Time-Aware Prospective Modeling of Users for Online Display Advertising

Djordje Gligorijevic, Jelena Gligorijevic and Aaron Flores

Presented by: Djordje Gligorijevic
Prospective Display
Advertising Introduction
Prospective display advertising

Retail adv. running a **prospecting** man suits sale campaign

Julie

Dave

“prom date gift”

search results

limo booking receipt

suit ad

Advertiser’s website

DSP analytics engine

purchase
Prospective display advertising - Reality

Retail adv. running a **prospecting** man suits sale campaign

Julie

Dave

“prom date gift”

search results

Suit retailer websites visits

DSP analytics engine

Dave

Suit retailer

suit ad

purchase

limo booking receipt

Julie

“prom date gift”

search results

Suit retailer websites visits

DSP analytics engine

Dave

Suit retailer

suit ad

purchase

limo booking receipt
Problem statement
Problem definition

Challenge: More and more advertisers are interested in prospective advertising while current systems tend to underperform there.

Problem: Powerful signals often referred as retargeting events overwhelm predictive systems

<table>
<thead>
<tr>
<th>Conversion</th>
<th>Adv. site visit</th>
<th>Site visit prior to conv.</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUE</td>
<td>FALSE</td>
<td>FALSE</td>
<td>0.01%</td>
</tr>
<tr>
<td>TRUE</td>
<td>TRUE</td>
<td>FALSE</td>
<td>0.02%</td>
</tr>
<tr>
<td>TRUE</td>
<td>TRUE</td>
<td>TRUE</td>
<td>99.97%</td>
</tr>
</tbody>
</table>

Table 1: Percentages of conversions with respect whether advertisers’ site visit (a retargeting) event occurred, and if it occurred before the conversion or not.

- A simple rule based system can achieve Recall of 99.97% (on this retail advertiser example)
- Thus a few retargeting events can dominate over many other useful events
- Particularly noticeable for retail advertisers audiences
Proposed solution

The idea:
99.97% of all conversions are coming from retargeting users - observed data should be altered

Dataset generation:
For each user, generate events sequence and remove all known retargeting events up to each conversion

Modeling goals:

to design more powerful models that can capture early useful signals becomes a necessity
Data
Dataset illustrated

**Dataset**: User activities collected in a chronological order

Canonicalized and normalized activities are derived from heterogeneous sources:

- Yahoo Search,
- Yahoo and AOL Mail receipts,
- Content reads on publisher’s webpages such as Yahoo and AOL news, HuffPost, TechCrunch, Tumblr, etc.,
- Advertising data from Yahoo Gemini and Verizon Media DSP,
- Flurry mobile analytics,
- Conditional data from all advertisers (e.g., ad impressions, conversions, and advertiser site visits).

Final data product is a sequence of activities with a timestamp

Proposed Approach
Proposed approach: Deep Time-Aware conversation model DTAIN

Architecture

❖ DTAIN takes 2 sets of inputs: events and timesteps

❖ Consists of 5 blocks: embedding, recurrent, two attention and a classification block

❖ Temporal Attention captures differences between event occurrence and inference timestamp through \( \mu \) and \( \theta \) parameters
Temporal Modeling in Deep Learning

- Temporal information is most frequently modeled as a decay function, though:
  - **Stop features** [1]
    - Linear: \( v_t = \frac{t}{T} \)
    - Tanh: \( v_t = \tanh \left( \gamma \frac{t}{T} \right) + 1 - \tanh(\gamma) \)
    - Exp: \( v_t = \exp \left( \gamma \frac{t - T}{T} \right) \)
  - **Attention regularization** [2] (where \( \Delta_t \) is time gap between event and prediction time:
    \[
    a_i = \frac{\exp(v^T_i u - \lambda \Delta_t)}{\sum_t \exp(v^T_t u - \lambda \Delta_t)}
    \]
  - **Attention modeling** using the temporal signal [3, 4] by handcrafting time features \( A_j(\Delta t_i) \)
    \[
    a_i = \text{Softmax} \left( \sum_{j=1}^{k} p_{j,i} A_j(\Delta t_i) \right)
    \]

---

Temporal Modeling in Deep Learning

- **Proposed approach** is motivated by *Euler’s forward method* of solving *linear dynamic systems* [5]

\[
\Delta_t = \tau_{e_j} - \tau_{e_i} \\
\delta(e_i, \Delta_t) = S(\theta_{e_i} - \mu_{e_i} \Delta_t) \\
S(x) = \frac{1}{1 + e^{-x}}
\]

- Learns **event-specific impact** onto prediction [4]

- Single dimensional **learnable parameters**:
  - theta is the **initial impact** of the event
  - mu is **temporal change** of the event
  - Final impact of the event is **scaled** to 0-1 scale using Sigmoid function

- The larger theta and the smaller mu -> the greater impact does the event have onto prediction
- The closer to 0 they are -> the smaller initial and/or temporal impact the event has

Experimental Evaluation
Experimental setup

The proposed DTAIN model was evaluated on two datasets and against 4 competitive baselines

Datasets:
1) Proprietary Verizon Media dataset of a single retail advertiser
   ○ 788,551 users in train and 196,830 in test set, downsampled to obtain ~7.5% positives
2) Public youchoose.com dataset from RecSys 2015 challenge
   ○ 1,965,359 sessions in train and 279,999 in test set, downsampled to obtain ~11% positives

Baselines:
1) CNN
2) GRU
3) GRU + Attention layer
4) GRU + Self Attention layer
Experimental results: Proprietary VerizonMedia dataset

- **Verizon Media dataset:**
  - 985,381 user sessions, 74,407 conversions
  - long-time sequences of activities
  - prediction task: to predict if a user is going to convert for the given advertiser (binary classification task)

<table>
<thead>
<tr>
<th></th>
<th>ROC AUC</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>0.8806</td>
<td>0.8110</td>
<td>0.2457</td>
<td>0.7871</td>
<td>1.0161</td>
</tr>
<tr>
<td>GRU</td>
<td>0.9018</td>
<td>0.8520</td>
<td>0.3004</td>
<td>0.7972</td>
<td>1.1983</td>
</tr>
<tr>
<td>GRU+Attn</td>
<td>0.8968</td>
<td>0.8438</td>
<td>0.2882</td>
<td>0.7982</td>
<td>0.8047</td>
</tr>
<tr>
<td>GRU+SelfAttn</td>
<td>0.8804</td>
<td>0.8364</td>
<td>0.2743</td>
<td>0.7756</td>
<td>0.9273</td>
</tr>
<tr>
<td>DTAIN</td>
<td>0.9263</td>
<td>0.8602</td>
<td>0.3219</td>
<td>0.8537</td>
<td>0.9871</td>
</tr>
</tbody>
</table>

Table 3: Performance metrics on the proprietary user trails dataset for all algorithms.

- **The proposed DTAIN model outperforms other baselines** on the conversion prediction task w.r.t. ROC AUC, Accuracy, Precision, Recall and Bias

- **Improvements** over all baselines are prominent thanks to the long-time sessions (>100 days)
Experimental results: Proprietary VerizonMedia dataset, contd.

- **Verizon Media dataset:**
  - 985,381 user sessions, 74,407 conversions
  - **long-time** sequences of activities
  - **prediction task**: to predict if a user is going to convert for the different conversion rules given by the advertiser (multi-class classification task)

- Due to class disbalance that occurs when splitting the binary into multi-classification task we report PRC-AUC

- The proposed DTAIN model outperforms other baselines on the majority of metrics

<table>
<thead>
<tr>
<th>Task 0</th>
<th>PRC AUC</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>0.9880</td>
<td>0.8139</td>
<td>0.9810</td>
<td>0.8153</td>
<td>1.0069</td>
</tr>
<tr>
<td>GRU</td>
<td>0.9896</td>
<td>0.8544</td>
<td>0.9821</td>
<td>0.8588</td>
<td>1.0030</td>
</tr>
<tr>
<td>GRU+Attn</td>
<td>0.9907</td>
<td>0.8511</td>
<td>0.9837</td>
<td>0.8537</td>
<td>0.9933</td>
</tr>
<tr>
<td>GRU+SelfAttn</td>
<td>0.9877</td>
<td>0.8456</td>
<td>0.9795</td>
<td>0.8515</td>
<td>0.9941</td>
</tr>
<tr>
<td>DTAIN</td>
<td>0.9926</td>
<td>0.8613</td>
<td>0.9876</td>
<td>0.8614</td>
<td>0.9982</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 1</th>
<th>PRC AUC</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>0.2523</td>
<td>0.9602</td>
<td>0.3161</td>
<td>0.2506</td>
<td>0.8836</td>
</tr>
<tr>
<td>GRU</td>
<td>0.2711</td>
<td>0.9629</td>
<td>0.3635</td>
<td>0.2720</td>
<td>0.9715</td>
</tr>
<tr>
<td>GRU+Attn</td>
<td>0.3013</td>
<td>0.9630</td>
<td>0.3788</td>
<td>0.3139</td>
<td>1.1163</td>
</tr>
<tr>
<td>GRU+SelfAttn</td>
<td>0.2452</td>
<td>0.9606</td>
<td>0.3277</td>
<td>0.2648</td>
<td>1.0645</td>
</tr>
<tr>
<td>DTAIN</td>
<td>0.2880</td>
<td>0.9652</td>
<td>0.4000</td>
<td>0.2539</td>
<td>1.0680</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Task 2</th>
<th>PRC AUC</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
<th>Bias</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>0.2495</td>
<td>0.9584</td>
<td>0.3287</td>
<td>0.2419</td>
<td>0.7588</td>
</tr>
<tr>
<td>GRU</td>
<td>0.2567</td>
<td>0.9597</td>
<td>0.3485</td>
<td>0.2464</td>
<td>0.8849</td>
</tr>
<tr>
<td>GRU+Attn</td>
<td>0.2374</td>
<td>0.9584</td>
<td>0.3355</td>
<td>0.2582</td>
<td>1.0453</td>
</tr>
<tr>
<td>GRU+SelfAttn</td>
<td>0.2081</td>
<td>0.9587</td>
<td>0.3081</td>
<td>0.1951</td>
<td>0.9887</td>
</tr>
<tr>
<td>DTAIN</td>
<td>0.2776</td>
<td>0.9633</td>
<td>0.4083</td>
<td>0.2348</td>
<td>0.9460</td>
</tr>
</tbody>
</table>

Table 4: Performance metrics on the proprietary user trails dataset for different tasks.
Interpretability analysis of the DTAIN model

On a dataset with 500 conversions and 500 last events in each trail we analyze attentions:

Figures (a) and (b) display attentions of GRU+Attn and DTAIN model:
- GRU+Attn looks on events mostly in the latter half
- DTAIN shows interesting pattern where it only focuses to last few events.

Analyzing temporal attention signals for theta (c) and mu (d):
- events both near and far from conversion are exploited

We suspect that the temporal-attention has captured the impacts of each event thus by biRNN modeling the information was compressed in last few event positions.
Experimental results: Public RecSys 2015 challenge dataset

- **Youchoose.com dataset:**
  - 2,245,358 sessions, 241,887 buys
  - **short-time** sequences of activities
  - **prediction task:** to predict if a session is going to end in purchase (binary classification task)

<table>
<thead>
<tr>
<th>Model</th>
<th>ROC AUC</th>
<th>PRC AUC</th>
<th>Accuracy</th>
<th>Precision</th>
<th>Recall</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN</td>
<td>0.7534</td>
<td>0.2870</td>
<td>0.6779</td>
<td>0.2087</td>
<td>0.7041</td>
</tr>
<tr>
<td>GRU</td>
<td>0.7504</td>
<td>0.2725</td>
<td>0.6958</td>
<td>0.2142</td>
<td>0.6746</td>
</tr>
<tr>
<td>GRU+SelfAttn</td>
<td>0.7029</td>
<td>0.2391</td>
<td>0.6734</td>
<td>0.1907</td>
<td>0.6184</td>
</tr>
<tr>
<td>GRU+Attn</td>
<td>0.7639</td>
<td>0.2973</td>
<td>0.6997</td>
<td>0.2195</td>
<td>0.6904</td>
</tr>
<tr>
<td>DTAIN</td>
<td>0.7666</td>
<td>0.3019</td>
<td>0.6943</td>
<td>0.2186</td>
<td>0.7047</td>
</tr>
</tbody>
</table>

- The **proposed DTAIN model outperforms other baselines** on the purchase prediction task w.r.t. ROC AUC, PRC AUC and Recall and is comparable to the second best baseline w.r.t. Accuracy and Precision.
- **Improvements** over GRU + Attention model are expectedly **smaller** (short sessions)
- However, **adding temporal information helps**, as it also models **initial impact** of the events to the conversion, thus providing additional information to the classifier.
Next steps

1. Analyze different dataset generation strategies
2. Predict first occurrence of retargeting events
3. Design regularization techniques that act on events highly associated with the target
4. Extend model optimization through labeling such events as adversarial ones
References


