Next Basket Recommendation

Alok Kumar(190099)* & Nilay Beniwal(190555)*
Supervisor: Prof. Amit Mitra

Indian Institute of Technology Kanpur
*Bachelors of Science in Mathematics and Scientific Computing

Abstract

The goal of next basket recommendation is to recommend items for the next basket of the user based on prior baskets of the user. Most conventional models either use sequential transaction features of user or general interests of a user. Further, some consider these two matters completely different and then merge them in some way to recommend next basket. In this project we explored the DREAM (Dynamic REcurrent bAsket Model), HRM (Hierarchical Representation Model), NN (Neural Network) model and Apriori Algorithm.

1 Introduction

The real world scenarios of most e-commerce website consist of purchasing different item baskets at different time and predicting the next item-set directly impacts the company sales . Hence, next basket recommendation have received much attention. In general, there are two easily distinguishable approaches used in present world for next basket recommendation. One perspective is the collaborative filtering models, which capture users general interests but have difficulty in considering sequential features of historical transactions. Matrix factorization is a successful collaborative filtering model. second method are based on sequential features from past transactions . Markov chain model are example of this.HRM model use two layer aggregating of item vector and last transaction with user vectors so that model can capture transaction property as well as general interest.

1.1 Problem statement

We are given a dataset which contains the purchase-history of the users. Based on this purchase-history our task is to recommend the items in the next basket of the user. To solve this problem, we want develop a model which recommend next item-basket to a user.

2 Methodology:

2.1 DREAM (Dynamic REcurrent bAsket Model)

The framework of this model is summarized by figure-2. Input in DREAM is sequence of baskets corresponding to a particular user's shopping. We represent $B^u_{t_i}$ as the basket of items purchased by a user u at time t_i.

$$B^u_{t_i} = \{n^u_{t_i,j} \in R^d | j = 1, 2..., |B^u_{t_i}|\}$$

where $n^u_{t_i,j}$ is the latent representation of the j^{th} basket and $|B^u_{t_i}|$ is the cardinality of the set. We then generate latent vector representation $\mathbf{b}^u_{t_i}$ for a basket $|B^u_{t_i}|$ by aggregation (using max or average pooling).

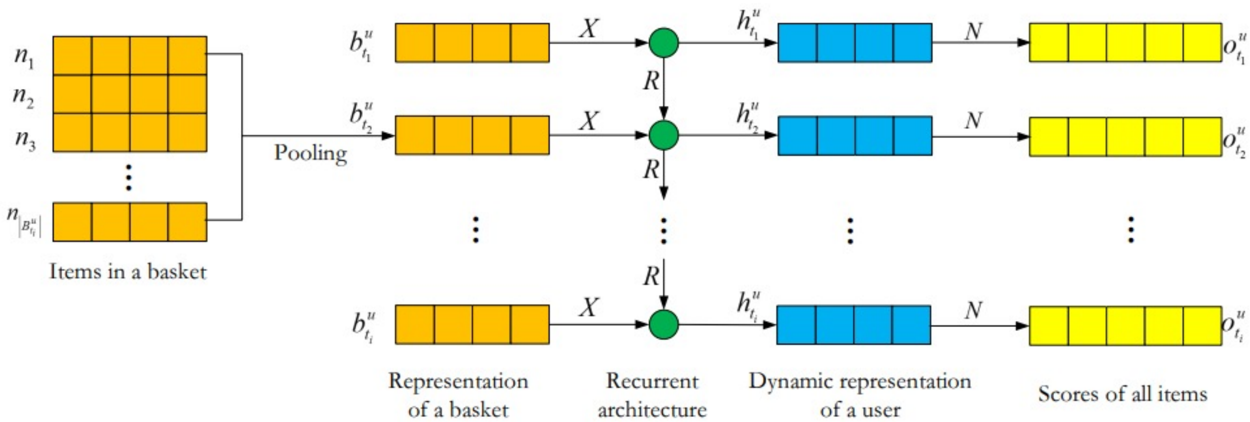

Figure 1: pictorial representation of DREAM algorithm

In max pooling, we aggregate a group of latent representation of item vecotrs by taking the maximum value of each dimension among all the vectors. Each dimension of $\mathbf{b}^u_{t_i}$ is formulated as,

$$\mathbf{b}^u_{t_i,k} = max(n^u_{t_i,1,k}, n^u_{t_i,2,k}..)$$

here, $\mathbf{b}^u_{t_i,k}$ is the k-th dimension of basket latent vector $\mathbf{b}^u_{t_i}$, $n^u_{t_i,1,k}$ means the value of k-th dimension of the vector representation of the j-th item $\mathbf{n}^u_{t_i,j}$ in the basket $B^u_{t_i}$
Average pooling is similar to max pooling but replaces max with average operation i.e.

$$\mathbf{b}^u_{t_i} = \frac{1}{|B^u_{t_i}|}\sum_{j=1}^{|B^u_{t_i}|}\mathbf{n}^u_{t_i,j}$$

we can write the vector representation of hidden layer as :

$$\mathbf{h}^u_{t_i} = f(\mathbf{X}\mathbf{b}^u_{t_i} + \mathbf{R}\mathbf{h}^u_{t_i-1})$$

where $\mathbf{R}$ helps in the transitioning of the representation of the user between purchases and how the dynamic representation $\mathbf{h}^u_{t_i}$ and $\mathbf{h}^u_t$ and $\mathbf{X}$ is a transition matrix between latent representation of baskets and $\mathbf{h}^u_{t_i-1}$ is the dynamic representation of the previous time t_{i-1}. Activation function f is choosen as *sigmoid* function, i.e. $f(x) = \frac{1}{1+e^{-x}}$.
The output $\mathbf{o}_{u,t_i}$ can be calculated through multiplication of item matrix $\mathbf{N}$ and users dynamic representation $\mathbf{h}^u_{t_i}$.

$$\mathbf{o}_{u,t_i} = \mathbf{N}^T\mathbf{h}^u_{t_i}$$

Learning Proceduere:-
DREAM incorporates Bayesian Personalized Ranking (BPR) during learning. BPR is a state-of-the-art pairwise ranking framework for the implicit feedback data. The assumption of BPR is that a user prefers an item in basket at a specific time than a negative item sample. The negative items are items which are not in the basket. In this way, we need to maximize the following probability:

$$p(u,t,v \succ v') = \sigma(o_{u,t,v} - o_{u,t,v'}),$$

where v' denotes a item never purchased and σ is taken to be sigmoid function($\sigma(x) = \frac{1}{1+e^{-x}}$)
Adding all the log likelihood and regularization term the objective function can be written as -

$$J = \sum ln(1 + e^{-(o_{u,t,v} - o_{u,t,v'})}) + \frac{\lambda}{2}||\Theta||^2$$

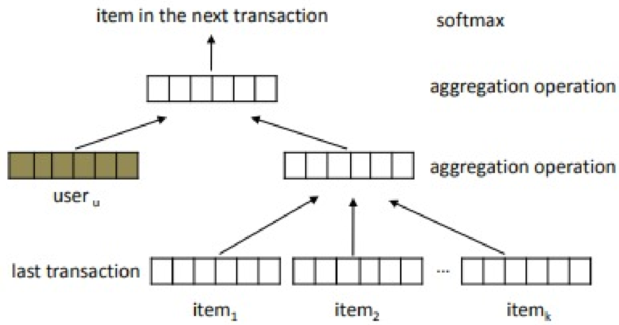

Figure 2: pictorial representation of HRM algorithm

where $\Theta = \{N, R, X\}$ denotes all the parameters to be learnt, λ is a parameter to control the power of regularization. This objective function can be optimized with back propagation through time approach.

2.2 HRM (Hierarchical Representation Model)

Formalization - $U = \{u_1, u_2, ..., u_{|U|}\}$ is set of users and $I = \{i_1, i_2, ..., i_{|I|}\}$ is the set of items. T^u denotes transaction history of a user u and T_t^u denotes transaction of user u at time t^{th} visit.
The basic idea of HRM is that it involve both sequential behaviour and users general preference and using them to predict. Defining more formally, let $V^U = \{\vec{v}_u^U \in R^n | u \in U\}$ denote the user vectors and $V^I = \{\vec{v}_i^I \in R^n | i \in I\}$ denote all the item vectors. Here V^U and V^I are model parameters to be learned by HRM.. Given a user u and two consecutive transactions T_{t-1}^u and T_t^u, probability of buying next item i given user u and his last transacton is give via softmax function:

$$p(i \in T_t^u | u, T_{t-1}^u) = \frac{exp(\vec{v}_i . \vec{v}_{u,t-1}^{hybrid})}{\sum_{j=1}^{|I|} exp(\vec{v}_j^I . \vec{v}_{u,t-1}^{Hybrid})} \tag{1}$$

where $\vec{v}_{u,t-1}^{hybrid}$ denotes the hybrid representation obtained from the hierarchical aggregation which is defined as follows-

$$\vec{v}_{u,t-1}^{hybrid} := f_2(\vec{v}_u^U, f1(\vec{v}_l^I \in T_{t-1}^u))$$

here $f_1(.)$ and $f_2(.)$ are aggregation functions at first and second layer respectively. Two common aggregating operations can be average pooling and max pooling as explained in DREAM .
Learning Proceduere:-
in learning of HRM we have to maximize the log probability defined in above condition over all users as follows

$$l_{HRM} = \sum_{u \in U} \sum_{T_t^u \in T^u} \sum_{i \in T_t^u} logp(i \in T_t^u | u, T_{t-1}^u) - \lambda ||\Theta||_F^2$$

where λ is th eregularizaton constant and Θ are model parameters(V^U, V^I).HRM defines ranking as

$$i >_{u,t} i' :\iff p(i \in T_t^u | u, T_{t-1}^u) > p(i' \in T_t^u | u, T_{t-1}^u)$$

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 dense (Dense)               (None, 500)               21000

 batch_normalization (BatchN  (None, 500)              2000
 ormalization)

 dropout (Dropout)           (None, 500)               0

 dense_1 (Dense)             (None, 250)               125250

 batch_normalization_1 (Batc  (None, 250)              1000
 hNormalization)

 dropout_1 (Dropout)         (None, 250)               0

 dense_2 (Dense)             (None, 250)               62750

 batch_normalization_2 (Batc  (None, 250)              1000
 hNormalization)

 dense_3 (Dense)             (None, 100)               25100

 batch_normalization_3 (Batc  (None, 100)              400
 hNormalization)

 dense_4 (Dense)             (None, 1)                 101

=================================================================
Total params: 238,601
Trainable params: 236,401
Non-trainable params: 2,200
_________________________________________________________________
```

Figure 3: NN model

we will dervie such ranking over whole purchase history.However, directly optimizing the above objective function is impractical so to make more efficiant optimization the objective function have been approximated with the following function

$$l_{NEG} = \sum_{u \in U} \sum_{T^u_t \in T^u} \sum_{i \in T^u_t} (log\sigma(\vec{v}^I_i . \vec{v}^{Hybrid}_{u,t-1}) + k.E_{i' \sim P_I}[log\sigma(-\vec{v}^I_{i'} . \vec{v}^{Hybrid}_{u,t-1})]) - \lambda ||\Theta||^2_F$$

where σ is sigmoid function, k is the number of negative samples and i is the sampled item, drawn according to the noise distribution P_I which is modeled by empirical unigram distribution over items.
We can then apply stochastic gradient descent algorithm for optimization.Then given a user u and their last transaction $T^u_{t_u-1}$, for each candidate item i $\in$ I, we calculate the probability $p(i \in I|_u, T^u_{u-1})$ according to Equation[1]. We than rank the items according to their probabilities, and select the top n results as the final recommended basket to the user.

2.3 Neural Network Model (NNM)

To train the neural network we normalize the data between 0 to 1. We design a neural network with one input , four hidden and an output layer. Two dropout layers are applied to prevent overfitting and 4 dense layers. Initialization of weights is done with small gaussian random number. Batch normalization is also done to each hidden layer to increase the performance . Sigmoid activation is used at output layer. Crossentropy is used for binary classification during training. Adamax optimization for gradient descent is used when the model is trained. Training is done till 200 epochs and 1000 samples per mini-batch. Five-fold validation combination are iterated and trained in the neural network model. When the test data is fed into the model, it will generate 5 predictions . Final predictioin is the average of all 5 predictions in order to reduce overfitting.

2.4 Apriori Algorithm

$\underline{SupportCount}(\sigma(\{\text{itemset}\}))$- Number number of occurrences of an itemset in the database.

$\underline{Support}$ - The fraction of transaction containing the itemset ($s(\text{itemset}) = \frac{\sigma(\text{itemset})}{|T|}$)

$\underline{Frequentitemset}$- Those items whose support is greaer than a minimum threshold .

$\underline{Confidence}$- A measure of how often B appears in the association rule (A $\implies$ B) i.e.
$C(A \implies B) = \frac{S(A,B)}{S(A)}$

$\underline{Lift}$ -(A $\implies$ $B) = \frac{support(A,B)}{support(A)*support(B)}$ which is liklihood of itemset B being sold when itemset A is sold.

$\underline{\text{Conviction}}$ - $conv(A => B) = \frac{1-S(B)}{1-C(\{A=>B\})}$

Apriori algorithm is a fast approach for getting associate rule. 2-step ARM of apriori algorithm are-
$\underline{Step1}$ - Generate all frequent itemset with support $\geq$ min support.

$\underline{Step2}$- generate associate rules wusing these frequent itemset.

Apriori algorithm follows Anti-monotonicity property i.e. any subset of a frequent itemset is frequent i.e. $\forall$ X $\subseteq$Y so Support(X) $\geq$ support (Y)$\forall$ X,Y this implies that if an itemset is not frequent then there is no need to explore its super-sets.
Algorithms for solving these two steps are given in [1], After applying the given algorithms we would get the required association rule.

3 Experiments:

3.1 Dataset:

We looked for several datasets and finally decided to choose Instacart Online Grocery Shopping Dataset.This dataset consists of over 3.4 million anonymized grocery orders from more than 206,000 users. Each user has on average 16 baskets where each basket contains 4 to 100 items.

3.2 Exploratory Data Analysis:

We sampled around .6 million orders for 75000 users. We used one hot encoding to initialize the item. Figure 4 shows order frequency of top products. Figure 5 shows the department wise distribution of items. Figure 6 and 7 shows frequency of orders per hour and per day of the week respectively. In

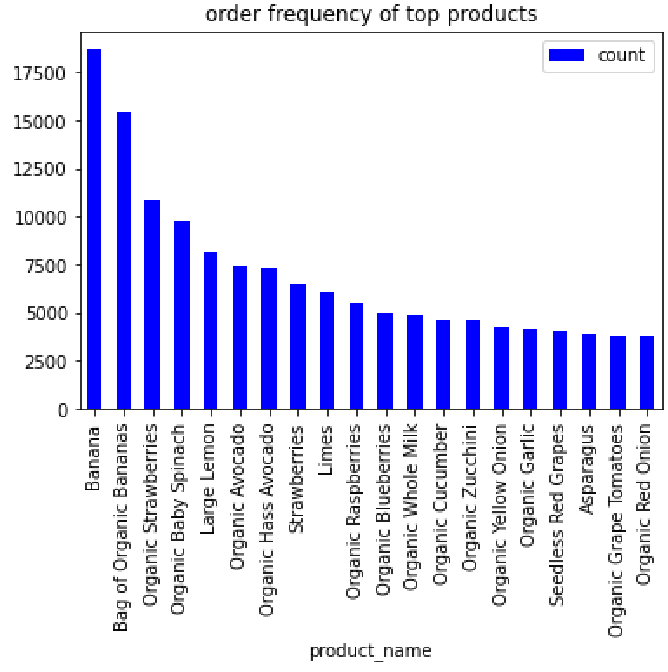

Figure 4

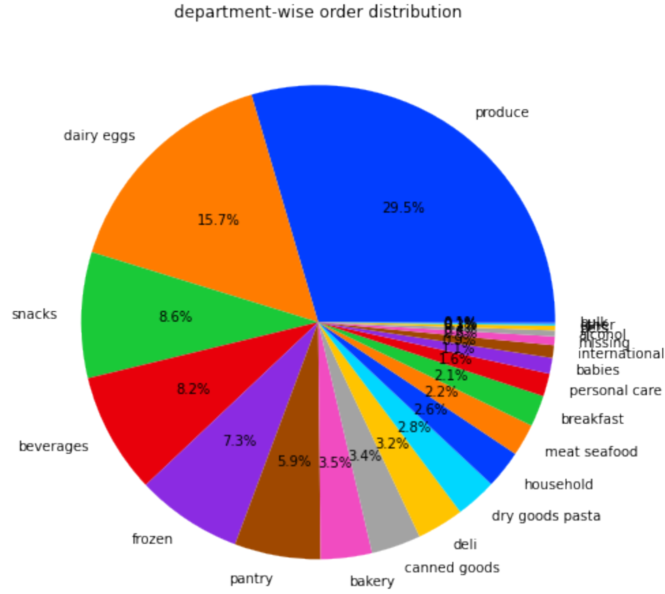

Figure 5

Neural Network model suppose the sequence of order is ranked as 1,2 ..,n-1,n.for the training data we utilize the transaction history before (n-1) th order for test data and the transaction history before n-th order for non-test users.

3.3 Results:

Figure 9 is an example of next basket recommended for a particular user using NNM. we can generate next basket recommendation for any user in such form. Figure 10 show association rules for various items which indicates the global association between two items and can be used to recommend next item if antecedent is present in user's interest.

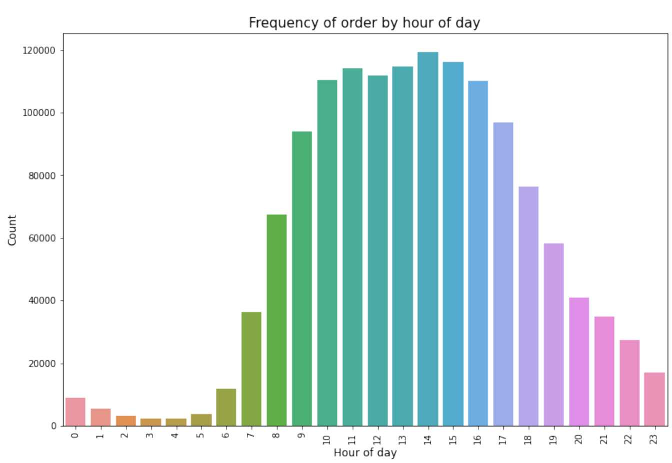

Figure 6

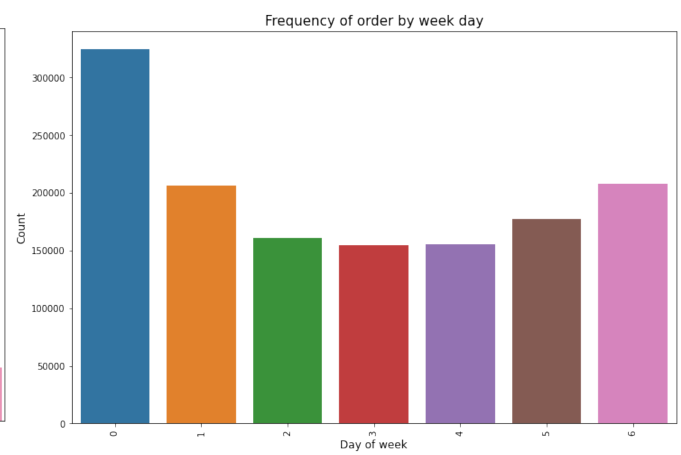

Figure 7

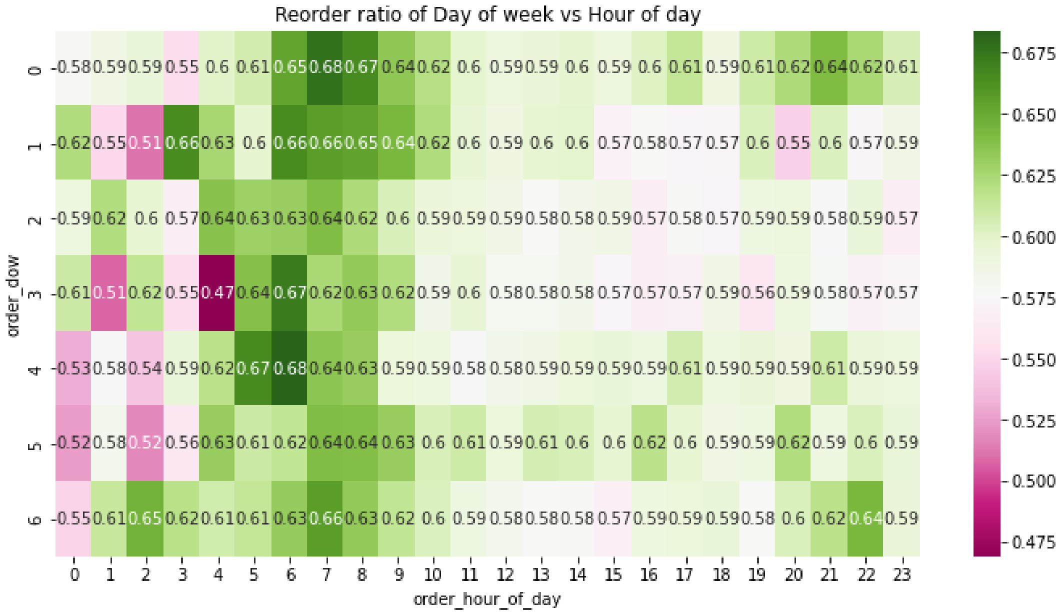

Figure 8

```
For user: 3
************************************************************
Predicted Next Basket is:
************************************************************
Electrolyte Water
Unsweetened Chocolate Almond Breeze Almond Milk
Garlic Couscous
Organic Lightly Salted Brown Rice Cakes
Granny Smith Apples
Organic Baby Spinach
Crackers, Crispy, Cheddar
Vanilla Unsweetened Almond Milk
Organic Avocado
Organic Peeled Whole Baby Carrots
************************************************************
```

Figure 9: Next Basket Recommended by NN model

	antecedents	consequents	antecedent support	consequent support	support	confidence	lift	leverage	conviction
19	(Limes)	(Large Lemon)	0.045115	0.063111	0.010325	0.228862	3.626315	0.007478	1.214942
18	(Large Lemon)	(Limes)	0.063111	0.045115	0.010325	0.163600	3.626315	0.007478	1.141661
30	(Organic Strawberries)	(Organic Raspberries)	0.091668	0.045594	0.013074	0.142625	3.128143	0.008895	1.113172
31	(Organic Raspberries)	(Organic Strawberries)	0.045594	0.091668	0.013074	0.286749	3.128143	0.008895	1.273511
21	(Organic Avocado)	(Large Lemon)	0.066340	0.063111	0.010293	0.155156	2.458445	0.006106	1.108948
20	(Large Lemon)	(Organic Avocado)	0.063111	0.066340	0.010293	0.163093	2.458445	0.006106	1.115608
3	(Organic Hass Avocado)	(Bag of Organic Bananas)	0.064379	0.142376	0.021449	0.333168	2.340054	0.012283	1.286117
2	(Bag of Organic Bananas)	(Organic Hass Avocado)	0.142376	0.064379	0.021449	0.150651	2.340054	0.012283	1.101574
4	(Bag of Organic Bananas)	(Organic Raspberries)	0.142376	0.045594	0.014811	0.104026	2.281578	0.008319	1.065217
5	(Organic Raspberries)	(Bag of Organic Bananas)	0.045594	0.142376	0.014811	0.324842	2.281578	0.008319	1.270257
29	(Organic Hass Avocado)	(Organic Strawberries)	0.064379	0.091668	0.012637	0.196293	2.141353	0.006736	1.130178

Figure 10: Association Rules given by Apriori Algorithm

4 Conclusions:

In this project we tried to solve the problem of next basket recommendation by various techniques. We solved the problem of predicting the next basket for a user using the purchase-history of user orders. We explored state-of the art techniques like DREAM and HRM which gives the best result in next basket recommendation. DREAM performs slightly better than HRM model .We also applied our own neural network model for the problem statement and out approach is able to recommend products which a user might want to buy again. We did Associate rule mining with apriory algorithm method on the dataset to recommend users an item b if ($a \implies b$) is a strong association rule and a is present in the basktet of that user.

5 Acknowledgement:

We would like to extend our gratitude towards our course instructor Prof. Amit Mitra for giving us an opportunity to work on this project. Working on a real-life problem in the form of next basket recommendation further strengthened the theoretical concepts we developed during the course, and broadened our perspective towards analyzing and possibly solving similar problems encountered in our day-to-day lives.

6 References:

1. *Agrawal, R., Srikant, R., et al.: Fast algorithms for mining association rules. In: Proc. 20th int. conf. very large data bases, VLDB. vol. 1215, pp. 487–499. Citeseer (1994)*

2. *S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme. Bpr: Bayesian personalized ranking from implicit feedback. In UAI, pages 452–461, 2009.*

3. *Yu F, Liu Q, Wu S, Wang L, Tan T (2016) A Dynamic Recurrent Model for Next Basket Recommendation. In: SIGIR 2016 - Proceedings of the 39th International ACM SIGIR Conference on Research and Development in Information Retrieval*

4. *Pengfei Wang, Jiafeng Guo, Yanyan Lan, Jun Xu, Shengxian Wan, Xueqi Cheng Learning Hierarchical Representation Model for Next Basket Recommendation CAS Key Lab of Network Data Science and Technology Institute of Computing Technology, Chinese Academy of Sciences*

5. *https://www.kaggle.com/competitions/instacart-market-basket-analysis/data*