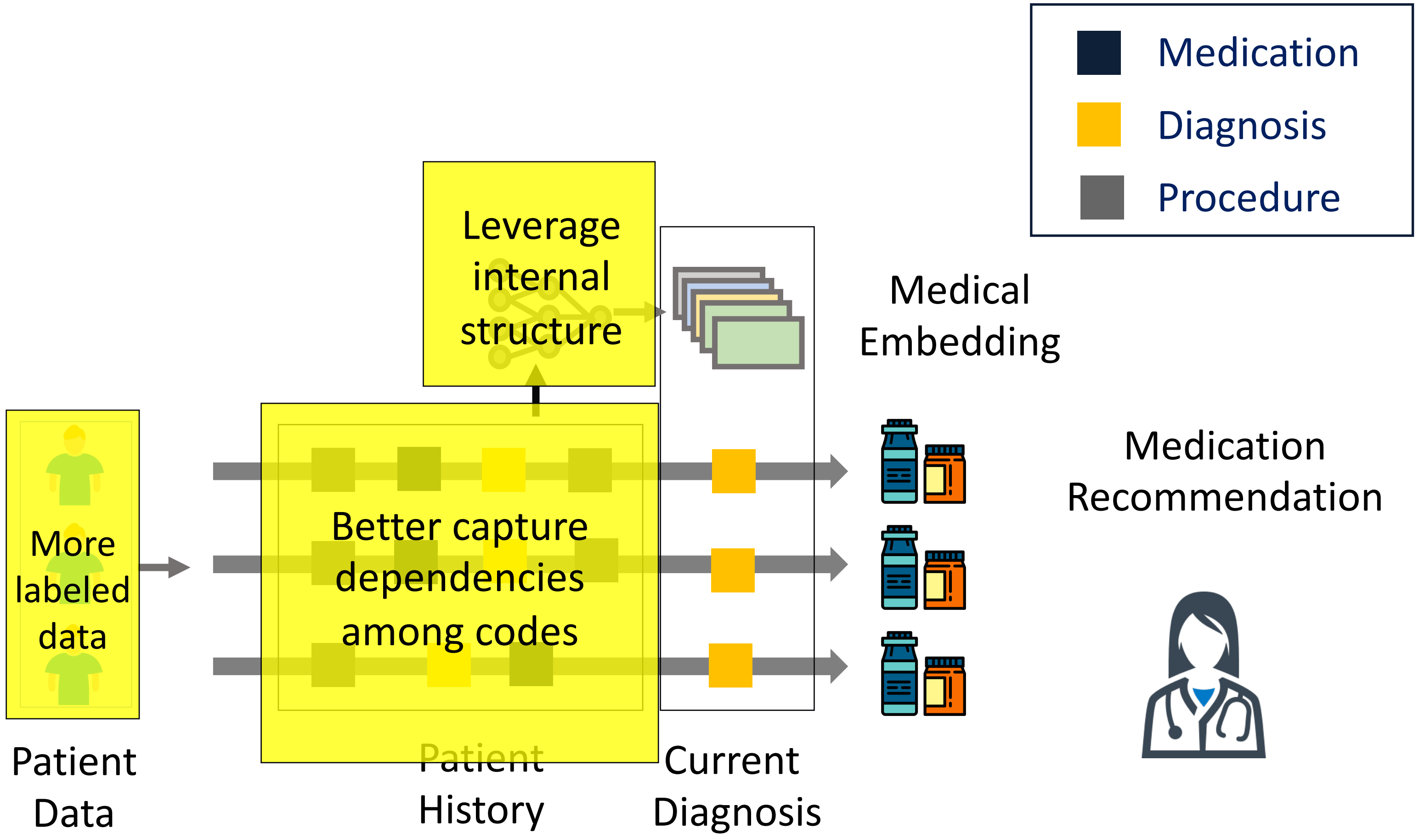
ILLINOIS

Background



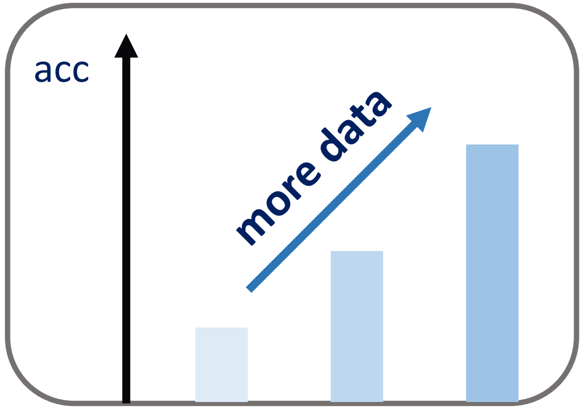
ILLINOIS

Medication Recommendation

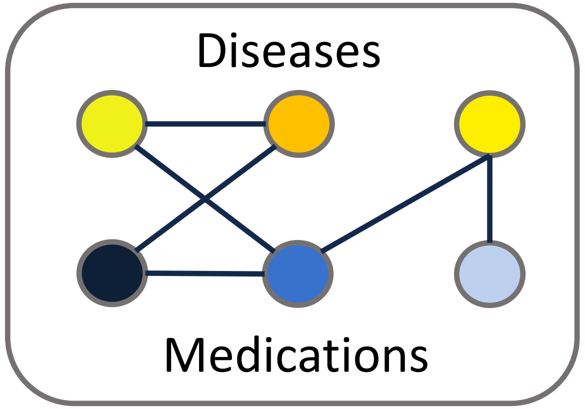


Our solution - How to improve accuracy

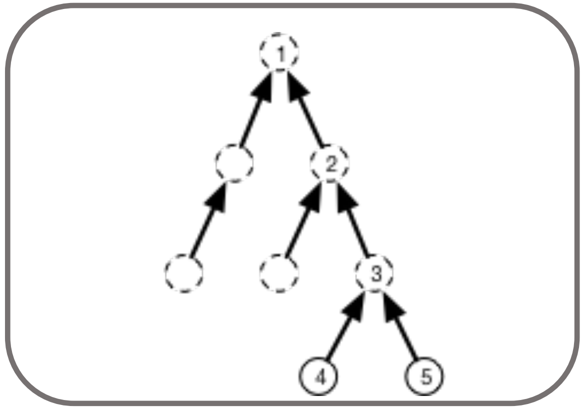
Pre-training



Dependencies Among Codes



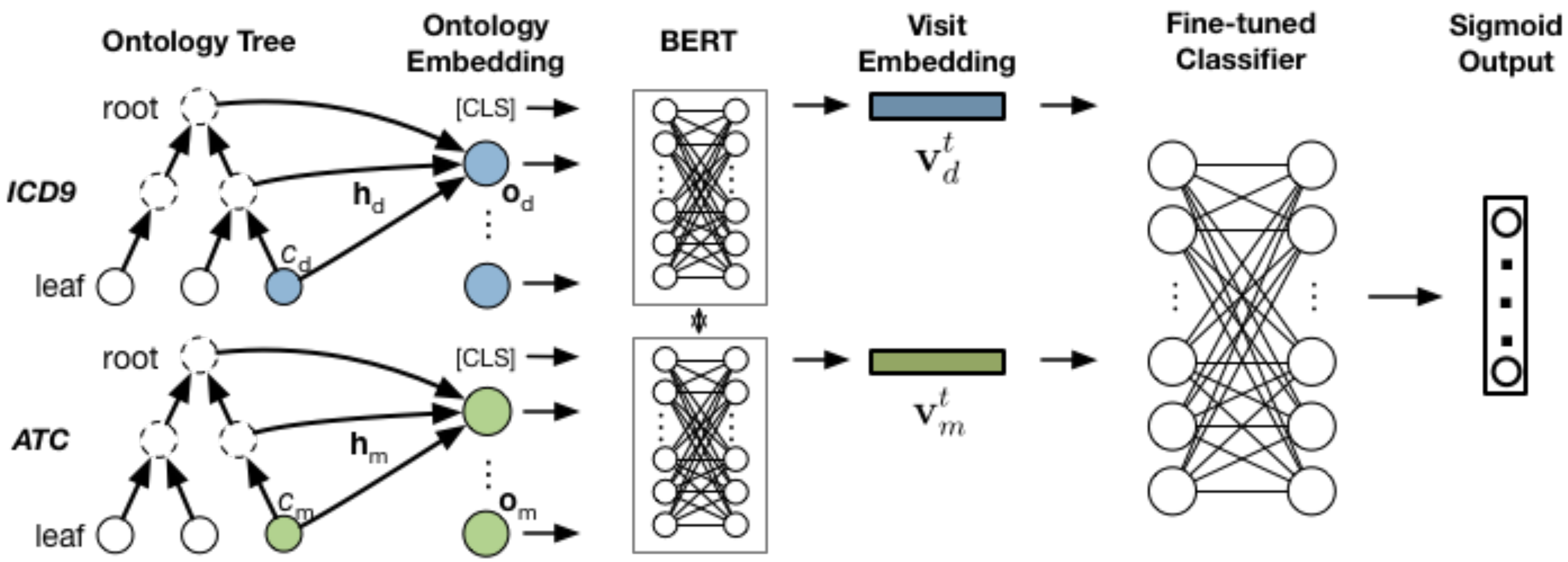
Hierarchical Knowledge



Outline

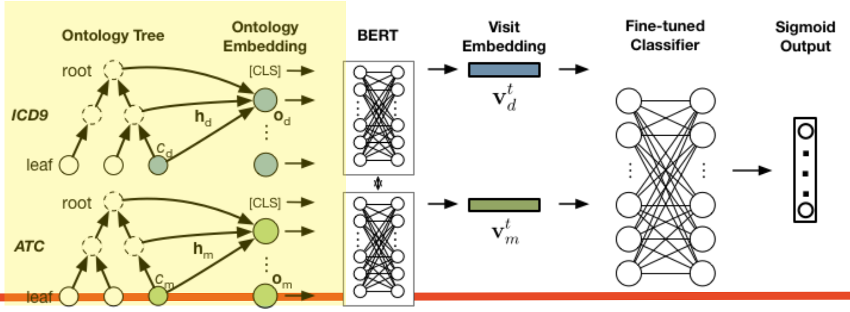
- ✓ Background
- ✓ Graph Augmented Transformers (G-BERT)
- ✓ Experiments

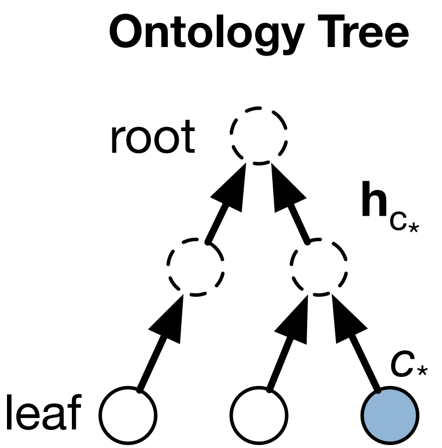
Graph Augmented Bidirectional Encoder Representations from Transformers (G-BERT)



Ontology Embedding Learning ➡ Visit Embedding & Pre-training ➡ Fine-tuning & Recommending

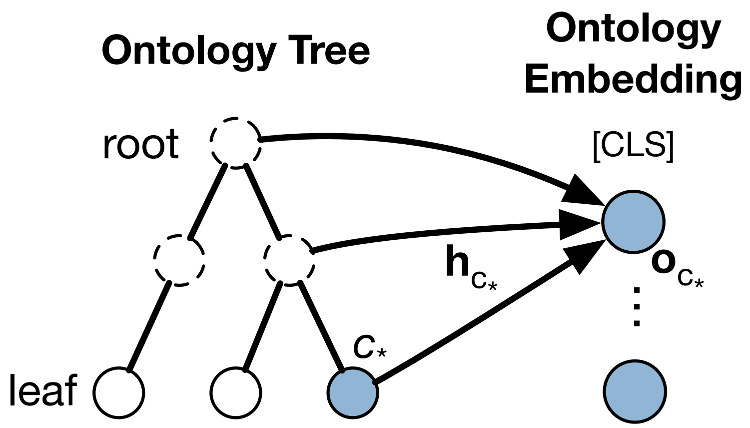
Ontology Embedding Learning





Stage 1. ancestor embeddings

$$\boldsymbol{h}_{c_*} = g(c_*, \mathrm{ch}(c_*), \boldsymbol{W}_e)$$



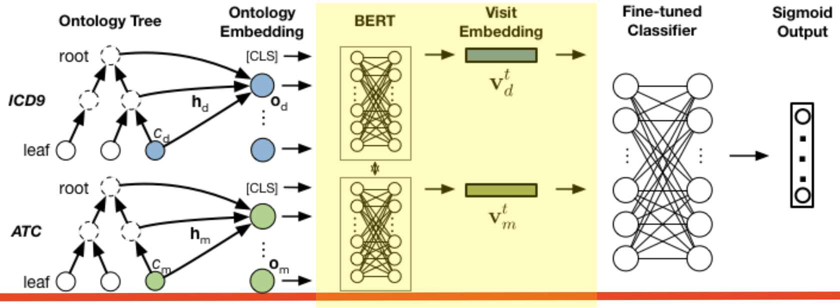
Stage 2. leaf embeddings

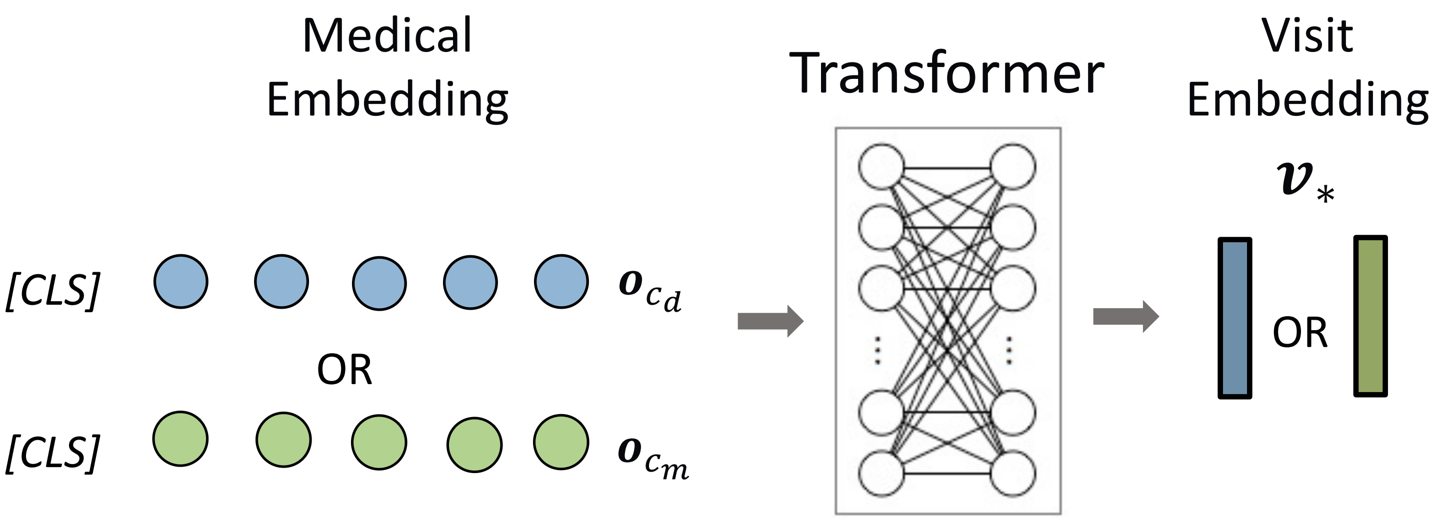
$$\boldsymbol{o}_{c_*} = g(c_*, pa(c_*), \boldsymbol{H}_e)$$

$$g(c_*, p(c_*), \boldsymbol{H}_e) = \|_{k=1}^{K} \, \sigma\left(\sum_{j \in \{c_*\} \cup pa(c_*)} \alpha_{i,j}^{k} \boldsymbol{W}^{k} \boldsymbol{h}_j\right)$$

Implemented as Graph attentional networks (GAT)

Visit Embedding



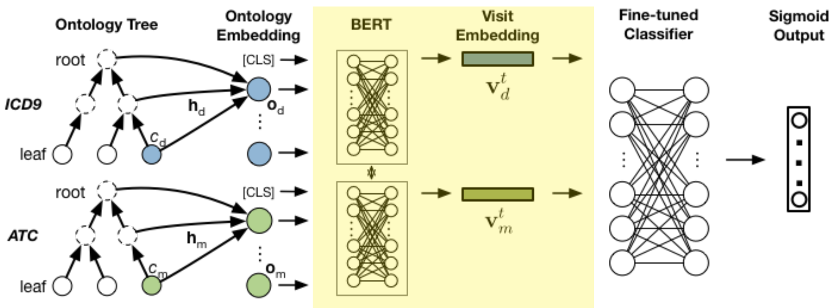


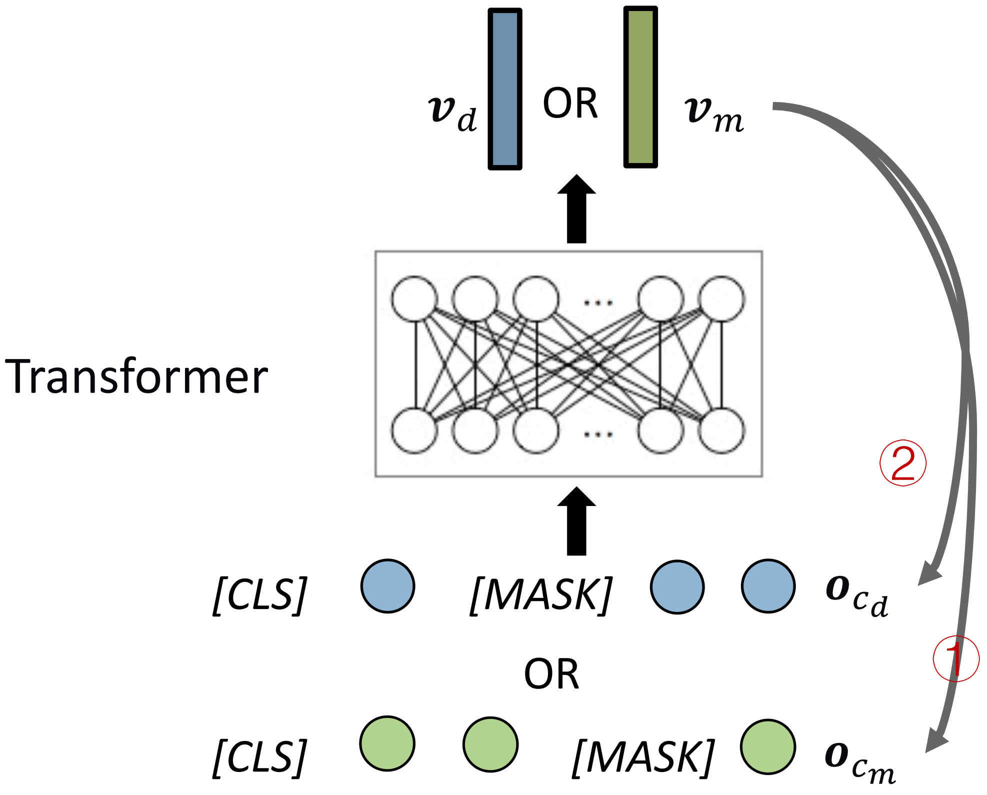
$$\boldsymbol{v}_*^t = \text{Transformer}\left(\{[CLS]\} \cup \left\{\boldsymbol{o}_{c.}^t \middle| c_* \in \mathcal{C}_*^t\right\}\right)[0]$$

[CLS]: Special token in BERT

Vaswani, Ashish, et al. "Attention is all you need." *Advances in neural information processing systems*. 2017.

Pre-training





① Self-prediction

Goal: the visit embedding to recover what it is made of

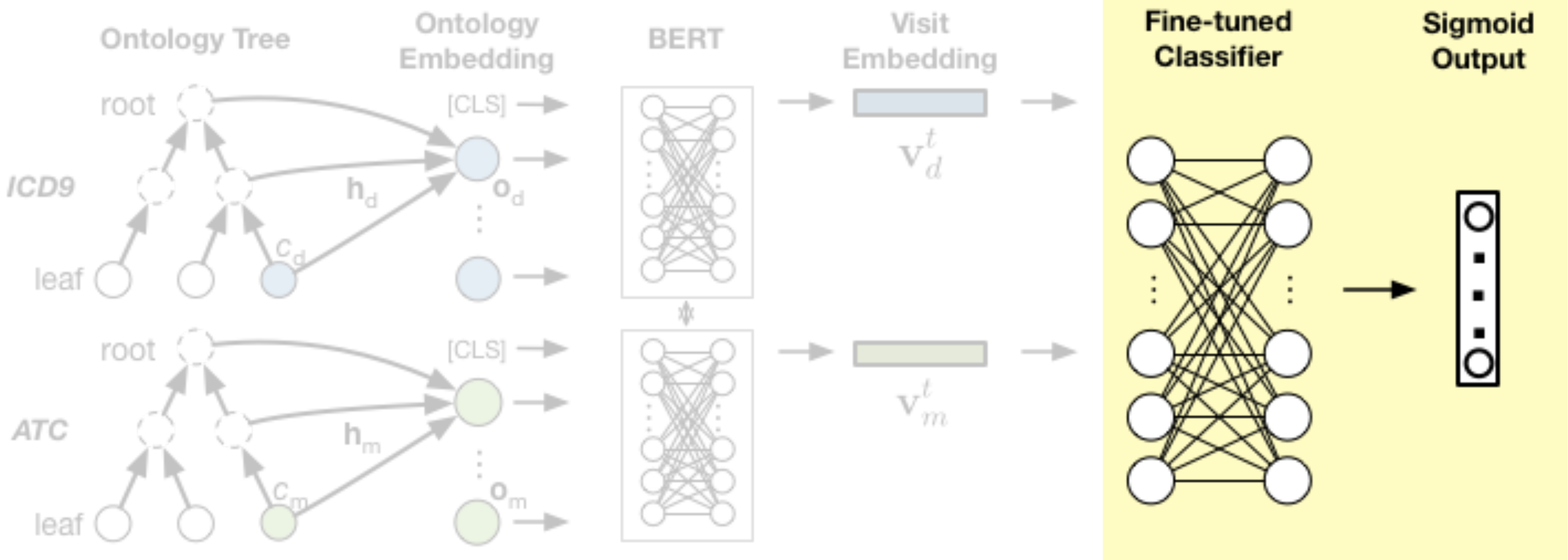
$$\mathcal{L}_{se}(\boldsymbol{v}_*^1, \mathcal{C}_*^1) = -\log p(\mathcal{C}_*^1 | \boldsymbol{v}_*^1)$$
$$= -\sum_{c \in \mathcal{C}_*^1} \log p(c_*^1 | \boldsymbol{v}_*^1) + \sum_{c \in \mathcal{C}_* \backslash \mathcal{C}_*^1} \log p(c_*^1 | \boldsymbol{v}_*^1)$$

② Dual-prediction

Goal: the visit embedding to learn dependencies between codes

$$\mathcal{L}_{du} = -\log p(\mathcal{C}_d^1 | \boldsymbol{v}_m^1) - \log p(\mathcal{C}_m^1 | \boldsymbol{v}_d^1)$$

Fine-tuning & Recommending



Given patient history visit embeddings $\boldsymbol{v}_*^\tau (\tau < t)$, and the latest diagnoses visit embedding $\boldsymbol{v}_d^t$, Output is

$$\hat{y}_t = \sigma(\boldsymbol{W}_1 \left[\left(\frac{1}{t-1} \sum \boldsymbol{v}_{\mathrm{d}}^\tau \right), \left(\frac{1}{t-1} \sum \boldsymbol{v}_{\mathrm{m}}^\tau \right), \boldsymbol{v}_d^t \right] + b)$$

Outline

- ✓ Background
- ✓ Graph Augmented Transformers (G-BERT)
- ✓ Experiments

ILLINOIS

Datasets

MIMIC-III

Statistics	1 Visit	>1 Visit
# of patients	30745	6350
Avg # of visits	1	2.36
Avg # of diagnoses	39	10.51
Avg # of medication	52	8.80
# of unique diagnoses	1,997	1,958
# of unique medication	323	145

- ✓ Pre-training: Patient (1 visit) + Patient (>1 visits) in training dataset
- ✓ Diagnoses in ICD-9 form
- ✓ Medication in ATC form

Results

Methods	Jaccard	PR-AUC	F1	# of parameters
LR	0.4075	0.6716	0.5658	-
GRAM	0.4176	0.6638	0.5788	3,763,668
LEAP	0.3921	0.5855	0.5508	1,488,148
RETAIN	0.4456	0.6838	0.6064	2,054,869
GAMENet	0.4555	0.6854	0.6126	5,518,646
G-BERT	**0.4565**	**0.6960**	**0.6152**	3,034,045

Summary

- An end-to-end deep learning model (G-BERT) that is suitable to learn medical representation and tested on Medication Recommendation task
- Graph Neural Networks can be used to learn enhanced medical ontology embedding
- Pre-training technique has huge potential leverage more data
- Transformer for building dependency between and among instances and labels