Predicting App Purchases

NewBees (Group 12)

Our Approach

- We care about a user's ***recent history*** the most
  - Include additional features that are calculated using only the most recent week(s) data

7 days:

- $Fea_{0-9} + Fea_{8-9} + Fea_{9} -> Purchase_{10}$ (train)
- $Fea_{0-10} + Fea_{9-10} + Fea_{10} -> Purchase_{11}$ (predict)

14 days:

- $Fea_{0-8} + Fea_{7-8} + Fea_{8} -> Purchase_{9-10}$ (train)
- $Fea_{0-10} + Fea_{9-10} + Fea_{10} -> Purchase_{11-12}$ (predict)

Feature Engineering

- A little over 70 features total
  - 19 shared by both models
  - 55 week-specific features
- A few niche indicator features
- Mainly numerical aggregates by user during a certain time period
- No categorical/one hot encoding features

7 Day Model	
1	Indicator for whether a user's attribute value contains a bracket '[]'
2	Number of unique attribute values by user
3	Number of events a user had between weeks 0 - 9
14 Day Model	
1	The average number of purchases a user made
2	Number of unique attribute values by user
3	Number of events a user had between weeks 0 - 9

Machine Learning Techniques

	Random Forest	Light GBM	Logistic Regression	XGBoost
PRO	Baseline model! Handle bias and variances	Fast to train	Simple to understand	Great performance
CON	Nothing we just want to explore more	Performance not consistent	Slow to train	Not yet discovered

Experimental Results

- Blending together models
- Final **AUC: 0.98672**

7 day prediction = (½ * XGBoost) + (½ * Random Forest)

14 day prediction = (⅓ * Logistic Regression) + (⅓ * Random Forest) + (⅓ * XGBoost)

Lessons Learned

- **Save memory and time**
  - Label Encoding user_id_hash : Attribute: 25G → 4G, Event:13G → 8G

- **Feature Engineering is super important**
  - Hard to discover that Indicator for ‘[]’ was the most important feature in the seven-day model

- **Blend Models Together**
  - Get better performance by combining Random Forest, LightGBM and XGBoost and assign weights on them.

Thank you