Weekly Meeting

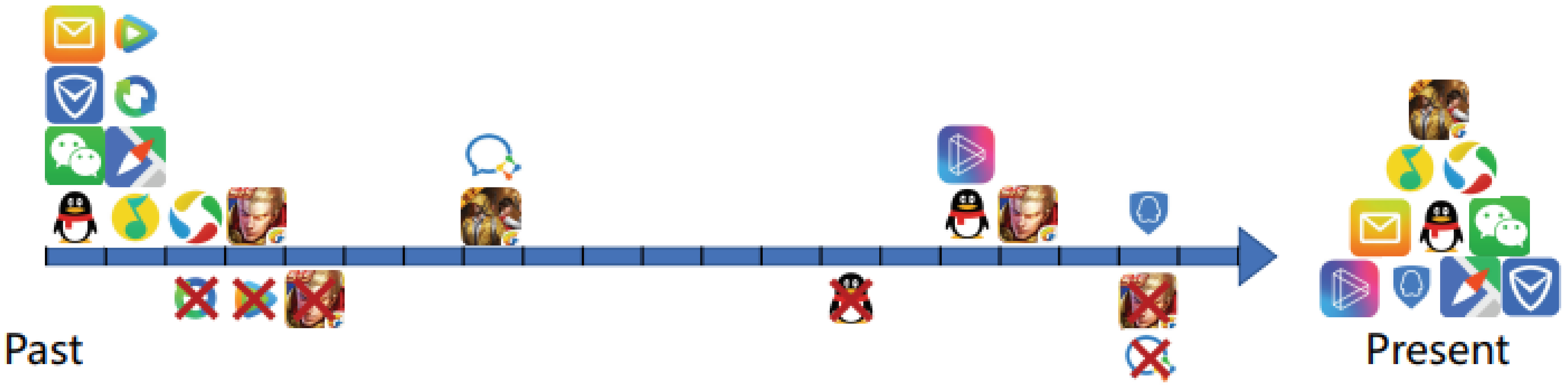
2020.9.13

General-Purpose User Embeddings based on Mobile App Usage

Junqi Zhang[1*], Bing Bai[1*], Ye Lin[1], Jian Liang[1], Kun Bai[1], Fei Wang[2]
[1]Cloud and Smart Industries Group, Tencent, China
[2]Department of Population Health Sciences, Weill Cornell Medicine, USA
{benjqzhang,icebai,yessicalin,joshualiang,kunbai}@tencent.com
few2001@med.cornell.edu

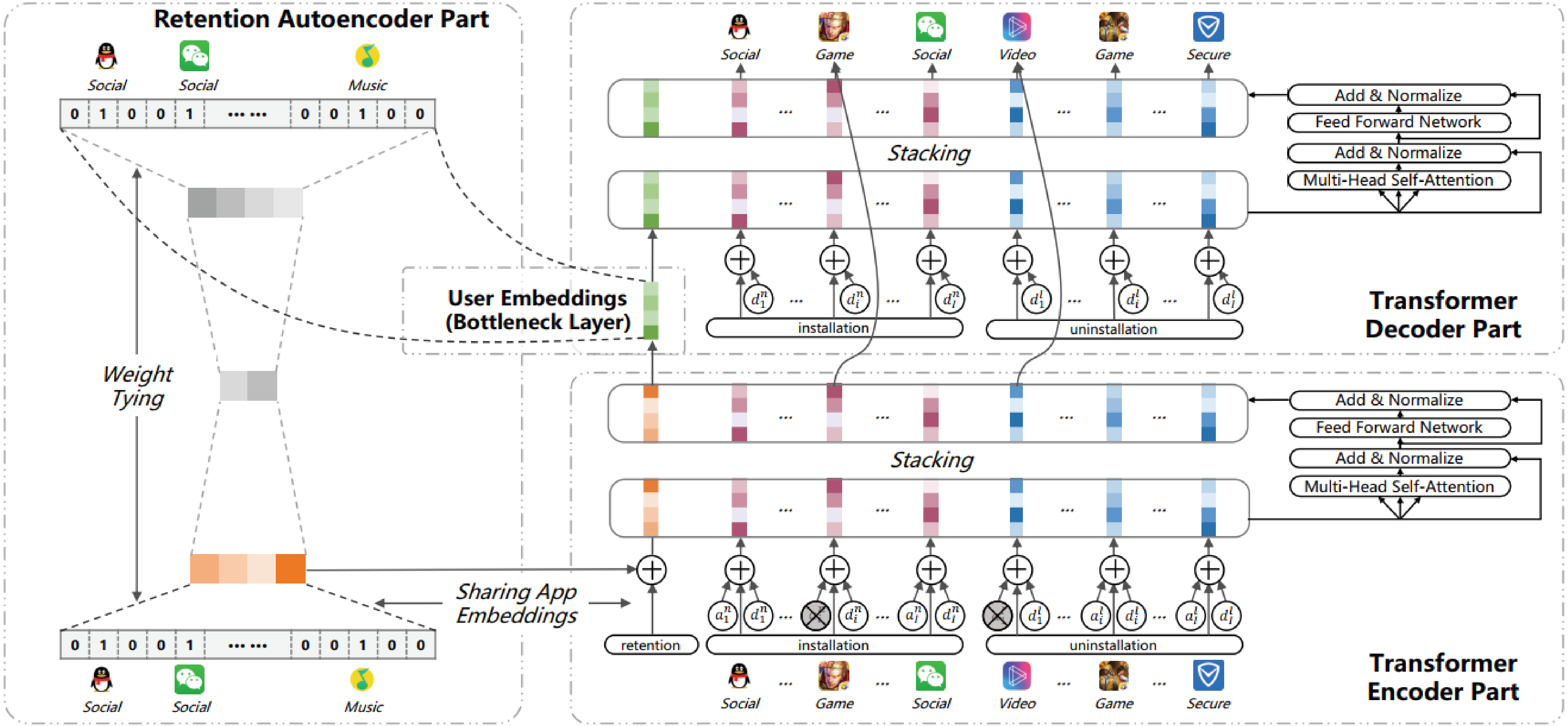
Motivation

- Retention, installation, and uninstallation are heterogeneous but need to be modeled collectively
- User behaviors are distributed unevenly over time
- Many longtailed apps suffer from serious sparsity



Methods

AutoEncoder-coupled Transformer Network (AETN)



Methods

- **Retention:** Multi-hot vector
- **Installing and Uninstalling:** $$\mathcal{S}_u = \{[a^n_{u,1}, \ldots, a^n_{u,i}, \ldots, a^n_{u,I}], [d^n_{u,1}, \ldots, d^n_{u,i}, \ldots, d^n_{u,I}], [a^l_{u,1}, \ldots, a^l_{u,i}, \ldots, a^l_{u,I}], [d^l_{u,1}, \ldots, d^l_{u,i}, \ldots, d^l_{u,I}]\}.$$
- **Retention Autoencoder:** $(f^{(p)}, \mathbf{W}^{(p)}, \boldsymbol{b}^{(p)})$

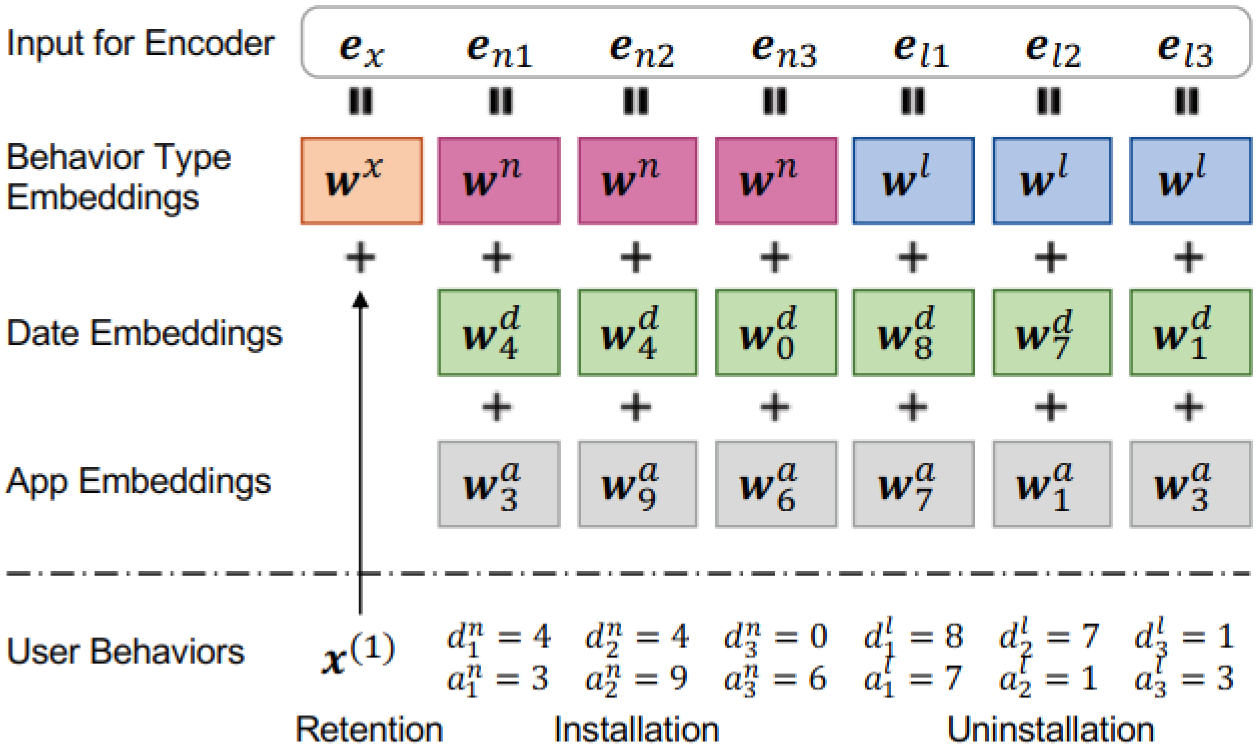
Output of each layer: $\boldsymbol{x}^{(p)} = f^{(p)}(\boldsymbol{x}^{(p-1)}\mathbf{W}^{(p)} + \boldsymbol{b}^{(p)}),\ p \in \{1, 2, 3, 4\},$

The weight matrix of the first hidden layer **W**(1) acts as the shared app embedding matrix $\mathbf{W}a$ for the whole network

this autoencoder provides effective representations of user retention for the transformer part

Methods

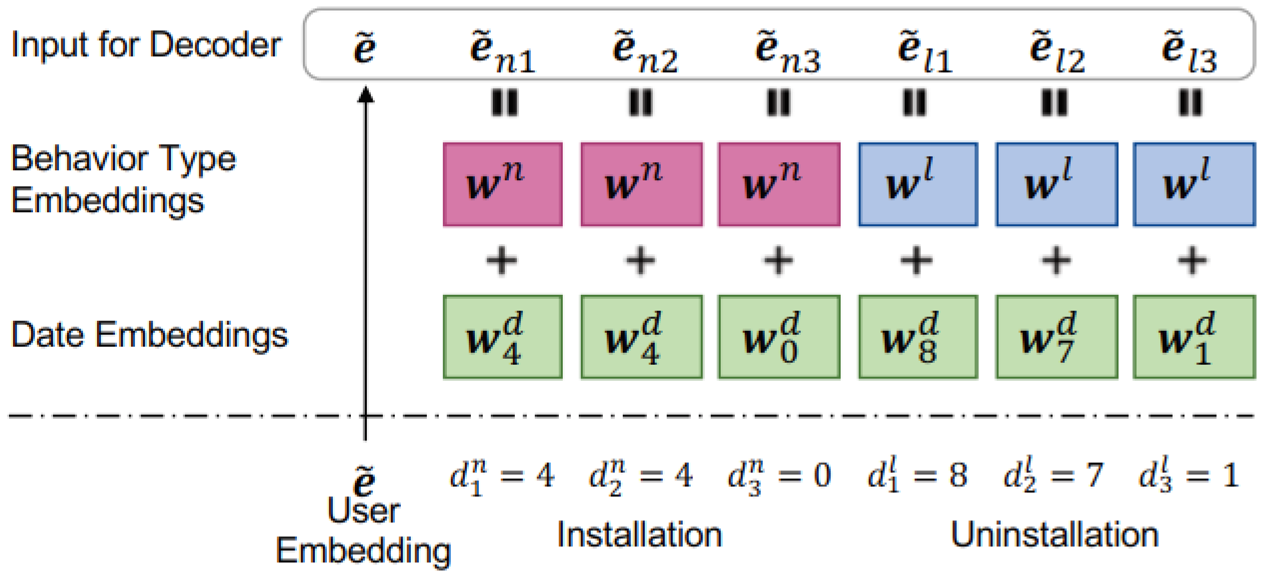
- **Transformer Encoder & Its Embedding Layer:**



- **Bottleneck Layer:** Transformer encoder -> Transformer decoder
  Retention encoder-> MLP decoder

Methods

- **Transformer Decoder & Its Embedding Layer:**



sharing date embeddings and behavior type embeddings with the embedding layer of the encoder

Results

Model	Next Week's Installation Prediction					Look-alike Audience Extension	Feed Recommendation
	Category #1	Category #2	Category #3	Category #4	Average		
DAE	0.7294	0.7297	0.7844	0.7132	0.7392	0.8175	0.6358
AETN w/o $\mathcal{L}_{mask}$	0.7903	0.7818	0.8166	0.7743	0.7908	0.8290	0.6395
AETN w/o $\mathcal{L}_{aux}$	0.8024	0.7913	0.8196	0.7866	0.8000	0.8301	0.6403
V-AETN	0.8014	0.7924	0.8133	0.7746	0.7954	0.8307	0.6401
AETN	**0.8026**	**0.7974**	**0.8215**	**0.7879**	**0.8023**	**0.8309**	**0.6406**

Recurrent Halting Chain for Early Multi-label Classification

Thomas Hartvigsen
Worcester Polytechnic Institute
twhartvigsen@wpi.edu

Cansu Sen
Worcester Polytechnic Institute
csen@wpi.edu

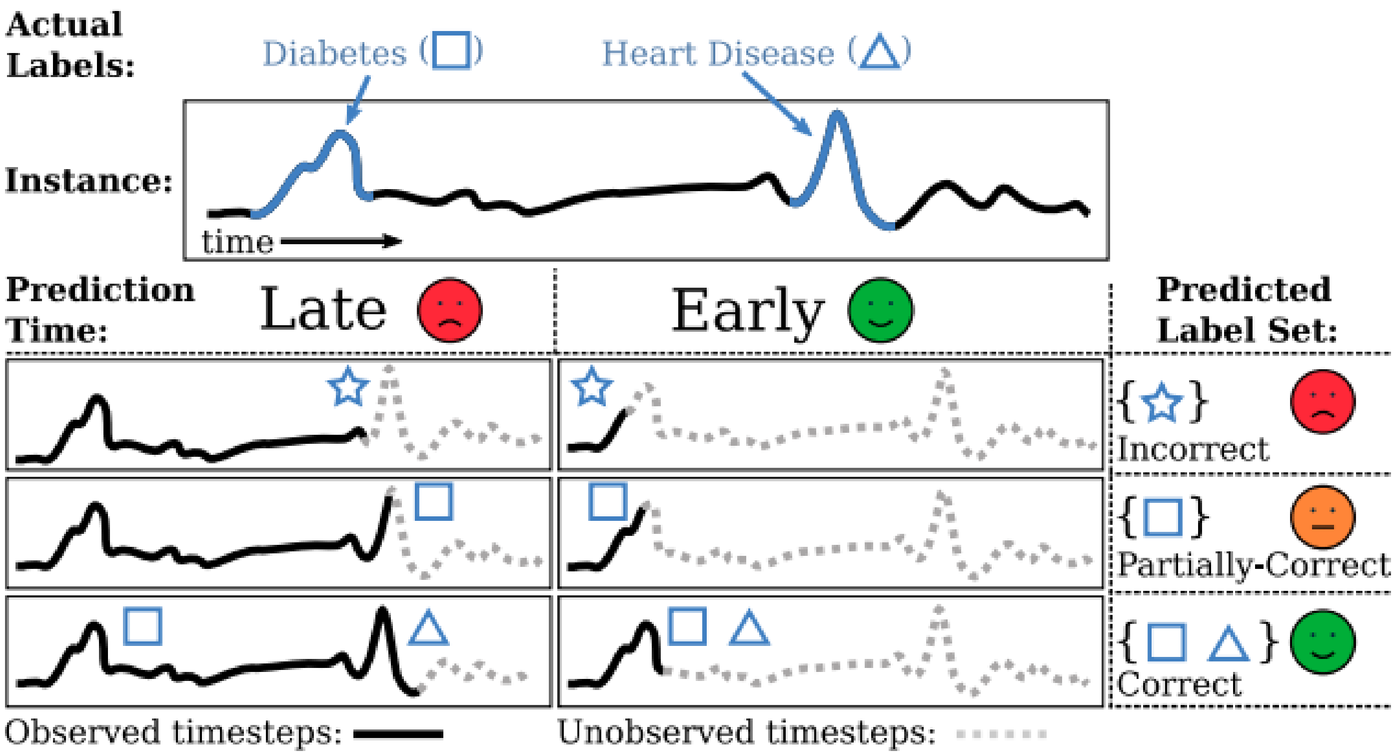
Xiangnan Kong
Worcester Polytechnic Institute
xkong@wpi.edu

Elke Rundensteiner
Worcester Polytechnic Institute
rundenst@wpi.edu

Motivation

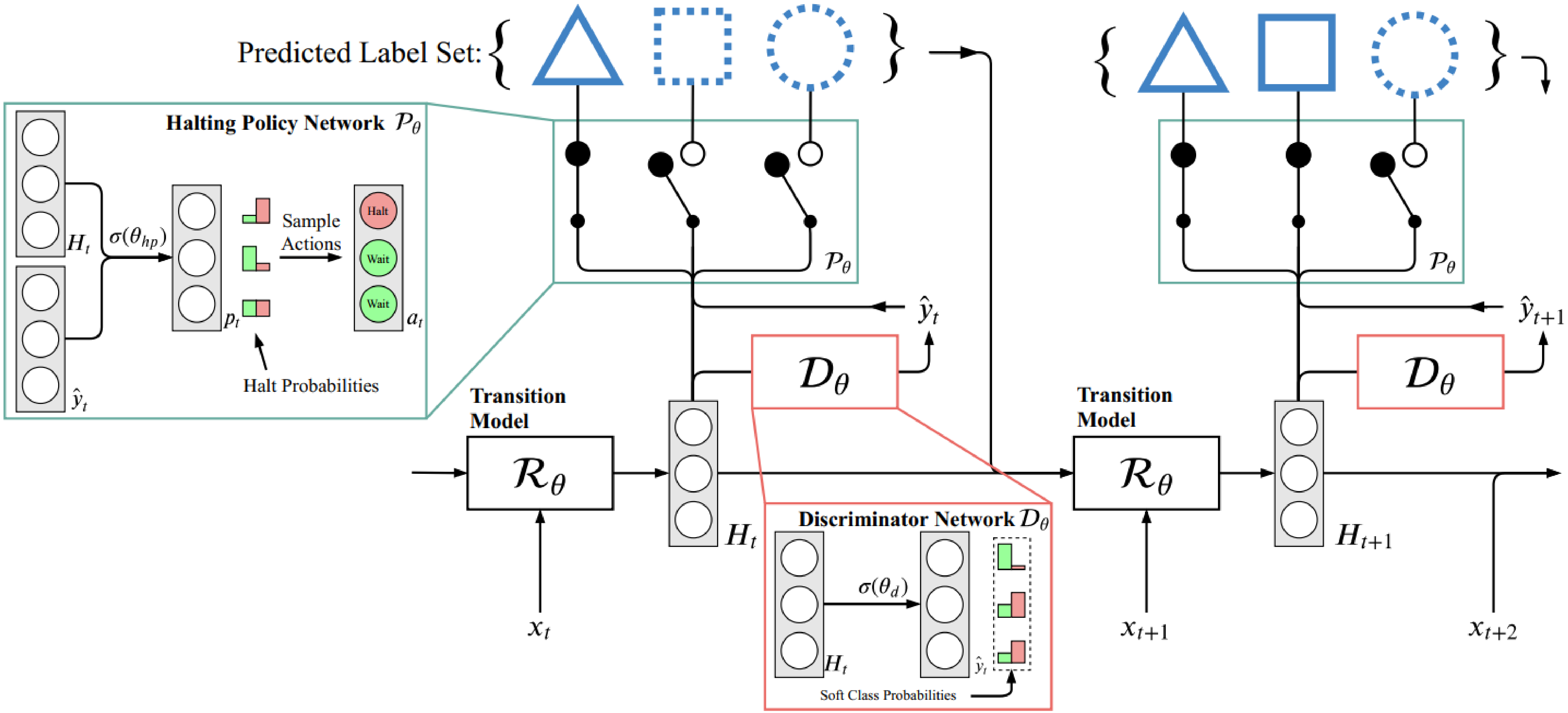
- Early multi-label classification of time series is critical for time-sensitive domains such as healthcare.

- Reliably predicting the correct label set of a time series *while* observing as few timesteps as possible is challenging.



Methods

Recurrent Halting Chain (RHC)



Methods

Transition Model:

- learns to jointly represent the map of $X \rightarrow Y$ and the conditional relationship between labels *while* the labels are being predicted *in time*

$$
\begin{aligned}
f_t &= \sigma(W_f \cdot [H_{t-1}, x_t] + b_f) \\
i_t &= \sigma(W_i \cdot [H_{t-1}, x_t] + b_i) \\
C_t &= f_t \odot C_{t-1} + i_t \odot \phi(W_c \cdot [H_{t-1}, x_t] + b_c) \\
o_t &= \sigma(W_o \cdot [H_{t-1}, x_t] + b_o) \\
H_t &= o_t \odot \phi(C_t)
\end{aligned}
$$

Standard LSTM

$$
\begin{aligned}
f_t &= \sigma(W_f \cdot [H_{t-1}, x_t, \bar{y}_{t-1}] + b_f) \\
i_t &= \sigma(W_i \cdot [H_{t-1}, x_t, \bar{y}_{t-1}] + b_i) \\
C_t &= f_t \odot C_{t-1} + i_t \odot \phi(W_c \cdot [H_{t-1}, x_t, \bar{y}_{t-1}] + b_c) \\
o_t &= \sigma(W_o \cdot [H_{t-1}, x_t, \bar{y}_{t-1}] + b_o) \\
H_t &= o_t \odot \phi(C_t)
\end{aligned}
$$

Update LSTM

Methods

Discriminator Network **:**

- predict soft confidence values *y*ˆ at each timestep *t*

$$\hat{y}_t = \mathrm{P}(Y \mid H_t) = \mathcal{D}_\theta(H_t)$$
$$= \frac{1}{1 + e^{W_{ho}H_t + b_{ho}}}$$

Halting Policy Network **:**

- decides at each step whether or not to halt each class using a *joint-learned representation* to model which classes can be predicted concurrently

- trained using Reinforcement Learning

- rewarded based on how accurately the *Discriminator* predicts each class and punished according to how many steps it takes to make accurate predictions

Results

Dataset:

HAR : Human Activity Recognition

Num of Label:6

Time-Steps Observed	Evaluation Metrics	Methods				
		LSTM-BD [17]	E-LSTM [4]	LSTM-CC [30]	EARLIEST [11]	RHC (ours)
20%	Instance-AUC↑	0.88 (0.00)	0.85 (0.00)	0.90 (0.02)	**0.92** (0.00)	**0.92** (0.01)
	Micro-AUC↑	0.86 (0.00)	0.85 (0.00)	0.88 (0.01)	**0.91** (0.00)	**0.91** (0.00)
	Macro-AUC↑	0.86 (0.00)	0.84 (0.00)	0.88 (0.02)	**0.91** (0.00)	**0.91** (0.00)
	Hamming Loss↓	0.18 (0.00)	0.21 (0.00)	0.17 (0.01)	**0.13** (0.00)	**0.13** (0.01)
	Micro-F1↑	0.62 (0.00)	0.57 (0.00)	0.66 (0.03)	**0.74** (0.00)	0.72 (0.02)
	Macro-F1↑	0.62 (0.00)	0.57 (0.00)	0.65 (0.03)	**0.74** (0.00)	0.71 (0.02)
40%	Instance-AUC↑	0.91 (0.00)	0.89 (0.00)	0.92 (0.02)	**0.94** (0.00)	**0.94** (0.00)
	Micro-AUC↑	0.90 (0.00)	0.90 (0.00)	0.92 (0.02)	0.92 (0.00)	**0.93** (0.00)
	Macro-AUC↑	0.91 (0.00)	0.89 (0.00)	0.92 (0.02)	0.93 (0.00)	**0.94** (0.00)
	Hamming Loss↓	0.17 (0.00)	0.17 (0.01)	0.15 (0.01)	**0.10** (0.00)	**0.10** (0.00)
	Micro-F1↑	0.65 (0.00)	0.67 (0.02)	0.72 (0.02)	0.79 (0.00)	**0.81** (0.00)
	Macro-F1↑	0.63 (0.00)	0.68 (0.02)	0.72 (0.02)	0.79 (0.00)	**0.81** (0.00)
60%	Instance-AUC↑	0.92 (0.00)	0.93 (0.01)	0.93 (0.01)	0.94 (0.00)	**0.95** (0.00)
	Micro-AUC↑	0.92 (0.00)	0.94 (0.01)	0.93 (0.01)	0.93 (0.00)	**0.95** (0.00)
	Macro-AUC↑	0.90 (0.00)	0.94 (0.01)	0.93 (0.01)	0.94 (0.00)	**0.95** (0.00)
	Hamming Loss↓	0.13 (0.00)	0.13 (0.02)	0.13 (0.01)	0.10 (0.00)	**0.08** (0.00)
	Micro-F1↑	0.74 (0.00)	0.77 (0.03)	0.75 (0.02)	0.80 (0.01)	**0.83** (0.00)
	Macro-F1↑	0.74 (0.00)	0.78 (0.03)	0.74 (0.02)	0.80 (0.01)	**0.83** (0.01)

Results

Dataset:

ExtraSensory

Num of Label:52

Time-Steps Observed	Evaluation Metrics	LSTM-BD [17]	E-LSTM [4]	LSTM-CC [30]	EARLIEST [11]	RHC (ours)
20%	Instance-AUC↑	0.71 (0.00)	0.71 (0.00)	0.74 (0.02)	0.71 (0.00)	**0.78** (0.00)
	Micro-AUC↑	0.70 (0.00)	0.69 (0.00)	**0.72** (0.02)	0.71 (0.00)	0.68 (0.00)
	Macro-AUC↑	0.60 (0.00)	0.62 (0.00)	0.63 (0.01)	0.63 (0.01)	**0.68** (0.00)
	Hamming Loss↓	0.31 (0.00)	0.32 (0.00)	0.31 (0.01)	0.31 (0.00)	**0.27** (0.00)
	Micro-F1↑	0.58 (0.00)	0.55 (0.00)	0.58 (0.02)	0.58 (0.00)	**0.61** (0.00)
	Macro-F1↑	0.48 (0.00)	0.47 (0.00)	0.46 (0.01)	0.47 (0.01)	0.46 (0.00)
40%	Instance-AUC↑	0.74 (0.00)	0.73 (0.00)	0.77 (0.01)	0.74 (0.00)	**0.79** (0.00)
	Micro-AUC↑	0.73 (0.00)	0.71 (0.00)	0.75 (0.01)	0.74 (0.00)	**0.78** (0.00)
	Macro-AUC↑	0.66 (0.00)	0.64 (0.00)	0.68 (0.01)	0.67 (0.00)	**0.70** (0.00)
	Hamming Loss↓	0.32 (0.00)	0.32 (0.00)	0.29 (0.01)	0.27 (0.00)	**0.26** (0.00)
	Micro-F1↑	0.56 (0.00)	0.59 (0.00)	0.60 (0.00)	0.60 (0.00)	**0.62** (0.00)
	Macro-F1↑	0.47 (0.00)	0.52 (0.00)	0.48 (0.02)	**0.53** (0.01)	0.48 (0.00)
60%	Instance-AUC↑	0.76 (0.00)	0.75 (0.01)	0.78 (0.01)	0.77 (0.01)	**0.79** (0.00)
	Micro-AUC↑	0.76 (0.00)	0.73 (0.01)	0.77 (0.01)	0.76 (0.01)	**0.78** (0.00)
	Macro-AUC↑	0.71 (0.00)	0.67 (0.01)	0.70 (0.01)	0.69 (0.01)	0.70 (0.00)
	Hamming Loss↓	0.28 (0.00)	0.32 (0.01)	0.28 (0.01)	0.28 (0.00)	**0.26** (0.00)
	Micro-F1↑	0.61 (0.00)	0.63 (0.01)	0.62 (0.01)	**0.62** (0.01)	**0.62** (0.00)
	Macro-F1↑	0.53 (0.00)	0.56 (0.01)	0.53 (0.01)	**0.55** (0.00)	0.47 (0.00)

技术路线

- 提取run_cut：每个数据表示一个行驶段，时间跨度几十秒-几分钟
- 阈值/特征值划分：选择合适的阈值，将run_cut中包含有急转弯，急加速，急减速，高速空挡行驶等危险行为的段落归类出来
- 生成标签：根据最后任务是做行为分类（上述各行为）还是危险性分类（二分类），生成个文件对应的标签。
- 用户建模：分为两种类型的建模，一是时序数据建模，使用AE/VAE/DAE等生成车辆行程的embedding；二是车辆静态特征建模，包括车辆类型（根据初始里程分为新旧车等），行驶时段（该行驶段所处的时间段）等的embedding。两者叠加作为分类模型的输入。
- 分类模型：选取合适的网络（如Feedforward NN）进行分类。
- Metric：AUC,ROC,F1-Score
- Baseline：直接使用LSTM做分类，不同的AE做embedding的效果，without某些特征（突出user profile的作用）。