ファッション業界向けセマンティクスとコンテキストアウェアハイブリッド推薦システム

Semantics and Context Aware Hybrid Recommender System for Fashion Industry

スタートトゥデイ研究所プロジェクトマネージャー　ラジャム・シドハント

Start Today Research Group, Project Manager, Rajam Sidhant (Mr.)

Start Today Technologies Inc. 1/2

Start Today Technologies Inc.

start today technologies

The mission of Start Today Technologies Inc. is to operation services and to develop technology that will help improve the entire Start Today Group. Start Today Technologies is a collection of those who are currently involved in technical productions, such as engineers, designers and analysts of the Start Today Group companies.

Start Today Technologies will also be in charge of operating "Start Today Research", which is our research institute. By applying the Strat Today Technologies' expertise in innovation, we aim to achieve Start Today Research's mission of "Scientifically defining and predicting fashion".

ZOZOTOWN

http://zozo.jp

WEAR

http://wear.jp

Start Today Technologies Inc. 2/2

ZOZOTOWN

http://zozo.jp

Transaction value

212 billion yen

ZOZOTOWN active members

4.59 million

Monthly number of shipments

22 million

Number of orders per second during special / seasonal sales

120 cases

Orders by device

80% **20%**

Table of Contents

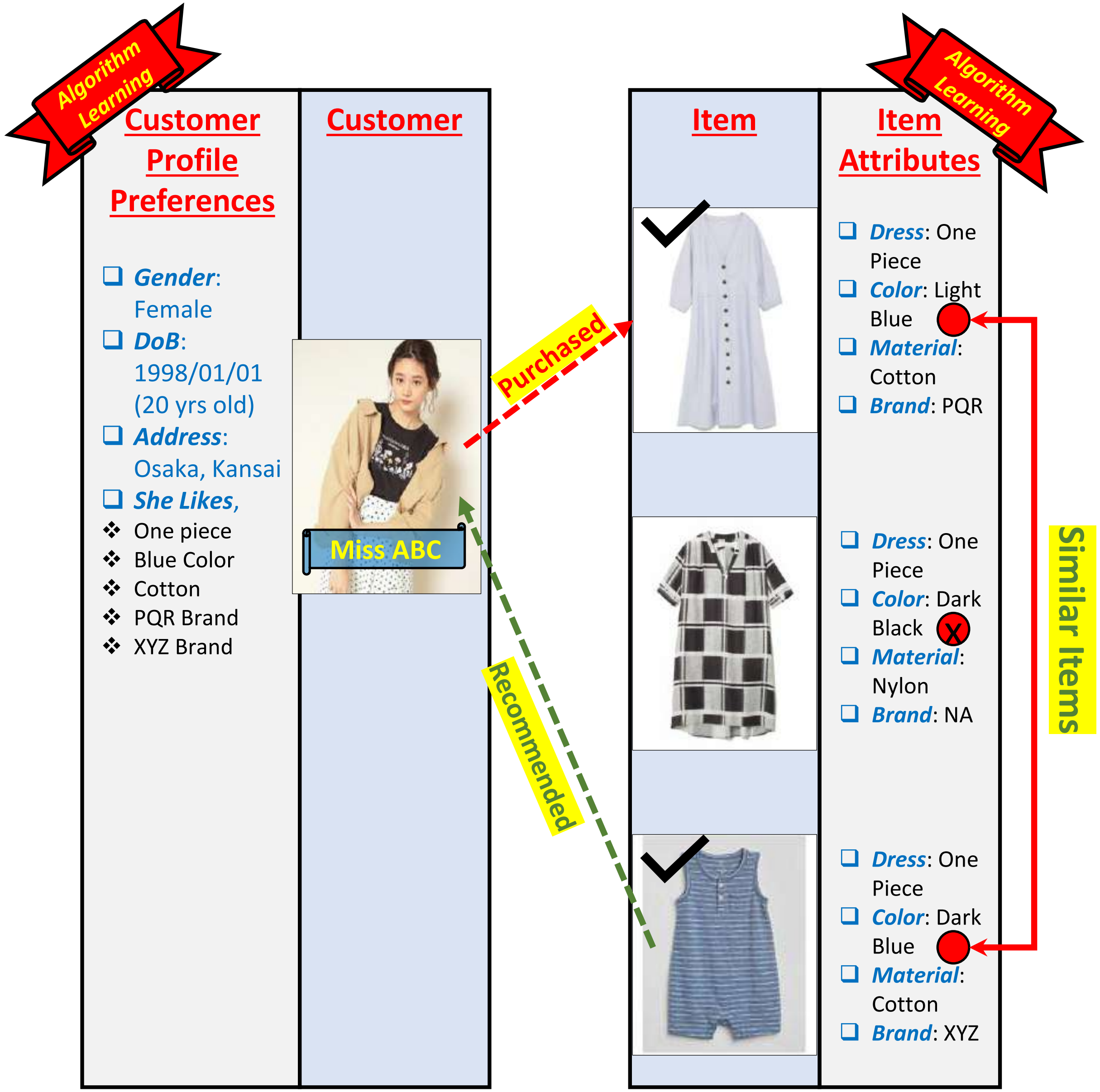
ZOZOTOWN

http://zozo.jp

WEAR

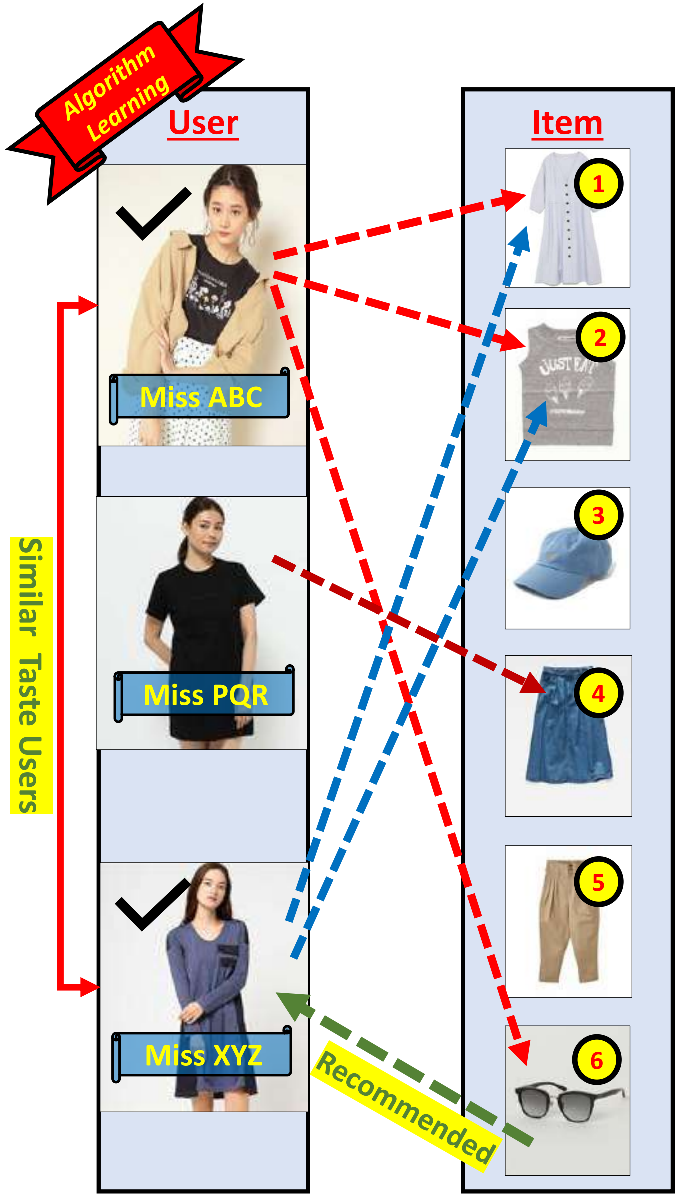
http://wear.jp

Recommendation → Content Based Filtering: Item Recommendation 1/3



- Content Based Filtering (Cognitive Filtering) Recommendations
  - Item's Contents and Features
  - Target Customer's profile and interests
- Content Based Filtering provides recommendations to customer based on,
  - Items with "Similar" contents w.r.t. target Customer's profile interests and past history.
  - Matches item features with customer's preferences and interests.
  - Computes Item's Similarity Matrix based on item's various features such as,
    - Type of dress (One piece, under garments, shorts etc.)
    - Color
    - Material used
    - Stock Keeping Unit (SKU)
    - Category
    - Brand Name
    - Review Ratings
  - Continuously updates the statistics of Item-Content-Matrix (ITM) whenever,
    - New item is added
    - New Customer is registered or Existing customer's profile interests are updated
- Learning Algorithms used to calculate ITM matrix are,
  - Vector Space Model
  - Latent Semantic Indexing
- Merit → Very efficient with less user data and newly added items.
- De-Merit → Over Specialization of item recommendations

Recommendation → Collaborative Filtering: User based Recommendation 2/3

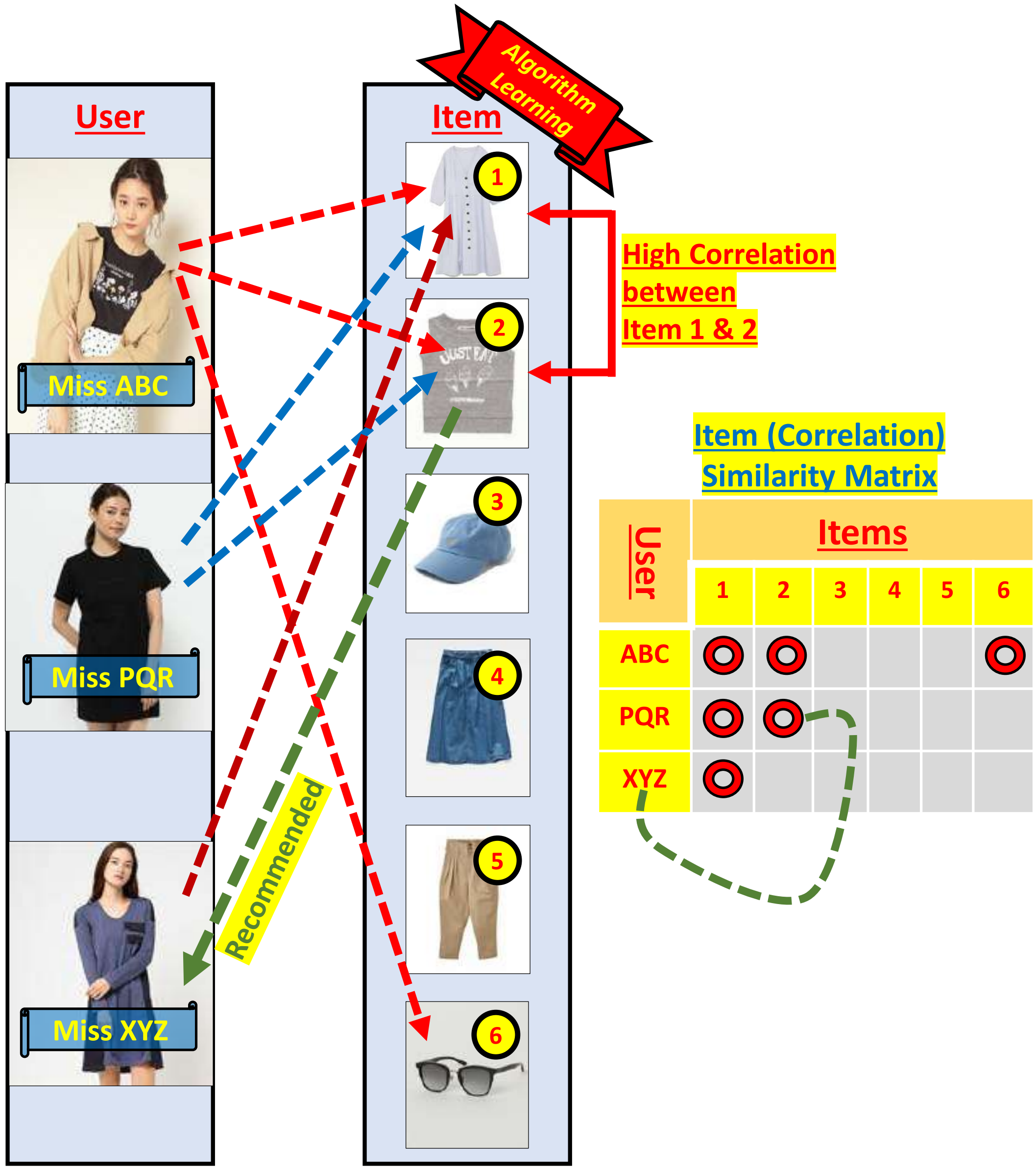


User Similarity Matrix

User	Items 1	2	3	4	5	6
ABC	O	O				O
PQR				O		
XYZ	O	O				

- ❑ Collaborative Filtering → finding other customers who share appreciations
  - ❑ If customers have somewhat same rated items in common, then they have similar taste.
  - ❑ Such customers are further classified in a group called neighborhood.
  - ❑ Customer gets recommendations for those items which customer has not rated before however those items are rated by other customers in the neighborhood.
- ❑ User-based collaborative filtering is social
  - ❑ It takes a "People First" approach, based on common interests
  - ❑ It builds the statistics of "Similarity Matrix" between customers.
  - ❑ It does not require availability of domain knowledge before system deployment
  - ❑ It does not require structured product description
- ❑ How does User-based CF Recommendation works,
  - ❑ Firstly, for a given customer, find other people who have similar tastes (neighborhood)
  - ❑ Secondly, recommend items based on past behavior of those customers.
- ❑ Learning Algorithms used to calculate "Similarity Matrix" for User-based CF are,
  - ❑ Jaccard Index (Tanimoto Similarity Coefficient)
  - ❑ Euclidean Distance Similarity
  - ❑ Log Likelihood Ratio Similarity Algorithms
  - ❑ Pearson Correlation Coefficient Similarity
- ❑ Challenges in User-based Collaborative Filtering (Research Problems)
  - ❑ Cold Start Problem → when there is very less data.
  - ❑ Scalability and performance of Sparse Matrix → when total customers are growing rapidly.
  - ❑ Shilling Attacks → Diversity → Long tail

Recommendation ➔ Collaborative Filtering: Item based Recommendation 3/3



User	Items 1	2	3	4	5	6
ABC	○	○				○
PQR	○	○				
XYZ	○					

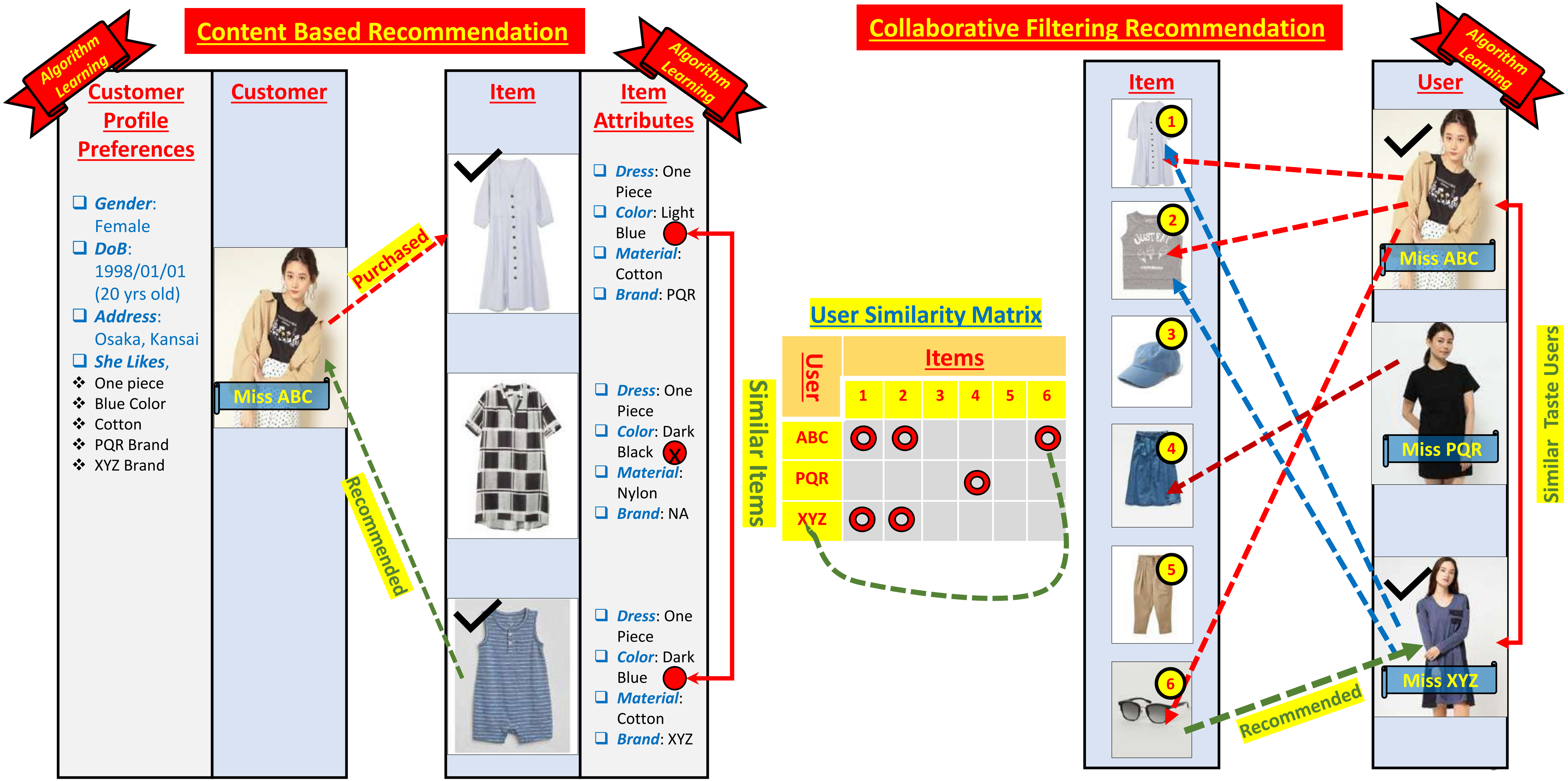
- ❑ Item based Collaborative Filtering Recommendation
  - ❑ Initially published by Amazon.com in 1998
  - ❑ Item-Item correlation model uses rating distributions per item and not per user.
  - ❑ Highly suitable for a business model where total number of users are more than items.
- ❑ How does Item-based CF Recommendation works,
  - ❑ Firstly, system builds item similarity matrix between all pairs of items with high correlation
  - ❑ Secondly, recommend items which are rated highly by others but not yet by given customer.
  - ❑ It's main purpose is to recommend other category items that customer has not yet tried.
- ❑ Item based CF matrix does not scale suddenly as compared to User based CF matrix. Hence, performance of Item based CF is better.
- ❑ Item neighborhood is fairly static and therefore it enables precomputation of matrix which improves online performance.
- ❑ Learning Algorithms used to calculate "Similarity Matrix" for Item-based CF are,
  - ❑ Slop One Algorithm suit (Already implemented in Apache Mahout libraries)
  - ❑ Cosine based Similarity
  - ❑ Adjusted Cosine Similarity
  - ❑ Pearson (Correlation) based Similarity
- ❑ Challenges in Item-based Collaborative Filtering (Research Problems),
  - ❑ Total number of users are less and items are more
  - ❑ Sparsity of Item based Dataset
  - ❑ Shilling Attacks → Diversity → Long tail
  - ❑ Time complexity can increase exponentially if both users and items are increasing rapidly

Why Hybrid Recommendation → ZOZOTOWN Big Data

❑ZOZOTOWN → contains Big Data

- ❑ [> 23million customers] + [> 30million brand product data] + [> 100million purchase history data]
- ❑ Hence, ZOZO has a rich data of user purchase history, item click rate and impression history, item rating etc.
- ❑ However, ZOZOTOWN has lot of new and newly introduced items.
- ❑ Newly introduced items do not have any history of purchase or click through rate etc.
- ❑ Likewise, newly registered members also do not have any history of purchase or clicks
- ❑ This leads to the problem of (directly/indirectly)
    - ❑ Cold Start
    - ❑ Sparsity of Item based Dataset
    - ❑ Shilling Attacks → Diversity → Long tail
- ❑ Therefore, it is recommended to use the hybrid approach by compositely applying the learning algorithms of Content Based and Collaborative Filtering.
- ❑ In this way, hybrid approach helps to quickly sell the newly added item inventory.

Hybrid Recommendation → Composite Learning Algorithms 1/2



Hybrid Recommendation → Composite Learning Algorithms 2/2

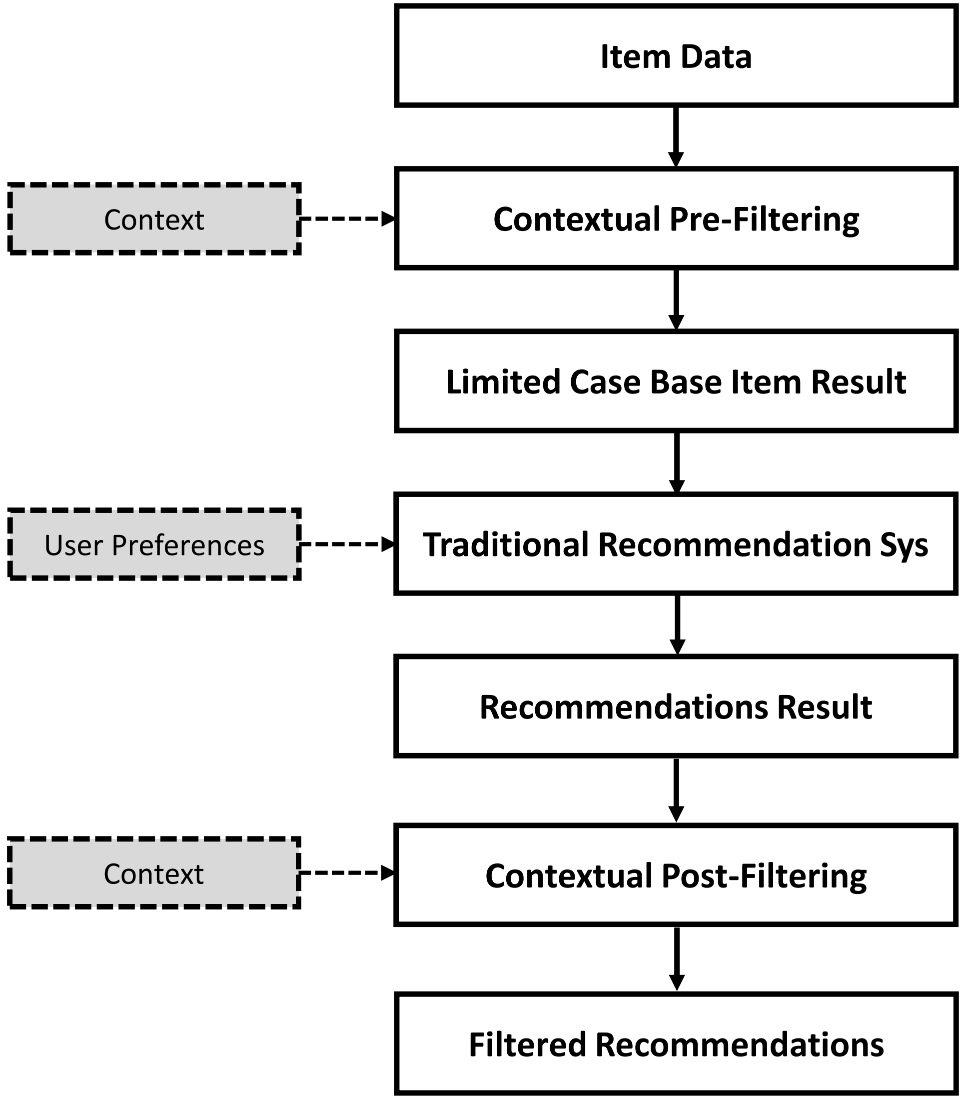
- ❑ Hybrid Learning Algorithm → approach
  1. Predict Item Ranking and Rating → Using Content Based Learning Algorithm
  2. Predict Item Ranking and Rating → Using Collaborative Filtering Learning Algorithm
  3. Create Hybrid Prediction → Using above two values
- ❑ Learning Algorithm → Learning with respect to
  1. Item's Frequency / Recency / Monetary Importance
  2. Item's Purchase History
  3. Item's Click History: Click Through Rate, Impression Rate
  4. Item's Ratings
  5. Item's Sales Importance: Paid sales (Advertisement) / Cross / Up / Down / Next Sale)
- ❑ Learning Algorithms → Main Action Items
  - ❑ Remove Biased Terms from each Item
  - ❑ Interpolating between an estimate computed from data (LABEL) and a predetermined value
  - ❑ Apply Classification and Regression for learned items
  - ❑ Train separate predictor for each item
  - ❑ K Nearest Neighbor Method
  - ❑ Compute Item/User Similarity Matrix

Context Aware Recommendation

- ❑ Data record is defined as {**user**, **item**, **rating**, **context**}
- ❑ Context → Any information used to characterize the situation of an entity (item/user) → semantically describes situation
- ❑ Context → can be
  - ❑ Paid Recommendations (Advertisement, Sponsored Events, Banners, Searches, Cross/UP/Down Sale)
  - ❑ ***Special Sales Events***
  - ❑ Location (Home / Outdoor / Leisure)
  - ❑ ***Time (for Time Sale)***
  - ❑ ***Occasion (Festival / Long Vacations / Social Gatherings)***
  - ❑ ***Season (spring/summer/autumn/winter)***
  - ❑ Weather
- ❑ Items viewed/clicked/purchased at SIMILAR CONTEXT → tend to have similar meaning → high Semantic Relatedness
- ❑ Why Context Aware Recommendation System (CARS) →
  - ❑ ZOZOTOWN conducts many Special Sales Events
  - ❑ ZOZOTOWN also conducts Time Sale Events
  - ❑ Fashion Outfits vary depending on above contextual situation
  - ❑ **[> 23million customers]** + **[> 30million brand product data]** + **[> 100million purchase history data]**
  - ❑ Customer shopping experience and behavior → changes with respect to contextual situation
  - ❑ Context-Aware Recommendation System (CARS) → helps us to read the customer's mind.

Traditional RecSys vs Context Aware RecSys

- Traditional RecSys only considers two entities → **{ User , Item }**
- Recommendation System – Main Tasks
  - Rating Prediction as per Scoring
  - Suggesting Top-N Recommendation
- On the contrary, Context Aware Recommendation System (CARS)
  - Customer's Purchase Decision Making →
    - Depends on Rational and Contextual factors
  - Therefore, to make best recommendations →
    - Necessary to read the mind of the customer
  - Context are external factors →
    - may vary when similar actions are done again and again
  - E.g. Context such as Time, Location, Occasion, Season
- Multi-Dimensional Context Aware Data Sets →
  - **{ User , Item , Context }**
- Two main pre-requisite for Context Aware Algorithm as following
  - Context Filtering → When to consider context data for Top-N Recommendation
  - Context Modelling



Implementation Design

- ❑ Machine Learning Libraries Used →
  - ❑ Apache Mahout
  - ❑ CARSKIT
  - ❑ LIBREC
  - ❑ Customized Algorithms

Retrofit & Rollout on-going

Amazon SageMaker + TensorFlow

Generic Interfaces for Customized Algorithms → Recommender | Iterative Recommender | Context Aware Recommender

Data

DB Platform	Data Structure	Data Processor
Redis KVS	Sparse Matrix	Data Access Object
Flat Files	Dense Matrix	Data Splitter & Preprocessor
Amazon S3	Sparse Vector	Data Object Serializer
Google BigQuery	Dense Vector	Data Object Deserializer
RDBMS	Sparse Tensor	

Library and Customized Algorithms

Traditional – Baseline Algorithms			Context Aware Recommendation System		
Averages	**Collaborative Filtering**	**Ranking & Scoring**	**Transformation**	**Adaption**	
UserAverage	UserKNN	BPR	ItemSplitting	Independent	Dependent
ItemAverage	ItemKNN	SLIM	UserSplitting	TF	CAMF_CU
ContextAverage	BiasedMF	LRMF	ContextSplitting	…..	CAMF_CI
….	….	…..	…..	…..	CSLIM_CI
….	….	…..	…..	…..	…..

Apache MahOut

Multi-Dimensional Context Aware Data Set

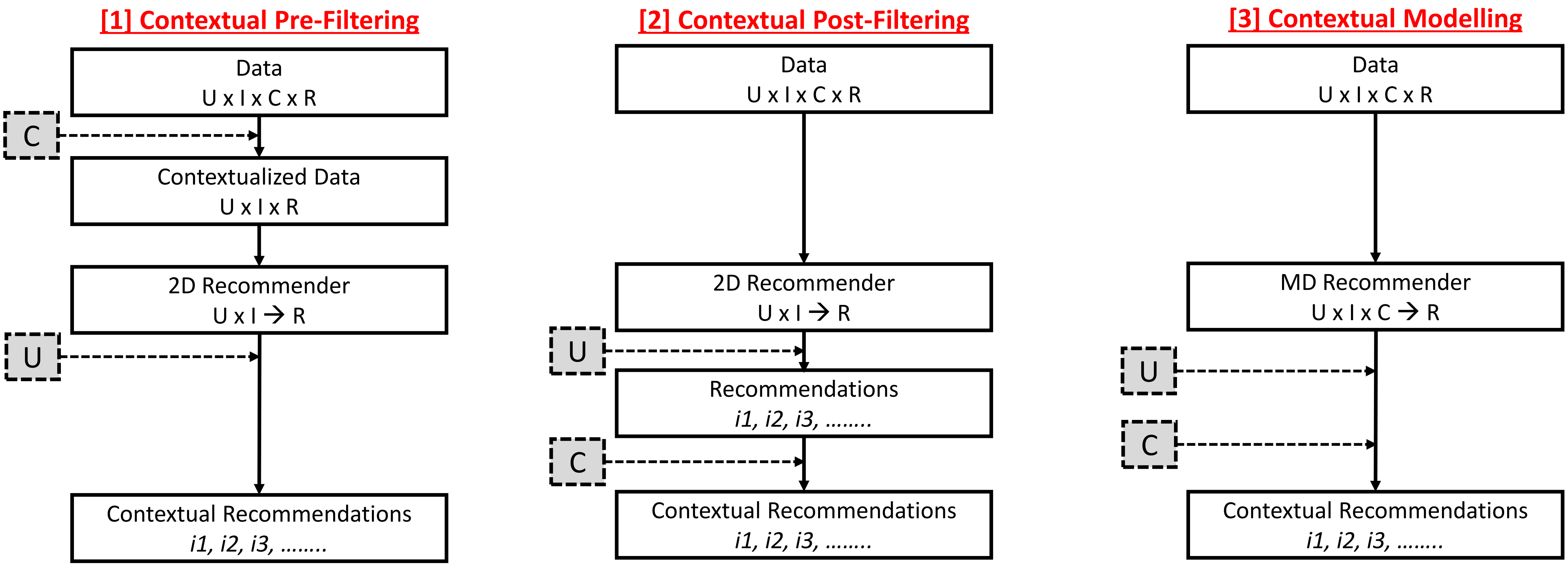
Traditional Rec Sys | **Context Aware Rec Sys – Additional Part**

User ID	Item ID	Rating	Time	Event	Season	Device
Usr1001	Pant_Jeans_01	1	Weekday	Regular	Spring	PC
Usr2002	Shirt_Denim_02	5	Weekday	ZOZO-Day	Autumn	Smart Phone (SP)
Usr3003	Skirt_Formal_01	3	Weekend	ZOZO-Day	Summer	PC
Usr4004	MenBelt_06	2	Weekend	ZOZO-Day	Spring	SP
usr5005	Shirt_Denim_04	4	Weekday	Regular	Summer	SP

- **Types or Dimensions of Context** → [Time] , [Event] , [Season] , [Device]
- **Condition of Context** → [Weekday / Weekend] , [Regular / ZOZO-Day] , [Spring / Autumn / Summer] , [PC / SP]
- **Situation of Context** → [Weekday + Regular + Spring + SP] , [Weekend + ZOZO-Day + Summer + PC]
- **Rating = (*#clicks + #bookmarks + #purchase_history*)** *(Range of Rating is between 1 to 5)*

Context Aware Algorithm → Pre-Requisite

- ❑ Two main pre-requisite for Context Aware Algorithm as following
  - ❑ Context Filtering → When to consider context data for Top-N Recommendation
  - ❑ Context Modelling → How to store context data
- ❑ CARS algorithm can be built by using any three of following methods:
- ❑ *Machine Learning on Contextualized Data → fairness in Training and Testing Matrix →* ***k-Fold Cross Validation***



Context Modelling → Differential Approach

Following three main steps:

- **Context Matching** → Match with only profiles given in {Weekday, ZOZO-Day, Autumn, SP}
- **Context Relaxation** → Use a subset of context type or dimensions to match
- **Context Weighting** → Scan through all profiles, but weighted by context similarity
- Applying and Tweaking the Following Algorithms
  - User K-Nearest Neighbor (UserKNN) (Euclidean Distance)
  - Item K-Nearest Neighbor (ItemKNN)
  - Matrix Factorization

Traditional Rec Sys | **Context Aware Rec Sys – Additional Part**

User ID	Item ID	Rating	Time	Event	Season	Device
Usr1001	Pant_Jeans_01	1	Weekday	Regular	Spring	PC
Usr2002	Shirt_Denim_02	5	Weekday	ZOZO-Day	Autumn	Smart Phone (SP)
Usr3003	Skirt_Formal_01	3	Weekend	ZOZO-Day	Summer	PC
Usr4004	MenBelt_06	2	Weekend	ZOZO-Day	Spring	SP
usr5005	Shirt_Denim_04	4	Weekday	Regular	Summer	SP

Context Modelling → Context Weighting

- Similarity of contexts is measured by ***Weighted Tanimoto Coefficient (Jaccard Index) Similarity Algorithm***

$$\text{tanimoto}(c, d, \boldsymbol{\sigma}) = \frac{\boldsymbol{\Sigma c \cap d}}{\boldsymbol{\Sigma c \cup d}}$$

- c and d are any two context rows as shown in the following table
- **σ** is the weighted vector {w1, w2, w3} for all the context dimensions
- If we assume that they are of equal weight, i.e., w1 = w2 = w3 = 1
- tanimoto (c, d, **σ**) = (total number of matched context dimensions) / (total number of all matched context dimensions)

- Other Similarity Matching and Matrix Algorithms are as following
  - Pearson's Correlation Similarity (PCC)
  - Constrained Perason's Correlation (CPC)
  - Cosine Similarity (COS)
  - Mean Squared Differences (MSD)

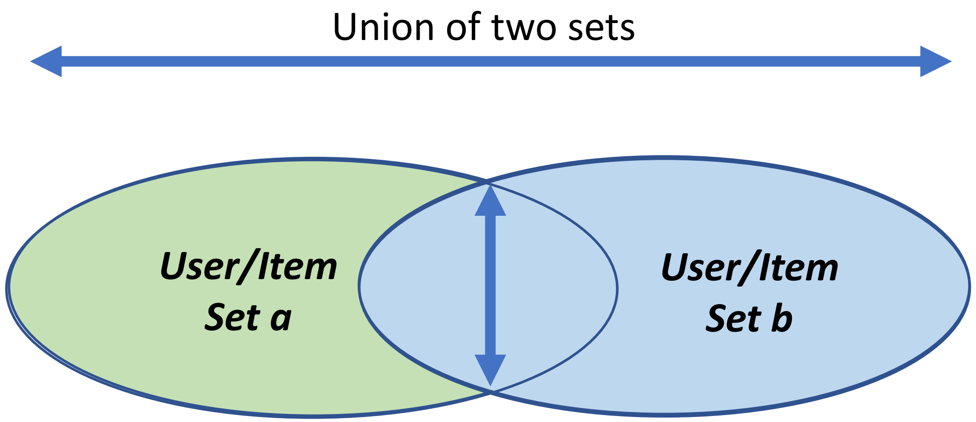
Solution → Jaccard Coefficient (Tanimoto Coefficient)

```
{
def tanimoto (set_a, set_b) :
        intersection = set_a.intersection (set_b)

        len_a = len (set_a)
        len_b = len (set_b)
        len_i = len (intersection)

        return float (len_i) / (len_a  +  len_b  -  len_i)
}
```

[Output value or score ranges between 0.0 and 1.0]



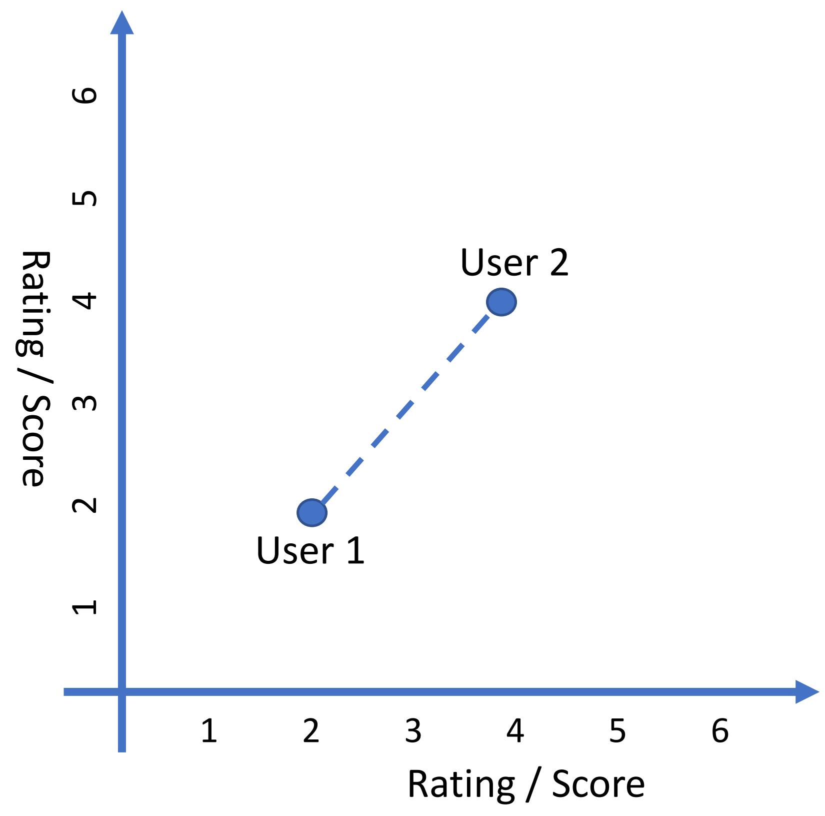
Solution → Euclidean Distance (correlation)

```
{
def euclidean (list_a, list_b) :
        dist = 0.0

        for I in range (len (list_a)):
                rate_a = list_a[i]
                rate_b = list_b[i]

                dist = dist  +  pow ((rate_a  -  rate_b),  2)
        return  sqrt (dist)
}
```

[Output value or score ranges between 0.0 and 1.0]

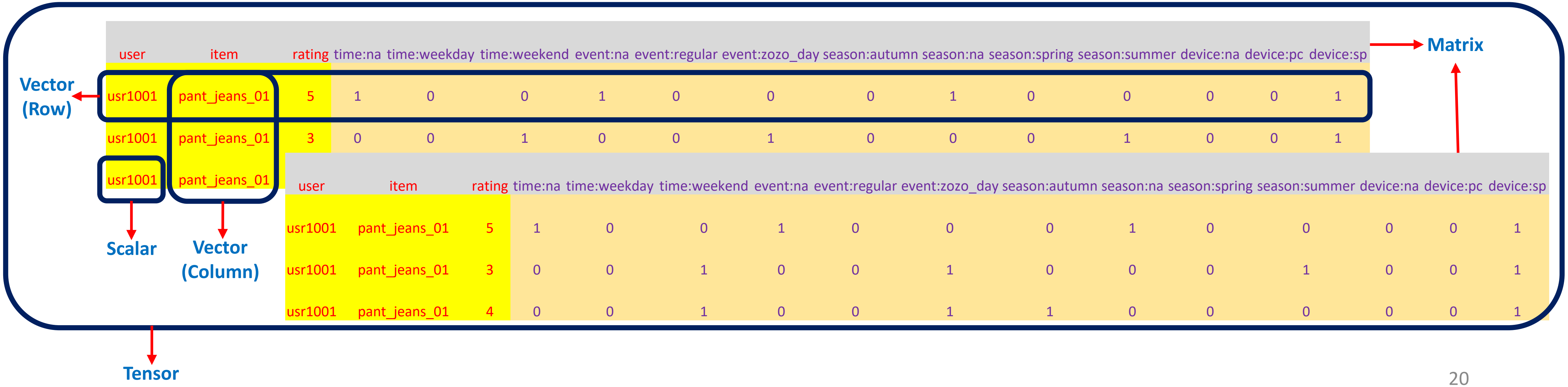


Context Modelling → Data Structure Storage

- ❑ Context Data is preprocessed and converted into Context-wised binary information

userid	itemid	rating	Time	Event	Season	Device
usr1001	Pant_Jeans_01	5	NA	NA	NA	SP
usr1001	Pant_Jeans_01	3	Weekend	ZOZO_Day	Summer	SP
usr1001	Pant_Jeans_01	4	Weekend	ZOZO_Day	Autumn	SP

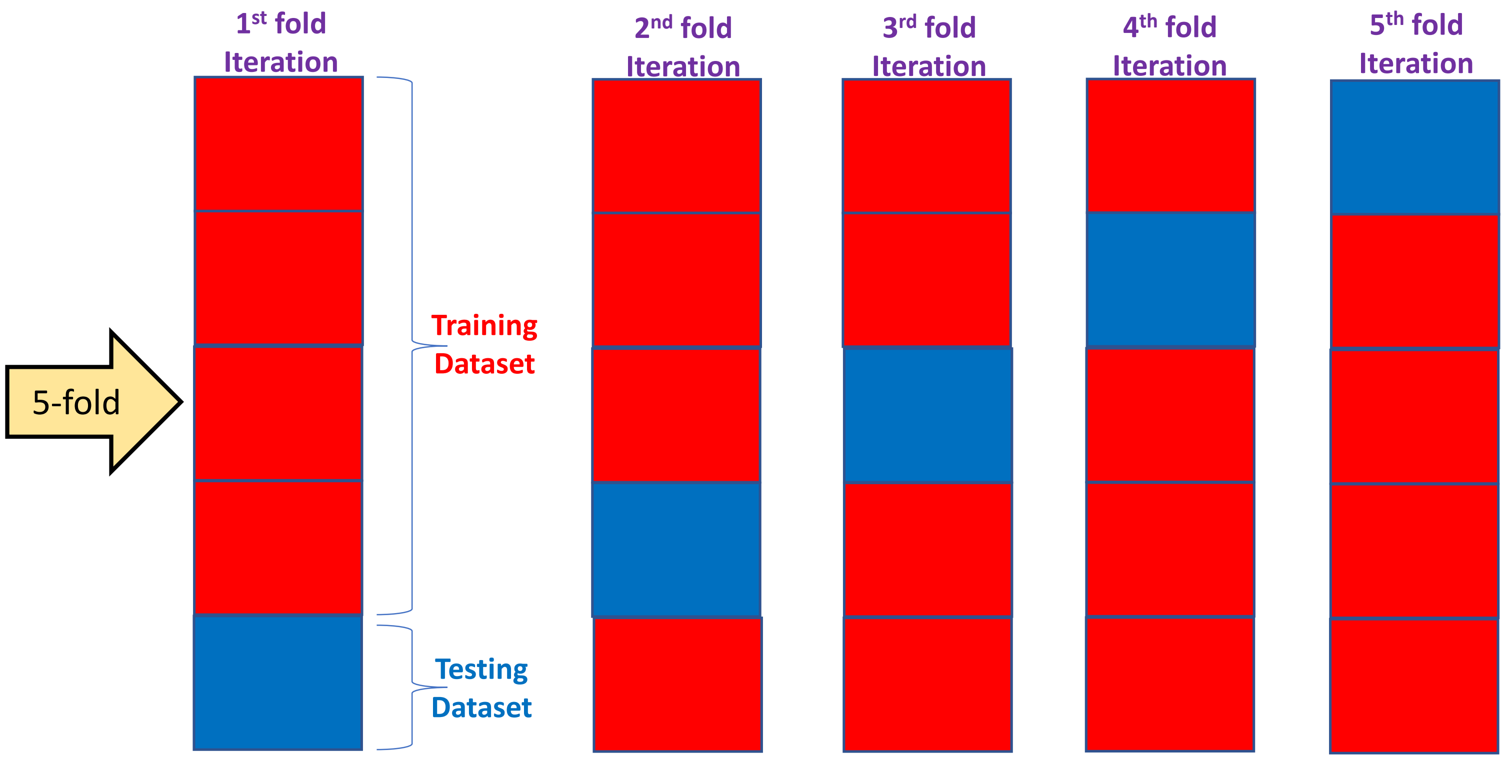
user	item	rating	time:na	time:weekday	time:weekend	event:na	event:regular	event:zozo_day	season:autumn	season:na	season:spring	season:summer	device:na	device:pc	device:sp
usr1001	pant_jeans_01	5	1	0	0	1	0	0	0	1	0	0	0	0	1
usr1001	pant_jeans_01	3	0	0	1	0	0	1	0	0	0	1	0	0	1
usr1001	pant_jeans_01	4	0	0	1	0	0	1	1	0	0	0	0	0	1



ML on Contextualized Data → k-Fold Cross Validation

Context-wise Binary Input Data set

```
usr1001,pant_jeans_01,5,0,0,1,1,0,0,0,1,0,0,1,0,0
usr1001,pant_jeans_01,3,0,0,1,0,0,1,0,0,0,1,0,0,1
usr1001,pant_jeans_01,4,0,0,1,0,0,1,1,0,0,0,0,0,1
usr1001,pant_formal_02,5,0,0,1,1,0,0,0,1,0,0,1,0,0
usr1001,pant_formal_02,2,0,0,1,0,0,1,0,0,0,1,0,0,1
usr1001,pant_formal_02,3,0,0,1,0,0,1,1,0,0,0,0,0,1
usr1001,shirt_denim_02,1,0,0,1,1,0,0,0,1,0,0,1,0,0
usr1001,shirt_denim_02,1,0,0,1,0,1,0,0,0,1,0,0,0,1
usr1001,shirt_denim_02,1,0,0,1,0,0,1,0,0,1,0,0,0,1
usr1001,t_shirt_formal_02,4,0,0,1,1,0,0,0,1,0,0,1,0,0
usr1001,t_shirt_formal_02,3,0,0,1,0,0,1,1,0,0,0,0,0,1
usr1001,skirt_formal_01,4,0,0,1,1,0,0,0,1,0,0,1,0,0
```



- Cross-Validation → a technique to evaluate ML model by,
  - training it on subsets of available input data
  - evaluating it on complementary subset of data.
- k-Fold Cross Validation → [1] reduces bias as most of data is used for fitting → [2] reduces variance as most of the data is used in validation set.
- K = 5 or K = 10 → works best as per certain empirical evidences.

Context Modelling Result → k-Fold Intermediate Recommendations

❑ K-Fold Intermediate Recommendation Result

User ID	Context Condition	Recommendation 1	Recommendation 2	Recommendation 3	Recommendation 4
usr1001	device:sp;event:na;season:na;time:na:	(onepeace_13 5.265406)	(womenbelt_03 5.239005)	(t_shirt_formal_13 4.4583693)	(menbelt_04 4.3078666)
usr1001	device:sp;event:zozo_day;season:summer;time:weekend:	(onepeace_13 3.9239478)	(womenbelt_03 3.8975468)	(womenbelt_06 3.6858723)	(pant_jeans_01 3.5560606)
usr1002	device:sp;event:na;season:na;time:na:	(half_shirt_formal_02 4.264601)	(menshoes_03 4.2473464)	(t_shirt_formal_04 4.141822)	(onepeace_01 4.0362325)
usr1002	device:sp;event:regular;season:summer;time:weekday:	(pant_formal_01 4.9841743)	(half_shirt_formal_02 4.734595)	(menshoes_03 4.71734)	(t_shirt_formal_04 4.6118155)
usr1003	device:sp;event:na;season:na;time:na:	(pant_jeans_12 3.5445545)	(socks_04 3.5021193)	(menbelt_06 3.2302444)	(menshoes_02 3.2236366)
usr1003	device:sp;event:zozo_day;season:spring;time:weekday:	(pant_jeans_12 2.2311978)	(socks_04 2.1887627)	(menbelt_06 1.9168875)	(menshoes_02 1.9102799)
usr1004	device:sp;event:na;season:na;time:na:	(socks_04 6.1128426)	(menbelt_04 6.0229573)	(shirt_formal_03 5.7797804)	(womenbelt_03 5.5392733)
usr1004	device:sp;event:zozo_day;season:autumn;time:weekday:	(socks_04 4.757492)	(menbelt_04 4.667607)	(shirt_formal_03 4.42443)	(pant_jeans_01 4.23796)
usr1005	device:sp;event:na;season:na;time:na:	(t_shirt_formal_02 5.1630282)	(top_formal_13 5.072566)	(half_shirt_formal_06 4.9986386)	(pant_formal_01 4.6567273)
usr1005	device:sp;event:zozo_day;season:summer;time:weekend:	(t_shirt_formal_02 4.7874174)	(top_formal_13 4.696955)	(half_shirt_formal_06 4.6230273)	(pant_formal_01 4.2811165)

The reverse recommendation can also be done based on the Context Condition e.g.,

device : sp; event : zozo_day; season : summer; time : weekend

This is very useful for non-member users whose purchase history is not available.

References

1. F. Ricci, L. Rokach, B. Shapira, "*Recommender Systems Handbook*", Springer, Second Edition, 2015.
2. X. Amatriain, "*Mining Large Streams of User Data for Personalized Recommendations*", SIGKDD Explorations, Volume 14, Issue 2, pages 37 – 48, December 2012.
3. B. Lamche, Y. Rodl, C. Hauptmann, W. Worndl, "*Context-Aware Recommendations for Mobile Shopping*", ACM RecSys'15, Workshop on Location-Aware Recommendations (LocalRec'15), Vienna, Austria, Sept. 19, 2015.
4. K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, et. all, "*Context-Aware Recommender Systems for Learning: A Survey and Future Challenges*", IEEE Transactions on Learning Technologies, Vol. 5, No. 4, pages 318-335, Oct-Dec 2012.
5. Y. Zheng, "*Tutorial: Context in Recommender Systems*", The 31st ACM Symposium on Applied Computing, Pisa, Italy, April 4th 2016.
6. P. Lops, "*Semantics-aware Content-based Recommender Systems: the Big Picture*", Workshop on New Trends in Content-based Recommender Systems, CBRecSys 2014 – Foster City, CA, US, October 6th 2014.
7. K. Nishida, K. Sadamitsu, R. Higashinaka, Y. Matsuo, "*Understanding the Semantic Structures of Tables with a Hybrid Deep Neural Network Architecture*", 31st AAAI Conference on Artificial Intelligence AAAI-17, San Francisco, US, pages 168-174, February 2017.
8. S. Rendle, Z. Ganter, et all, "*Fast Context-aware Recommendations with Factorization Machines*", ACM SIGIR'11, Beijing, China, July 24-28, 2011.

Thanks for your kind attention!!!


start today technologies

Project Manager
Starttoday Research

Sidhant Rajam

sidhant.rajam@starttoday.jp

Start Today Technologies Inc.

Head office : Aoyama Oval Building 3F, 5-52-2
Jingumae, Shibuya-ku, Tokyo 150-0001, Japan
Tel : +81-3-3407-8811 / Fax : +81-3-5774-8844

Makuhari office : WBG West 16F, 2-6-1 Nakase,
Mihama-ku, Chiba-shi, Chiba 261-7116, Japan
Tel : +81-43-213-7474 / Fax : +81-43-213-5175

starttoday-tech.com zozo.jp wear.jp

Any Questions / Suggestions / Feedback ??

sidhant.rajam@starttoday.jp

(LinkedIN → https://www.linkedin.com/in/sidhantrajam/)

start today technologies

We are Recruiting!!

Researchers, Data Scientists, etc.

https://www.starttoday-tech.com/en/recruit/statistics/