Causally Driven Incremental Multi Touch Attribution Using a Recurrent Neural Network

Ruihuan Du†, Yu Zhong†, Harikesh S. Nair*, Bo Cui†, Ryan Shou†

†JD.com
*JD.com and Stanford University
Thanks To

Xi Chen, PM, JD.com
Jack Lin, Sen. Dir., JD.com
Paul Yan, VP, JD.com
JD is both an eCommerce platform and a publisher of ads

1. Front facing banner ads
2. Search ads
3. SKU recommendation display ads

300+ “ad-positions”
Brands buy these ads and want us – as the publishing platform – to help them understand how they perform

**Multi Touch Attribution** Problem: How much of the purchase propensity was due to each ad/touchpoint?
Overall Goal

- Develop an MTA product that could be provided as an “add-on” for large advertisers or as a paid-service

<table>
<thead>
<tr>
<th>MTA</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Help advertisers understand which ad-inventory performs best</td>
</tr>
<tr>
<td>- Help advertisers implement better campaigns on JD</td>
</tr>
<tr>
<td>- Input to budget allocation and bidding</td>
</tr>
<tr>
<td>- Better ad pricing</td>
</tr>
</tbody>
</table>
Ad-Business is “high scale”

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of ad-positions</td>
<td>300+</td>
</tr>
<tr>
<td>Number of ad-impressions per day</td>
<td>5B</td>
</tr>
<tr>
<td>Number of advertisers</td>
<td>75K</td>
</tr>
<tr>
<td>Number of product categories</td>
<td>180</td>
</tr>
<tr>
<td>Number of brands</td>
<td>140K</td>
</tr>
<tr>
<td>Number of Users</td>
<td>300M+</td>
</tr>
</tbody>
</table>
Marginality and Efficiency

• **Marginality Principle:**
  • Credit allocated to a unit should depend only on touchpoint’s marginal contributions to total incremental benefit

• **Allocative Efficiency Principle:**
  • Credit allocations should sum up to total incremental benefit without waste

• **Question:** figure out a credit allocation policy that satisfies both marginality and efficiency
  • Difficult, because in general, sum of marginal contributions will not add up to total incremental benefit (think increasing or decreasing returns from co-operation)
Remarkable Result: The Shapley Value

- The Shapley Value is the unique allocation rule that satisfies incrementality and efficiency.
- Has a lot of theoretical appeal as a fair allocation system.
This Project

• Uses the Shapley Value as the attribution mechanism (fairness)

• Defines it appropriately to only allocate incremental benefits from advertising to underlying ad-positions (incrementality)

• Uses the data to learn marginal-effects, which form an input into the computation of the Shapley Values (data-driven)
  • Train an RNN on user-data and obtain marginal effects from this response model
What’s New

- Empirical literature on data-driven attribution (response model + credit allocation system)
  - Empirical response models following Shao and Li (2011): data-driven, simple forms of dependence, not focused on fairness + incrementality
  - Dalessandro et al. (2012) was the first to propose using the Shapley value as a credit allocation mechanism for the MTA problem. They call this “causally-motivated” attribution because of the causal interpretation associated with the “marginality” property of the Shapley Value rule. Simple forms of dependence. Yadagiri et al. (2015) presents semi-parametric extension.
  - This paper: explicit consideration of role of sequence
    - RNN (nothing new in terms of methods), but first application to MTA problem in combination with credit allocation to our knowledge. Some new aspects in specification to ad-response.
    - Shapley Value computation – allocation at ad-position-day level, plus exact aggregation. Efficient MapReduce algorithm for application at scale.

- Theory: Agarwal et al. (2009); Wilbur and Zhu (2009); Jordan et al. (2011); Hu et al. (2016); Berman (2018) propose efficient contracts
Selection & Confounding

A limitation of our approach and of all the response models cited previously, is the lack of exogenous variation in user exposure to advertising. Extant papers that have trained ad-response models on data with full or quasi-randomization have done so at smaller scale, over limited number of users and ad-types due to the cost and complexity of such randomization, and have not considered the corresponding credit-allocation problem. The approach adopted here is to include a large set of user features into the model as synthetic control, so that by including these, we convert a “selection on unobservables” problem into a “selection on observables” problem. Controlling flexibly for these observables mitigates the selection issue somewhat, albeit not Perfectly.
Overall Architecture

**Training step**

1. Train a model for ad-response on historical data

**Attribution step**

2.1. Pick all observed orders for brand on a day

2.2. Find all ad-exposures by brand to that user over past $T$ days

2.3. Use the trained model to compute marginal effects and Shapley Values to attribute order to those ad-exposures

2.4. Aggregate to desired level and report to advertiser
In paper

1. Define incremental benefit from a brand’s ads to an observed order

2. Define a Shapley Value over ad-position-day tuples that occurred before order, such that they sum to this incremental benefit

3. Develop a scalable algorithm that mixes Monte Carlo simulation and exact computation to compute Shapley Values in few hours (for daily reporting)
**Overall Architecture**

**Training step**

1. Train a model for ad-response on historical data

**Attribution step**

2.1. Pick all observed orders for brand on a day

2.2. Find all ad-exposures by brand to that user over past $T$ days

2.3. Use the trained model to compute marginal effects and Shapley Values to attribute order to those ad-exposures

2.4. Aggregate to desired level and report to advertiser
We need..

• A response model that takes the sequence into account
  • *Intensity* of ad-exposure
  • *Timing* of ad-exposure
  • *Stock effects* of ad-exposure
  • *Competitive* ad-exposure effects
  • *User heterogeneity* and (limited) selection into exposure
  • Accommodates dimension: roughly 140,000+ brands across roughly 180 categories
RNN with Hidden-Hidden Recurrence

\[ \mathcal{L}_i = \sum_{t=1}^{T} \mathcal{L}_{it} \quad \text{Overall loss = total log-likelihood} \]

\[ \tilde{Y}_{it} = \text{sigmoid}(ad_i + G h_{it}) \]

\[ h_{it} = \text{sigmoid}(B + W \cdot [x_{it}, p_t] + Vh_{i,t-1}) \]

Key parameters controlling dependence (and depth)

\[ \text{Log-likelihood} \]

\[ \text{output layer: predicted probability of purchase} \]

\[ \text{observed action: binary for purchase in } t \]

\[ \text{input layer: own & competitive ad-impressions, price index} \]
Bi-directional LSTM RNN

RNN has ~18M “parameters”
TensorFlow Code: Simulates data, trains RNN, computes Shapley Values

https://github.com/jd-ads-data/jd-mta
Benchmarks

Cell-phone ad exposure and purchase data, 15-day window, 2017, JD.com

<table>
<thead>
<tr>
<th>Benchmark</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of users in Overall Sample</td>
<td>75,768,508</td>
</tr>
<tr>
<td>Number of users in Positive Sample</td>
<td>2,100,687</td>
</tr>
<tr>
<td>Number of users in Negative Sample</td>
<td>73,667,821</td>
</tr>
<tr>
<td>Num: of ad-impessions in category</td>
<td>7,153,997,856</td>
</tr>
<tr>
<td>Num: of orders in category</td>
<td>3,477,621</td>
</tr>
<tr>
<td>Number of orders made on day $T = 15$</td>
<td>175,937</td>
</tr>
<tr>
<td>Number of brands ($B$)</td>
<td>31</td>
</tr>
<tr>
<td>Number of ad-positions ($K$)</td>
<td>301</td>
</tr>
</tbody>
</table>

*Notes:* The figures benchmark the accuracy, precision, recall and AUC (area under the curve) statistics of the bi-directional RNN, against a unidirectional LSTM RNN and a logistic model.
Interpret as: the smaller the Shapley Value at an ad-position, the more it deviates from “last-clicked” attribution

Max Shapley is 0.6 => last-clicked never contributes more than 60% incrementally

Deviations least for search

Deviation high for cart and payment
**Pooled Model**

- Number of categories in the model: 180
- Number of brands in the model: 140 thousand
- Number of impressions in 15 days: 80 billion
- Size of training dataset: 187 million
- Number of advertisers we can serve per day: 75 thousand
- Total model training time: 10 hours
- Models’ average recall: 85%~95%
- Models’ average precision: 60%~70%
- Time of Shapley value computation: 3 hours
- GPU cards: 130
https://jzt.jd.com
JD MTA分析结果展示说明

整体效果分析

三大分析模块

整体投放数据

筛选数据聚合方式

可筛选数据指标

可维度数据指标

可报告数据指标

By cross-tab
CAT benchmark

Aggregation level

GMV
ROI
CPM
CPC

Day of campaign
Thanks!