Collaborative Embedding Features and Diversified Ensemble for E-Commerce Repeat Buyer Prediction

Zhanpeng Fang*, Zhilin Yang*, Yutao Zhang
Tsinghua Univ. (* equal contribution)
Results

• Team “FAndy&kimiyoung&Neo”
• 2nd place in stage 1
• 3rd place in stage 2
• The only team marching in top 3 of both stages
Team Members

• Zhanpeng Fang
  – Master student, Tsinghua Univ. & Carnegie Mellon Univ.

• Zhilin Yang
  – Bachelor E., Tsinghua Univ.

• Yutao Zhang
  – PhD student, Tsinghua Univ.
Task

• Input:
  – User behavior logs
    • user, item, category, merchant, brand, timestamp, action
  – User profile
    • age, gender.

• Output:
  – The probability that a new buyer of a merchant is a repeat buyer
Challenges

• Heterogeneous data
  – User, merchant, category, brand, item

• Repeat buyer modeling
  – What are the characteristic features for modeling repeat buyer?

• Collaborative information
  – How to leverage the collaborative information between users and merchants [in a shared space]?
Framework

Feature Engineering

Basic features
Repeat features
Embedding features

Models

LR models
GBDT models
FM models

Ridge Regression

Ensemble
Framework

Two novel feature sets,
Repeat features &&
Embedding features

Feature Engineering

Models

Ridge Regression

Ensemble
Framework

Three individual models

Feature Engineering

Ridge Regression

Ensemble
Framework

Feature Engineering

Basic features
Repeat features
Embedding features

Diversified Ensemble

Models

LR models
GBDT models
FM models

Ridge Regression

Ensemble
Feature Engineering – Basic Features

• User-Related Features
  – Age, gender, # of different actions
  – #items/merchants/… that clicked/purchased/favored
  – Omitting add-to-cart in all actions related features increases performance (since almost identical to purchase)

• Merchant-Related Features
  – Merchant ID
  – #actions and #distinct users that clicked/purchased/favored (only in Stage 1)
Feature Engineering – Basic Features

• User-Merchant Features
  – # different actions
  – Category IDs and brand IDs of the purchased items

• Post Processing
  – Feature binning in Stage 1
  – Log(1+x) conversion in Stage 2
  – Perform similarly. Both much better than raw values.
Repeat Features

• User Repeat Features
  – Average span between any two actions
  – Average span between two purchases
  – How many days since last purchase
Repeat Features

- **Average active days** for one merchant/category/brand/item
- **Maximum active days** for one merchant/category/brand/item
- **Average span** between any two actions for one merchant/category/brand/item
- **Ratio** of merchants/categories/brands/items with repeated actions
Repeat Features

• Category/Brand/Item Repeat Features
  – **Average active days** on given category/category/brand/item of all users
  – **Ratio of repeated active users** on given category/brand/item
  – **Maximum active days** on given category/brand/item of all users
  – **Average days** of purchasing the given category/brand/item of all users
  – **Ratio** of users who purchase the given categories/brands/item more then once
  – **Maximum days** of purchasing the given category/brand/item of all users
  – **Average span** between two actions of purchasing the given category/brand/item of all users
Embedding Features

Heterogeneous interaction graph
Embedding Features

Heterogeneous interaction graph

Random walk

\[ W = \begin{bmatrix} \circ & \circ & \circ & \circ & \cdots \\ \circ & \circ & \circ & \circ & \cdots \\ \circ & \circ & \circ & \circ & \cdots \\ \circ & \circ & \circ & \circ & \cdots \\ \circ & \circ & \circ & \circ & \cdots \end{bmatrix} \]
Embedding Features

Heterogeneous interaction graph

Random walk

\[ W = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \]

Skipgram model

Embedded vectors

\[
\begin{align*}
\text{u1} & \quad \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \\
\text{u2} & \quad \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix} \\
\text{m1} & \quad \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}
\end{align*}
\]
Embedding Features: Interaction Graph

• Let the graph $G = (V, E)$
  – $V$ is the vertex set
  – $E$ is the edge set

• $V$ contains all users and merchants
• If user $u$ interacts with merchant $m$, then add an edge $<u, m>$ into $E$
Embedding Features: Random Walk

• Repeat a given number of times
  – For each vertex \( v \) in \( V \)
    • Generate a sequence of random walk starting from \( v \)
    • Append the sequence to the corpus

\[
W = \begin{bmatrix}
  \circ & \circ & \circ & \circ & \circ \\
  \circ & \circ & \circ & \circ & \circ \\
  \circ & \circ & \circ & \circ & \circ \\
  \circ & \circ & \circ & \circ & \circ \\
  \circ & \circ & \circ & \circ & \circ \\
\end{bmatrix} \quad \text{Generate random walk corpus}
\]
Embedding Features: Skipgram

Use the current word $W(j)$ to predict the context.

Objective function:

$$L = - \sum_{W \in \mathcal{W}} \frac{1}{T} \sum_{t=1}^{T} \sum_{-c \leq j \leq c, j \neq 0} (f'_{W_{t+j}}^T f_{W_t} - \sum_{w \in V} f'_w^T f_{W_t})$$

Use SGD to optimize the above objective and obtain embeddings for users and merchants.
Embedding Features: Dot Products

- Now we have embeddings of all users and merchants.
- Given a pair \( <u, m> \), we derive a feature

\[
f_u \top f_m
\]

- to represent the semantic similarity between \( u \) and \( m \).
- \( f \) means embeddings.
Embedding Features: Diversification

• Simply applying the dot product of embeddings is not powerful enough.
• Recall that we use SGD to learn the embeddings.
• We use embeddings at different iterations of SGD.
• An example
  – Run 100 iterations of SGD.
  – Read out embeddings at iteration 10, 20, …, 100.
  – Obtain a 10-dim feature vector of dot products
• Intuition: similar to ensemble models with different regularization strengths
Individual Models

• Logistic regression
  – Use the implementation of Liblinear
• Factorization machine
  – Use the implementation of LibFM
• Gradient boosted decision trees
  – Use the implementation of XGBoost

<table>
<thead>
<tr>
<th>Method</th>
<th>Implementation</th>
<th>Best AUC in Stage 1 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>Liblinear</td>
<td>69.782</td>
</tr>
<tr>
<td>Factorization Machine</td>
<td>LibFM</td>
<td>69.509</td>
</tr>
<tr>
<td>GBDT</td>
<td>XGBoost</td>
<td>69.196</td>
</tr>
</tbody>
</table>
Diversified Ensemble

Feature set

Ridge regression

Model set

Final Results
Diversified Ensemble: Appending New Features

- Feature set F0: Basic Features
- Feature set F1: Basic Features
  - Repeat Features
- Feature set F2: Basic Features
  - Repeat Features
  - Embedding Features

New Features
Diversified Ensemble: Cartesian Product

<table>
<thead>
<tr>
<th>Feature Set</th>
<th>LR</th>
<th>GBDT</th>
<th>FM</th>
</tr>
</thead>
<tbody>
<tr>
<td>F0</td>
<td>Ensemble 1</td>
<td>Ensemble 2</td>
<td>Ensemble 3</td>
</tr>
<tr>
<td>F1</td>
<td>Ensemble 4</td>
<td>Ensemble 5</td>
<td>Ensemble 6</td>
</tr>
<tr>
<td>F2</td>
<td>Ensemble 7</td>
<td>Ensemble 8</td>
<td>Ensemble 9</td>
</tr>
</tbody>
</table>
Diversified Ensemble Results

- Simple ensemble: Only ensemble the top 3 models
- Diversified ensemble outperforms simple ensemble

<table>
<thead>
<tr>
<th>Method</th>
<th>Implementation</th>
<th>Best AUC in Stage 1 (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Logistic Regression</td>
<td>Liblinear</td>
<td>69.782</td>
</tr>
<tr>
<td>Factorization Machine</td>
<td>LibFM</td>
<td>69.509</td>
</tr>
<tr>
<td>GBDT</td>
<td>XGBoost</td>
<td>69.196</td>
</tr>
<tr>
<td>Simple Ensemble</td>
<td>Sklearn Ridge</td>
<td>70.329</td>
</tr>
<tr>
<td>Diversified Ensemble</td>
<td>Sklearn Ridge</td>
<td><strong>70.476</strong></td>
</tr>
</tbody>
</table>
Factor Contribution Analysis

- Clear performance increase after adding each feature set
- Both embedding features and repeat features provide unique information to help the prediction
- The results are based on Logistic Regression

<table>
<thead>
<tr>
<th>No.</th>
<th>Feature Sets</th>
<th>Stage 1 AUC (%)</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Basic features</td>
<td>69.369</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>1 + Embedding features</td>
<td>69.495</td>
<td>0.126</td>
</tr>
<tr>
<td>3</td>
<td>2 + Repeat features</td>
<td>69.782</td>
<td>0.287</td>
</tr>
</tbody>
</table>
Stage 2 Performance

- Repeat features are consistent in both stages
- **Data cleaning** is important
  - duplicated/inconsistent records exist in this stage
- The results are based on Logistic Regression

<table>
<thead>
<tr>
<th>No.</th>
<th>Method</th>
<th>AUC (%)</th>
<th>Gain</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Basic features</td>
<td>70.346</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>1 + Repeat features</td>
<td>70.589</td>
<td>0.243</td>
</tr>
<tr>
<td>3</td>
<td>2 + Data cleaning &amp; more features</td>
<td>70.898</td>
<td>0.309</td>
</tr>
<tr>
<td>4</td>
<td>3 + Fine-tuning parameters</td>
<td>71.016</td>
<td>0.118</td>
</tr>
</tbody>
</table>
Summary

• “Tricks” on how to win top 3 in both stages
  - Diversified ensemble
  - Novel embedding features
Thank you!

Questions?