User Device Interaction Prediction via Relational Gated Graph Attention Network and Intent-aware Encoder

Jingyu Xiao*, Qingsong Zou*, Qing Li ‡, Dan Zhao, Kang Li, Wenxin Tang, Runjie Zhou, Yong Jiang
Background

- **Smart Home**: Internet of Things (IoT) devices have been increasingly involved in home life.
- **User Device Interaction (UDI)**: interaction between user and device reflects the habits of users using IoT devices.
Background

- User Device Interaction (UDI) prediction is necessary for smart homes
  - It can help **remind users of key operations** they have forgotten, e.g., “close the water valve”
  - It can help **automate device control** and realize fully automatic whole-house intelligence, e.g., turn on the dryer automatically after washing the clothes.
2 Problem Definition

• User Device Interaction (UDI) Prediction in Smart Home:

  • Given a behavior sequence \( s = [b_1, b_2, ..., b_n] \), where \( b = [t, d, c, i] \) consists of time \( t \), device \( d \), device control \( c \) and intent \( i \). For example, \( b = [2022-10-15 11:30, \text{oven}, \text{oven:switch}, \text{cooking}] \) describes the behavior turn on the oven at 11:30 on 2022-10-15, with the intent of cooking.

  • The UDI prediction aims at predicting next behavior \( b_{n+1} \).
Complex and Heterogeneous Transitions:

- **Complex**: there are transitions not only between consecutive devices, but also in broader contexts.
- **Heterogeneous**: transitions caused by different device control.

![Diagram](Image)
Challenge #2

- Multiple Intents:
  - **Laundry-related**: water valve, washing machine and dryer.
  - **Cooking-related**: oven, microwave and dish washer.
Multi-level Periodicity:

- **Day-level Periodicity:** user leaves work on time on Wednesday and Thursday, but works overtime once every 3 days, on Tuesday and Friday.

- **Week-level Periodicity:** user stays up late for games every Saturday night.
We propose **DeepUDI**: A novel approach for accurate UDI prediction.

**Idea #1: Relational Gated Graph Attention Network**
- To construct graph from behavior sequence and leverage relational gated GNN to capture the transitions between different device.

**Idea #2: Intent-aware Encoder**
- To view intents as capsules and capture multiple intents by capsule network.

**Idea #3: Historical Attention Mechanism**
- To model correlation between current sequence and historical sequences by attention mechanism.
Multi-Modal Embedding Layer

- Time Embedding
  - Time2vec can capture both periodic and non-periodic patterns of time.
  - We denotes time as day of week and hour of day.
  - For time $\tau$, $t2v(\tau)[i]$ denotes the $i$-th element of time embedding, $\omega_i$ and $\varphi_i$ are learnable parameters, $F$ is a periodic activation function e.g., a sine function.

$$t2v(\tau)[i] = \begin{cases} 
\omega_i \tau + \varphi_i, & \text{if } i = 0 \\
F(\omega_i \tau + \varphi_i), & \text{if } 1 \leq i \leq L - 1
\end{cases}$$
Multi-Modal Embedding Layer

- Device and Device Control Embedding
- Relational Sequence Graph (RS-Graph):
  - each node in the graph represents a device $d$, each edge $(d_{n-1}, d_n)$ indicates that the user accesses device $d_n$ after accessing device $d_{n-1}$
  - each device control represents a relation $r$
Multi-Modal Embedding Layer

- Device and Device Control Embedding
- Relational Gated Graph Attention Network:
  - message passing based on relation.
  - feature update by GRU.
Intent-aware Encoder

- Self-attention and capsule network are employed as Intent-aware Encoder.
  - Self-attention Layer: mine global semantic information of behavior.
  - Capsule Network: we treat behavior as primary capsules and use dynamic routing to learn the probability of high-level intent capsule.
  - Finally, we aggregate intent vector $c$ as sequence embedding $C$.
  - $W_C$ is the learnable weight of different intents.

\[ C = W_C[c_1, ..., c_K] \]
Historical Attention Mechanism

- Summarize the history sequence vector \([C_1, C_2, C_3, \ldots C_{t-1}]\) into \(p\) to capture the multi-level periodicity
  - \(\alpha_i\) and \(\beta_i\) are normalized and unnormalized scores of \(C_i\) for \(C_t\), respectively, \(W_H\) is the learnable weight.

\[
\alpha_i = \frac{\exp(\beta_i)}{\sum_{j=1}^{t-1} \exp(\beta_j)}
\]

\[
\beta_i = \tanh(C_t W_H C_i)
\]

\[
p = \text{Concat} \left( C_t, \sum_{i=1}^{t-1} \alpha_i C_i \right)
\]

\[
\hat{y} = \text{softmax} \left( W_p p \right)
\]
Datasets

- **Datasets**: we use four real-world datasets to evaluate DeepUDI
  - Three datasets *(US/SP/FR)* from public dataset
  - One dataset *(AN)* collected by ourselves
  - Datasets are split into training/validation/testing with a ratio of 7:1:2
  - All sequence instances are of length 10, and we use the first 9 behaviors as input to predict the next behavior
  - Eight intents: entertainment, shower, sleep/getup, leave/return, study, cooking, cleaning, others.

<table>
<thead>
<tr>
<th>Name</th>
<th>Time period (Y-M-D)</th>
<th>Sizes</th>
<th># Devices</th>
<th># Device controls</th>
</tr>
</thead>
<tbody>
<tr>
<td>US</td>
<td>2022-02-22~2022-03-21</td>
<td>67,882</td>
<td>40</td>
<td>268</td>
</tr>
<tr>
<td>SP</td>
<td>2022-02-28~2022-03-30</td>
<td>15,665</td>
<td>34</td>
<td>234</td>
</tr>
<tr>
<td>FR</td>
<td>2022-02-27~2022-03-25</td>
<td>4,423</td>
<td>33</td>
<td>222</td>
</tr>
<tr>
<td>AN</td>
<td>2022-07-31~2022-08-31</td>
<td>1,765</td>
<td>36</td>
<td>141</td>
</tr>
</tbody>
</table>
Baselines and Evaluation Metrics

- **Baselines**: we compare DeepUDI with 7 competitors
  - Traditional Models: HMM and FPMC
  - RNN-based Models: LSTM, CA-RNN and DeepMove
  - GNN-based Models: SR-GNN
  - Transformer-based Models: SmartSense

- **Evaluation Metrics**:
  - Acc@K: Top-K accuracy
    \[
    \text{Acc@K} = \frac{|\{s \in S : p(s) \in P_K(s)\}|}{|S|}
    \]
  - Macro-F1: Macro averaging of F1 score
    \[
    \text{Macro-F1} = \frac{\sum_c F1_c}{|C|}
    \]
Questions

- We answer the following research questions:
  - **RQ1 (Performance).** Compared with other methods, does DeepUDI have higher prediction performance of user device interaction?
  - **RQ2 (Ablation study).** How do main components of DeepUDI affect the performance of UDI prediction?
  - **RQ3 (Parameter study).** How key parameters affect the performance of DeepUDI?
  - **RQ4 (Interpretability study).** Can DeepUDI give a reasonable explanation for the prediction results?
- **RQ1:** Compared with other methods, does DeepUDI have higher prediction performance of user device interaction?
- **A1:** DeepUDI outperforms all competitors in most cases.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Metric</th>
<th>HMM</th>
<th>FPMC</th>
<th>LSTM</th>
<th>CA-RNN</th>
<th>DeepMove</th>
<th>SRGNN</th>
<th>SmartSense</th>
<th>DeepUDI(Ours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AN</td>
<td>Acc@1</td>
<td>0.6099</td>
<td>0.6557</td>
<td>0.7062</td>
<td>0.7026</td>
<td>0.7116</td>
<td>0.9245</td>
<td>0.9407</td>
<td>0.9784</td>
</tr>
<tr>
<td></td>
<td>Acc@3</td>
<td>0.7501</td>
<td>0.7959</td>
<td>0.7843</td>
<td>0.8302</td>
<td>0.9272</td>
<td>0.9864</td>
<td>0.9731</td>
<td>0.9865</td>
</tr>
<tr>
<td></td>
<td>Acc@5</td>
<td>0.7714</td>
<td>0.7902</td>
<td>0.8328</td>
<td>0.9003</td>
<td>0.9542</td>
<td>0.9872</td>
<td>0.9838</td>
<td>0.9892</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.2439</td>
<td>0.2845</td>
<td>0.3759</td>
<td>0.4159</td>
<td>0.5027</td>
<td>0.7368</td>
<td>0.7519</td>
<td>0.7997</td>
</tr>
<tr>
<td>FR</td>
<td>Acc@1</td>
<td>0.6536</td>
<td>0.6814</td>
<td>0.6962</td>
<td>0.7893</td>
<td>0.7762</td>
<td>0.7819</td>
<td>0.7923</td>
<td>0.8144</td>
</tr>
<tr>
<td></td>
<td>Acc@3</td>
<td>0.7813</td>
<td>0.8271</td>
<td>0.8011</td>
<td>0.9148</td>
<td>0.9221</td>
<td>0.9197</td>
<td>0.9371</td>
<td>0.9238</td>
</tr>
<tr>
<td></td>
<td>Acc@5</td>
<td>0.8242</td>
<td>0.8508</td>
<td>0.8565</td>
<td>0.9425</td>
<td>0.9446</td>
<td>0.9435</td>
<td>0.9628</td>
<td>0.9512</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.1127</td>
<td>0.1279</td>
<td>0.1302</td>
<td>0.2102</td>
<td>0.2288</td>
<td>0.2482</td>
<td>0.2603</td>
<td>0.3425</td>
</tr>
<tr>
<td>SP</td>
<td>Acc@1</td>
<td>0.6315</td>
<td>0.6964</td>
<td>0.7517</td>
<td>0.7853</td>
<td>0.7756</td>
<td>0.7815</td>
<td>0.7921</td>
<td>0.7923</td>
</tr>
<tr>
<td></td>
<td>Acc@3</td>
<td>0.7863</td>
<td>0.7916</td>
<td>0.8864</td>
<td>0.8915</td>
<td>0.9125</td>
<td>0.9303</td>
<td>0.9342</td>
<td>0.9375</td>
</tr>
<tr>
<td></td>
<td>Acc@5</td>
<td>0.8361</td>
<td>0.8605</td>
<td>0.9346</td>
<td>0.9117</td>
<td>0.9521</td>
<td>0.9603</td>
<td>0.9511</td>
<td>0.9642</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.1382</td>
<td>0.1586</td>
<td>0.1756</td>
<td>0.1745</td>
<td>0.2159</td>
<td>0.2239</td>
<td>0.2244</td>
<td>0.3112</td>
</tr>
<tr>
<td>US</td>
<td>Acc@1</td>
<td>0.3327</td>
<td>0.3543</td>
<td>0.4286</td>
<td>0.5212</td>
<td>0.5527</td>
<td>0.5784</td>
<td>0.5935</td>
<td>0.6056</td>
</tr>
<tr>
<td></td>
<td>Acc@3</td>
<td>0.6881</td>
<td>0.6992</td>
<td>0.8209</td>
<td>0.8577</td>
<td>0.8844</td>
<td>0.8955</td>
<td>0.9056</td>
<td>0.9123</td>
</tr>
<tr>
<td></td>
<td>Acc@5</td>
<td>0.7258</td>
<td>0.7712</td>
<td>0.8929</td>
<td>0.9135</td>
<td>0.9418</td>
<td>0.9463</td>
<td>0.9489</td>
<td>0.9521</td>
</tr>
<tr>
<td></td>
<td>Macro-F1</td>
<td>0.1069</td>
<td>0.1123</td>
<td>0.1265</td>
<td>0.1396</td>
<td>0.2388</td>
<td>0.2431</td>
<td>0.2451</td>
<td>0.3538</td>
</tr>
</tbody>
</table>
**RQ2:** How do main components of DeepUDI affect the performance of UDI prediction?

**A2:** All three components (MME: Multi-Modal Embedding Layer, IAE: Intent-aware encoder and HAM: Historical Attention Mechanism) of DeepUDI are contributive for UDI prediction.

<table>
<thead>
<tr>
<th>Model</th>
<th>AN (Acc@1</th>
<th>Macro-F1)</th>
<th>FR (Acc@1</th>
<th>Macro-F1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DeepUDI-MME</td>
<td>0.9487</td>
<td>0.7364</td>
<td>0.7851</td>
<td>0.2805</td>
</tr>
<tr>
<td>DeepUDI-IAE</td>
<td>0.9595</td>
<td>0.7585</td>
<td>0.7853</td>
<td>0.3103</td>
</tr>
<tr>
<td>DeepUDI-HAM</td>
<td>0.9676</td>
<td>0.7812</td>
<td>0.7988</td>
<td>0.3234</td>
</tr>
<tr>
<td>DeepUDI-ALL</td>
<td>0.9137</td>
<td>0.6934</td>
<td>0.7578</td>
<td>0.2511</td>
</tr>
<tr>
<td>DeepUDI</td>
<td><strong>0.9784</strong></td>
<td><strong>0.7997</strong></td>
<td><strong>0.8144</strong></td>
<td><strong>0.3425</strong></td>
</tr>
</tbody>
</table>
Experiment Results

- **RQ3**: How key parameters affect the performance of DeepUDI?
- **A3**: The best parameter combination: # of layers of RGGAT=2, Embedding Dimension=50, # of History Sequence=15, Batch Size=512.
Experiment Results

- **RQ4**: Can DeepUDI give a reasonable explanation for the prediction results?
- **A4**: DeepUDI successfully learns the correlation between devices by RGGAT and intents of user by capsule network.
Conclusions

- We propose DeepUDI for accurate UDI prediction.
- Our main contributions are summarized as follows:
  - Idea #1: Relational Gated Graph Attention Network
  - Idea #2: Intent-aware Encoder
  - Idea #3: Historical Attention Mechanism
- DeepUDI consistently outperforms state-of-the-art baselines and also offers highly interpretable results.
Thank you!

• Speaker: Jingyu Xiao
• Homepage: https://whalexiao.github.io/
• Email: jy-xiao21@mails.tsinghua.edu.cn