

一から学ぶ

開発者のための

レコメンド機能

その2

作成者

Author: のりふみ

ふらっと考える人

URL: <http://omohikane.com>

Twitter: @norio_yht

前回の内容をちょっと・・・

開発者目線のレコメンドとは

「お勧め提示」

≡システム的には
「インプットに対する
アウトプットのこと」

=>入力に対してランダムで
結果をアウトプットしても
レコメンドになる！

(極論ではあるが)



①サイコロ

どんなインプットに対してもランダムで返す

(Ex)買った商品に対して、ランダムで選んだ商品をおすすめする。



インプット、アウトプットだけに着目すると
「サイコロ」も立派なレコメンドかも…

②コンテンツベースレコメンド

インプットの属性で引っ張った結果を返す



属性に着目する

(例) コーヒー豆のレコメンド



属性

	テイストバランス	香り	ロースト	産地	相性の良い淹れ方	ラベル
コーヒー豆①	苦味強め	豊か	深煎り	ブラジル	フレンチプレス	フェアトレード
コーヒー豆②	酸味強め	豊か	深煎り	ベトナム	フレンチプレス	フェアトレード
コーヒー豆③	苦味強め	控えめ	浅煎り	ブラジル	フレンチプレス	無し
コーヒー豆④	苦味強め	豊か	深煎り	コロンビア	ペーパードリップ	バードフレンドリー
⋮						
コーヒー豆N	バランス	控えめ	深煎り	ブラジル	ペーパードリップ	無し

属性に着目しておすすみを引っ張ってくる

多彩な応用力

- ・ 各属性に対する重み付け
- ・ リアルタイム処理

その他諸々…



前処理と、後処理を加えることで
他にはないレコメンドサービスが作れる！！

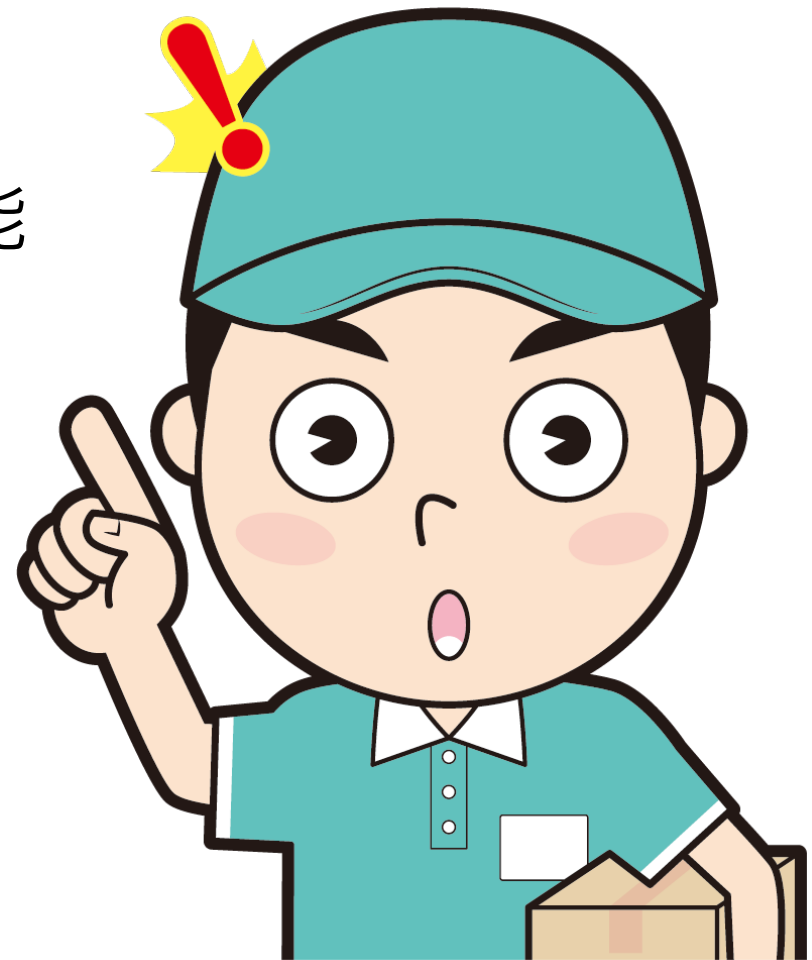
コンテンツベースレコメンズの メリットとデメリット

メリット

- ・細かいチューニングが可能

デメリット

- ・ターゲットが変わると再チューニングが必要
- ・意外性の欠如

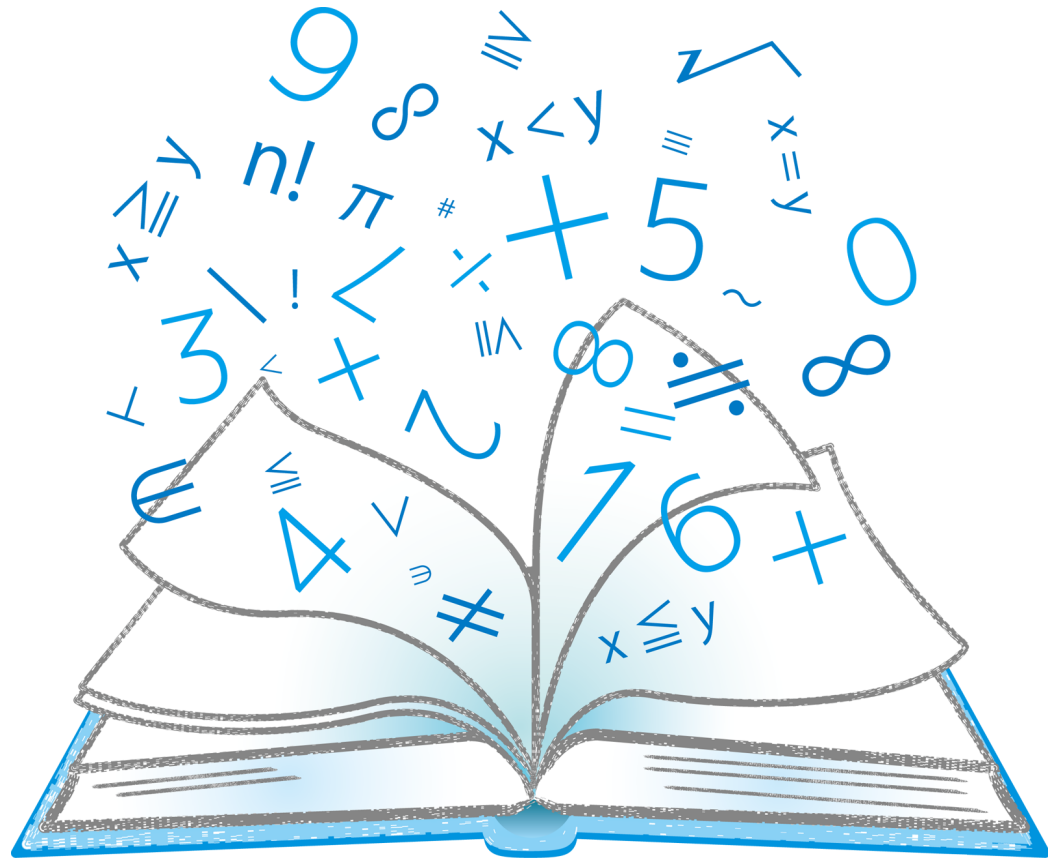


今回のテーマ

協調フィルタリング

協調フィルタリングとは

数学的なアルゴリズムを用いて類似アイテムを見つけ出す。

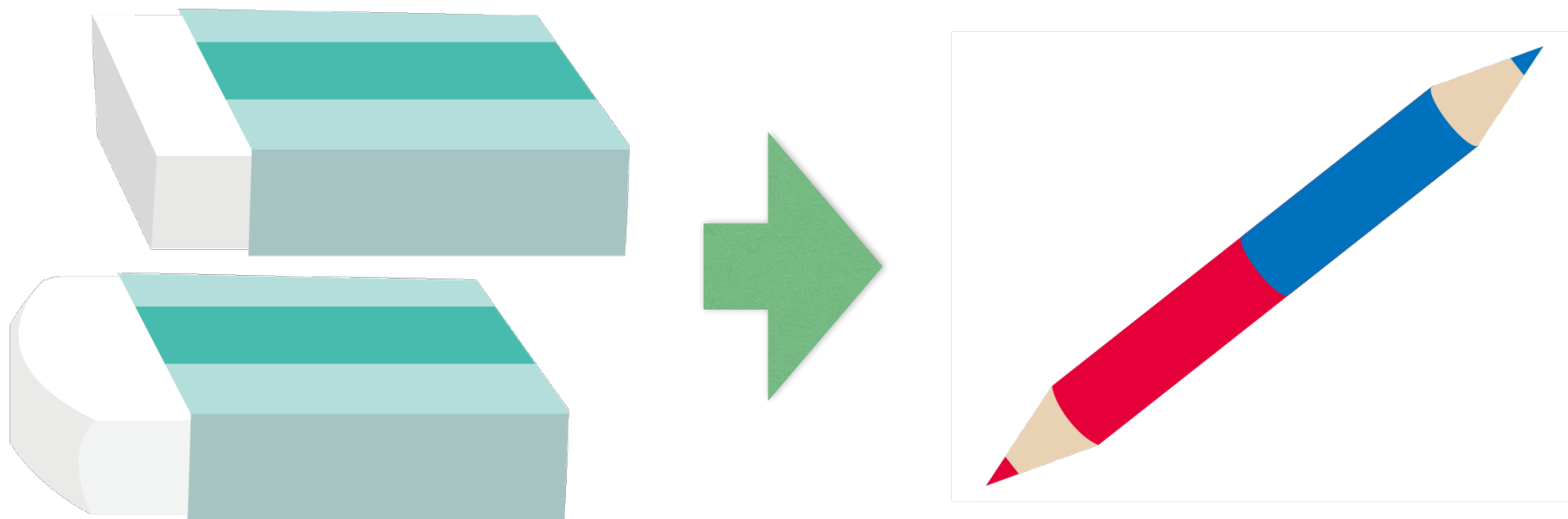


代表的なレコメンド

- ① アイテムベースレコメンド
- ② ユーザーベースレコメンド

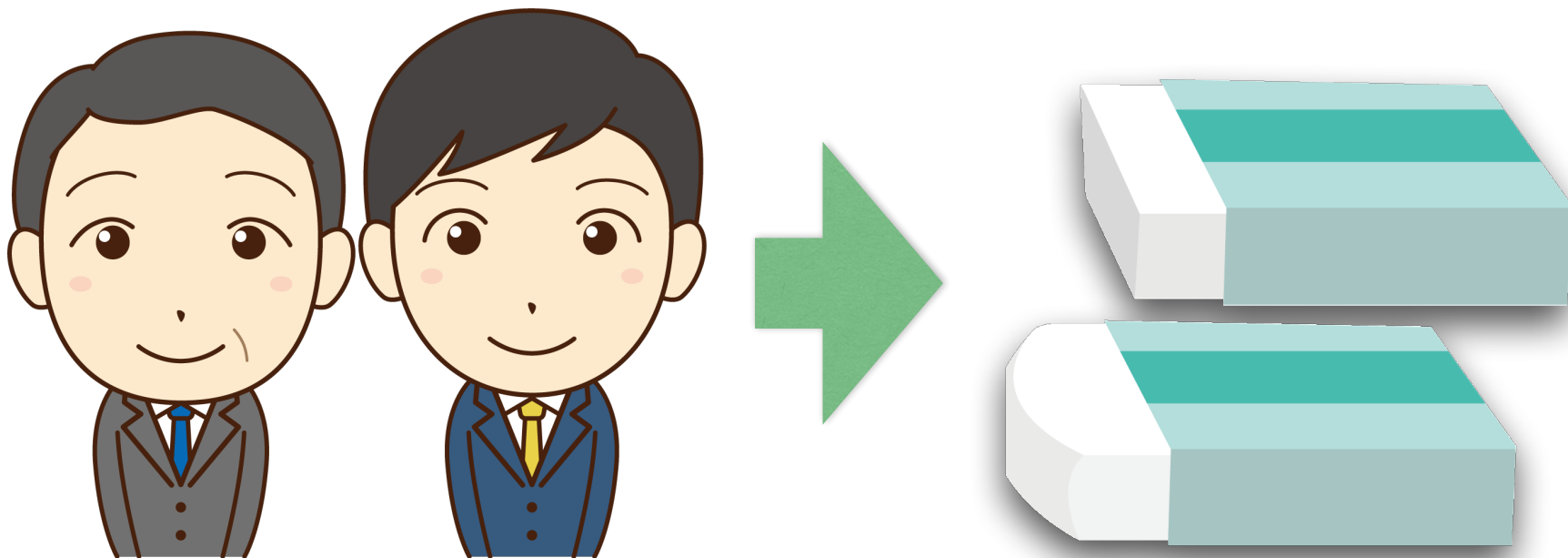
①アイテムベースレコメンド

「この商品と同時に買われた商品です」



②ユーザーベースレコメンド

「この**商品**を買ったユーザーは
こちらの**商品**も買っています」



コンテンツベースと 協調フィルタリングの違い

属性を使わなくていい

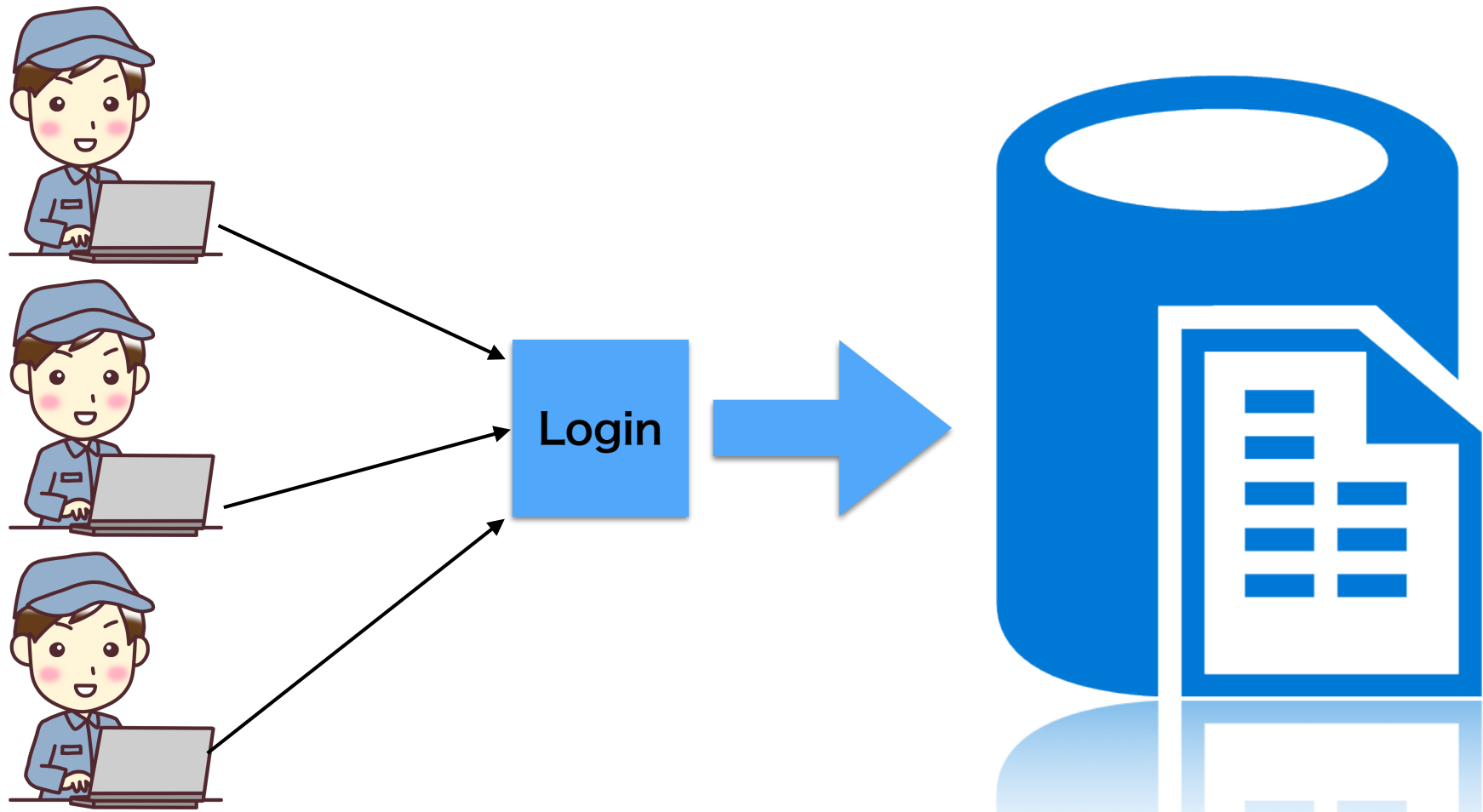
他のコンテンツでの応用が効く

メモリは食っちゃうけど
実装者がそのコンテンツ
に精通している必要は無い
から楽や〜ん



類似度を計算する前に

類似商品(ユーザー)を計算するためには、
多くの場合、ログイン有のシステムによって**ユーザーの特定が容易**



類似計算は様々な方法がある

- ・ 共起カウント
- ・ コサイン角度
- ・ ユークリッド距離

etc...

今回は**共起カウント**を
使って類似計算するよ



共起

あるアクションに対して、
同時に起きうる可能性の高い
アクションのこと

Ex)ある商品を買った時、
同時に購入されている商品を

共起商品という。

(Ex)ECサイトの場合

収集したログには、ユーザーとそのユーザーが買った商品が記録されている(Aさんがbとcとdの商品を買った等)

商品 ユーザー	a	b	c	d	e
A	○		○		○
B		○			○
C		○			
D	○	○			○
E			○	○	

商品とユーザーの2軸

アイテムベースレコメンドの場合

商品 ユーザ	a	b	c	d	e
A	○		○		○
B		○			○
C		○			
D	○	○			○
E			○	○	

商品の類似計算のため
共起マトリックスを作る

Ex) aとbが同じユーザに買われた回数は?
→1回

商品 商品	a	b	c	d	e
a		1	1	0	2
b	1		0	0	2
c	1	0		1	1
d	0	0	1		0
e	2	2	1	0	

商品aと商品eは同じユーザに
買われる回数が多いので
商品a、商品c等と比べ
関連度が高い(類似している!!)

アイテムベースレコメンドの場合

商品 ユーザ	a	b	c	d	e
A	○		○		
B		○			
C		○			
D	○	○			
E			○	○	

aはeは似ているっぽいので
商品aを買った人には
商品bをおすすめしよう！！

商品aと商品eは同じユーザに
買われる回数が多いので
商品a、商品c等と比べ
関連度が高い(類似している!!)

c				1	1
d	0	0	1		0
e	2	2	1	0	

ユーザーベースレコメンドの場合

商品 ユーザ	a	b	c	d	e
A	○		○		○
B		○			○
C		○			
D	○	○			○
E			○	○	

ユーザーの類似計算のため
共起マトリックスを作る

Ex) AとDが同じ商品を買った回数は?
→2回

ユーザ ユーザ	A	B	C	D	E
A		1	0	2	1
B	1		1	2	0
C	0	1		1	0
D	2	2	1		0
E	1	0	0	0	

ユーザAとユーザDは同じ
商品を回数が多いので
ユーザA、ユーザC等と比べ
同一嗜好の可能性が
高い(類似している!!)

ユーザーベースレコメンドの場合

商品 ユーザ	a	b	c	d	e
A	○		○		○
B		○			
C		○			
D	○	○			
E			○	○	

ユーザAとユーザDは同じ
商品を回数が多いので
ユーザA、ユーザC等と比べ
関連度が高い(類似している!!)

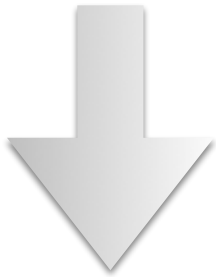
A、Dは類似したユーザー
なので、Aの買った
商品cをDにおすすめしよう！

*ちなみにDが商品cをまだ
買っていないのもポイント

D	2	2	1	0	0
E	1	0	0	0	0

2軸の選択は自由!!

2軸の入力マトリックスまで
落とし込めればレコメンドは可能



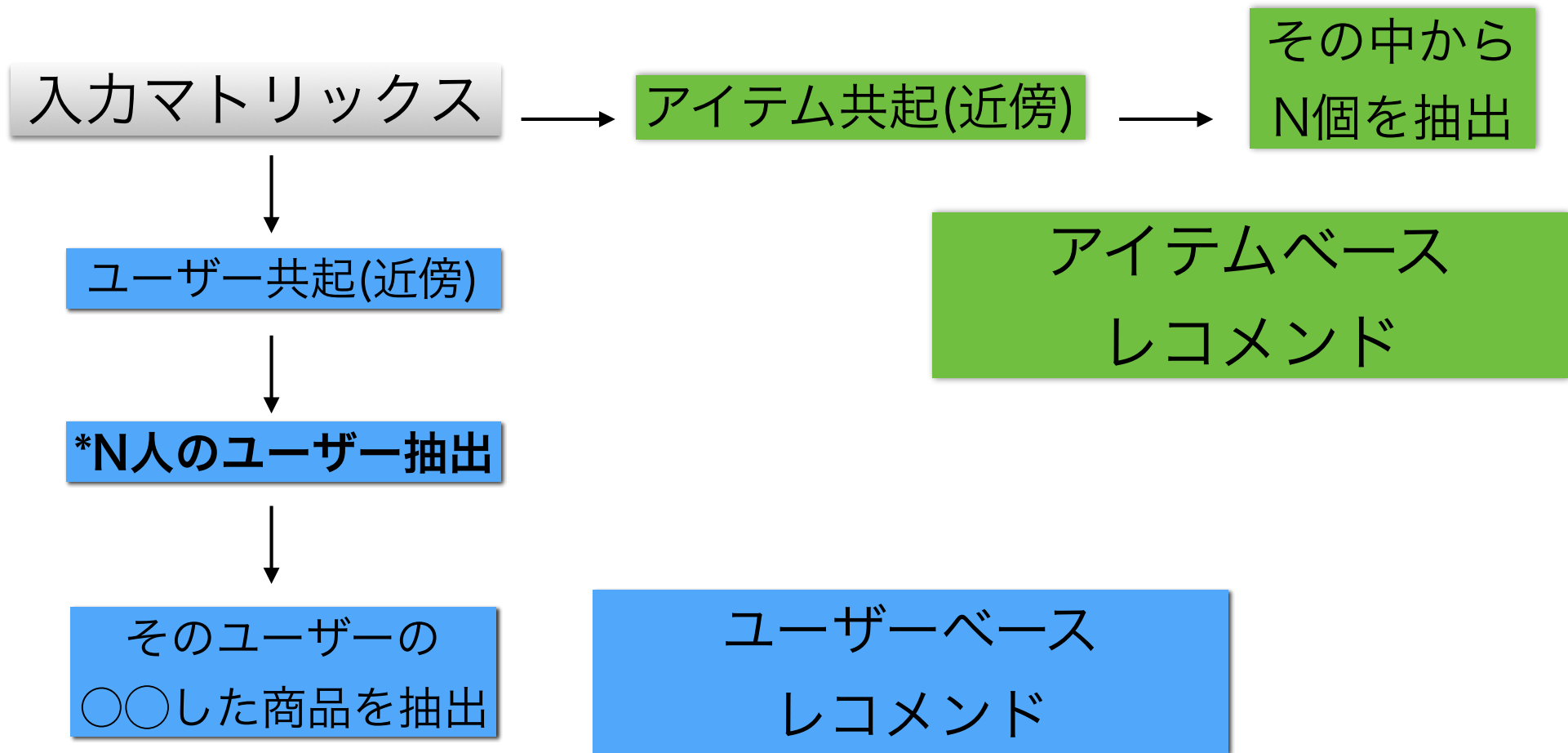
ユーザーや商品に
拘る必要はない



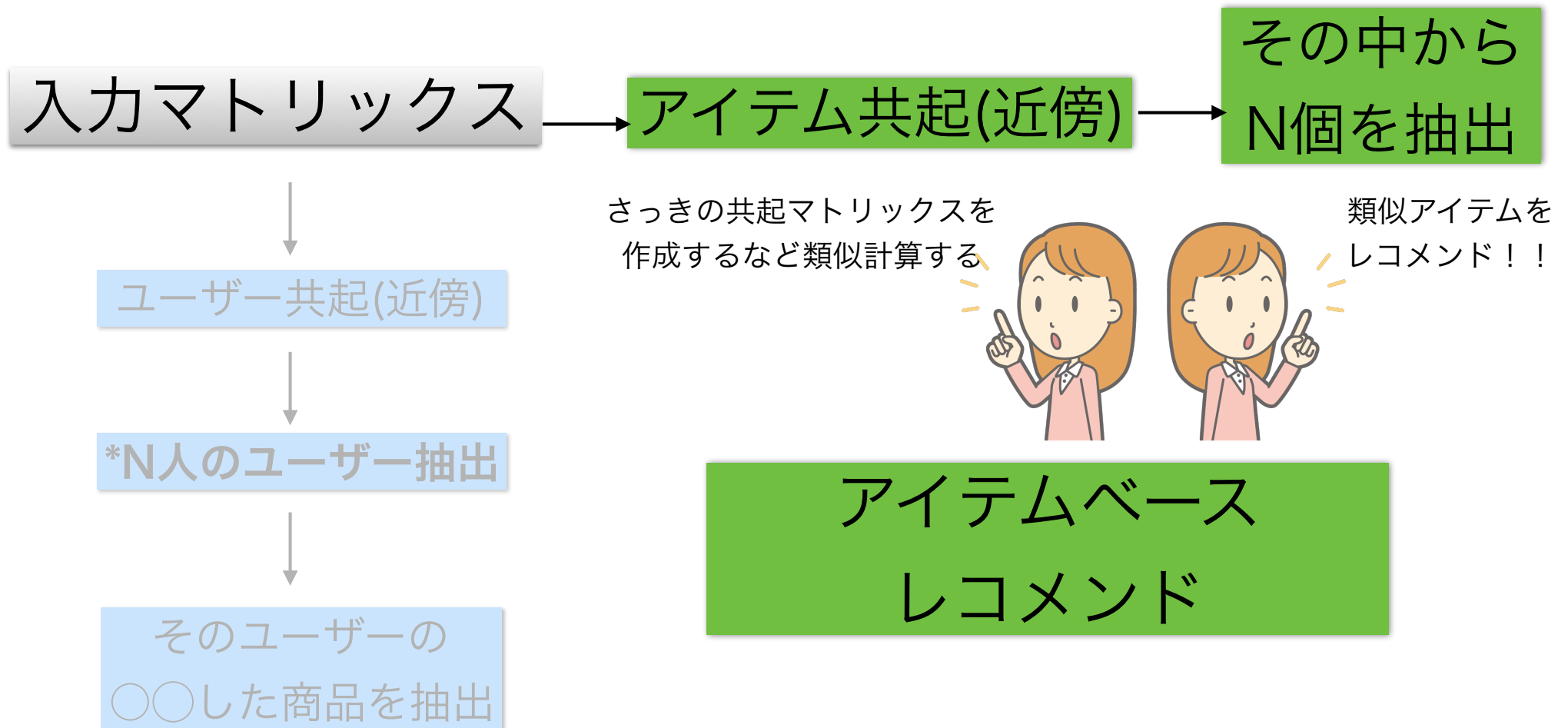
「似ている」の定義付けや
近傍の作り方しだいで
結果が変わる!!



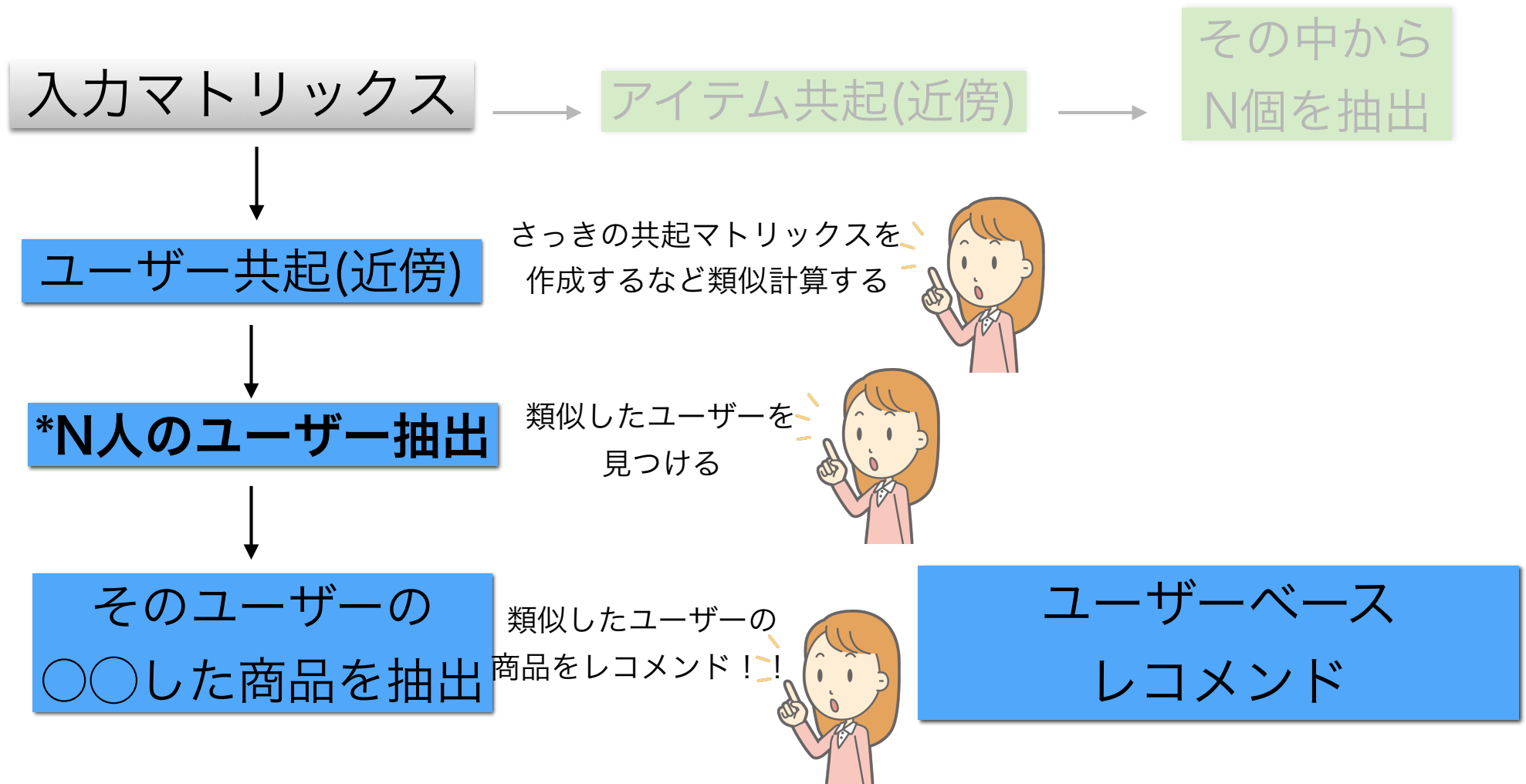
協調フィルタリングのステップ



アイテムベースレコメンドのステップ



ユーザーベースレコメンズのステップ



協調フィルタリングのステップ



協調フィルタリングの メリットとデメリット

メリット

- ・他のドメインでの応用が可能
- ・様々な軸に基づいたレコメン
ドが可能

デメリット

- ・新しくはじめるサービスの
場合、データ収集が困難

≡コールドスタート問題



改めて...

協調フィルタリングの重要なポイント

類似計算は様々な方法がある

- ・ 共起
- ・ コサイン角度
- ・ ユークリッド距離

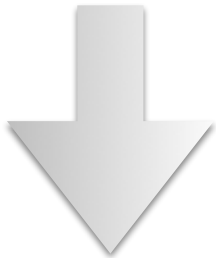
etc...

数学的なアプローチで
様々な計算方法があるぞ！！



2軸の選択は自由!!

2軸の入力マトリックスまで
落とし込めればレコメンドは可能



ユーザーや商品に
拘る必要はない

「似ている」の定義付けや
近傍の作り方しだいで
2軸は造りたい放題!!



「商品と購入時間帯」も
2軸の入力マトリックスに
落とし込めそうだ!!..とかね



そう考えると...

協調フィルタリングは、
類似の定義や計算方法な
ど**設計者のアイディア**が
重要になってくる！？



作成者より

このサイトのコンテンツはこれからシステムの基礎から学びたい！という人向けに作成しております。内容に対するご指摘、ご感想はどんどん受け付けております。よりよい知識の伝播のためにもぜひご協力お願い致します。

ふらっと考える

検索

