Better Capturing Interactions between Products in Retail: Revisited Negative Sampling for Basket Choice Modeling

Jules Désir, Vincent Auriau, Martin Mozina, Emmanuel Malherbe

CentraleSupélec

ARTEFACT RESEARCH CENTER

FORTENOVA GROUP

ECML PKDD 2025

– Internal –

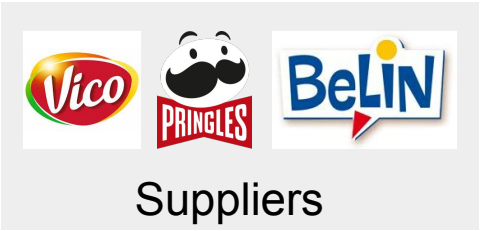
Agenda.

1. Introduction - Retail, assortment optimization & products interactions
2. Models Presentation
3. Results on Synthetic Experiments
4. Results on Real World Experiments

ARTEFACT

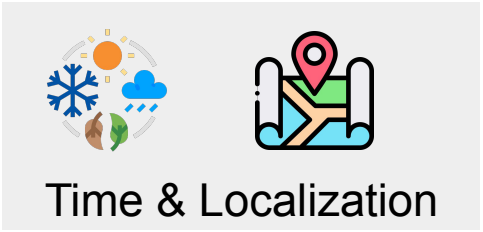
– Internal –

Supermarket Assortment Optimization



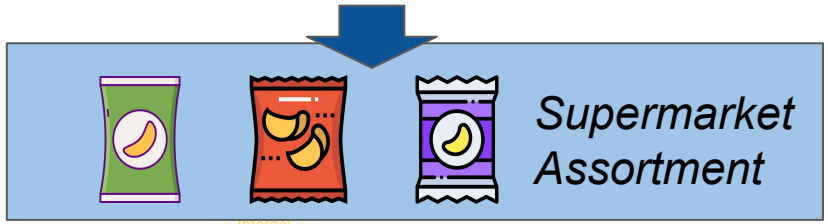












– Internal –

Focus of our work: complementarity & cannibalization detection



Light Cola



Cola



Tonic

Available Products



Naive Assortments Ranking

– Internal –

Focus of our work: complementarity & cannibalization detection



Zero Cola



Cola

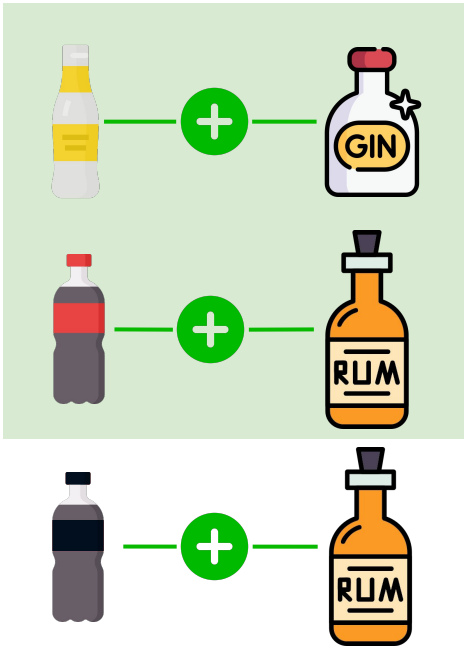


Tonic

Available Products



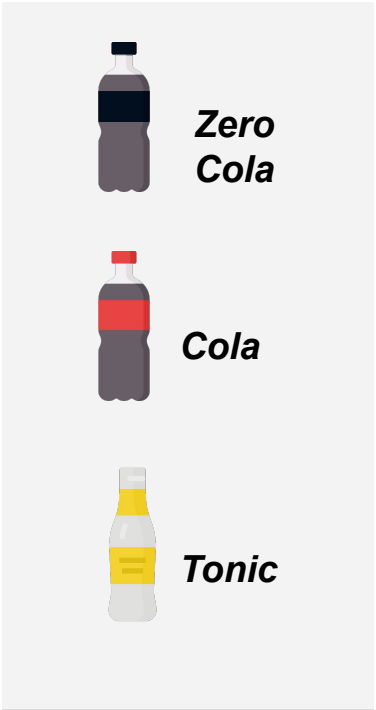
Naive Assortments Ranking



Complementarities

– Internal –

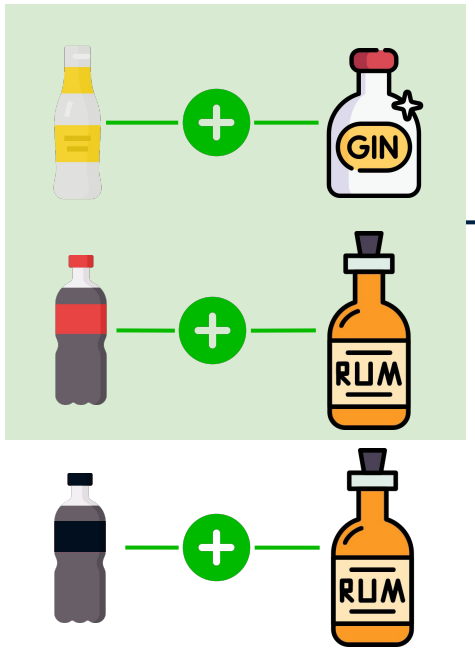
Focus of our work: complementarity & cannibalization detection



Available Products



Naive Assortments Ranking



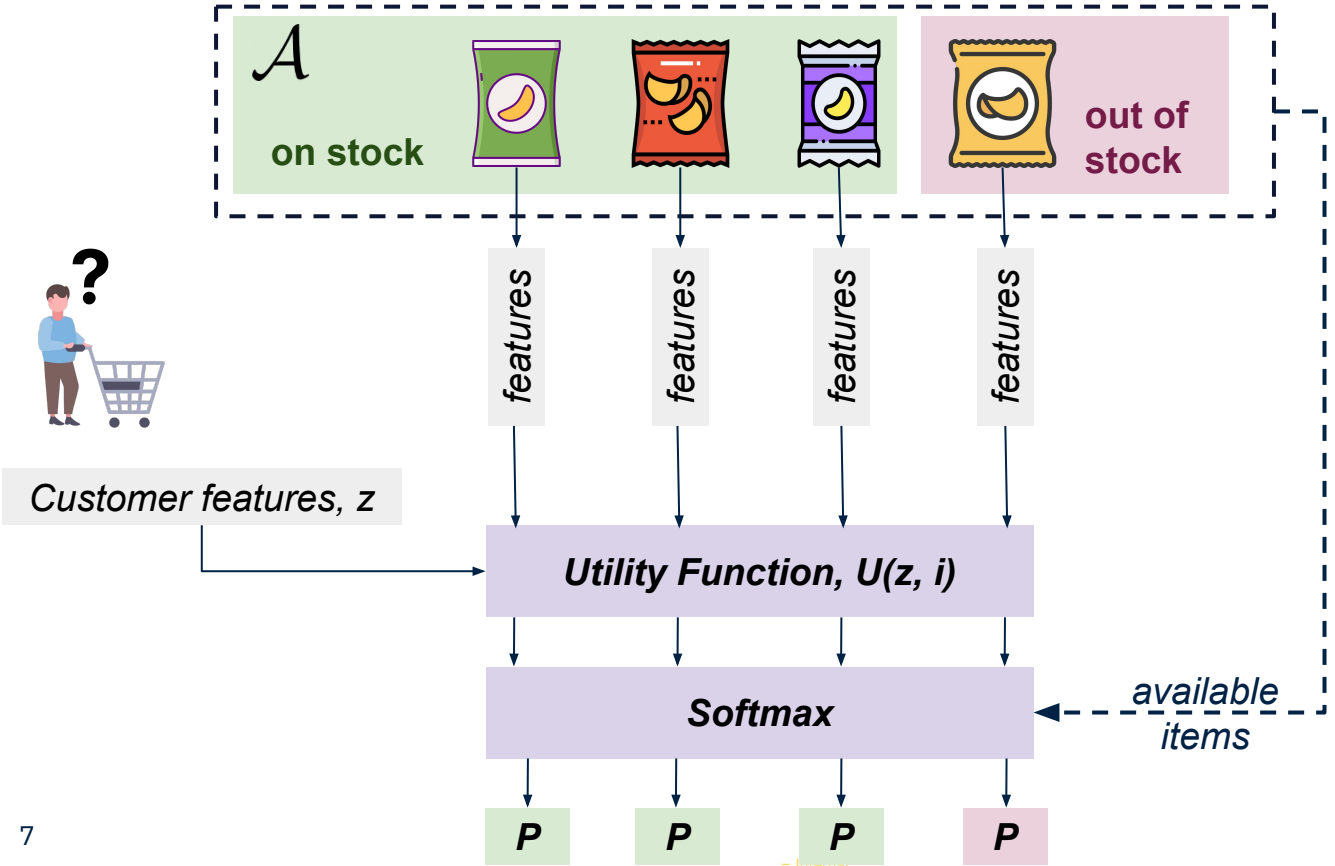
Complementarities

Final "Best" Assortment

– Internal –

ARTEFACT

A usual framework for assortment optimization: choice modeling

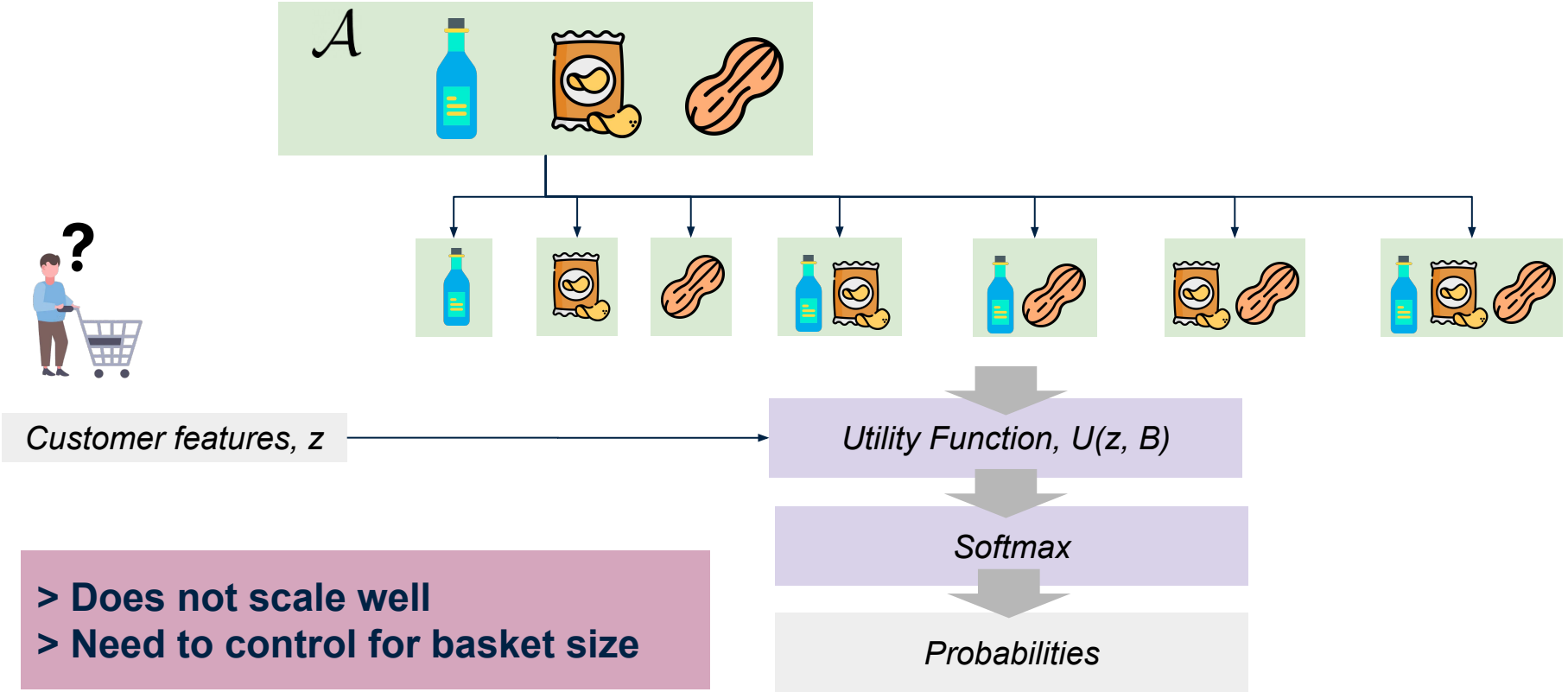


$$U(z,i) = \alpha_i + \beta_z \cdot p(i)$$

$$\mathbb{P}(i) = \frac{e^{U(z,i)}}{\sum_j 1_{j \in \mathcal{A}} \cdot e^{U(z,j)}}$$

ARTEFACT

The Basket-MVL model

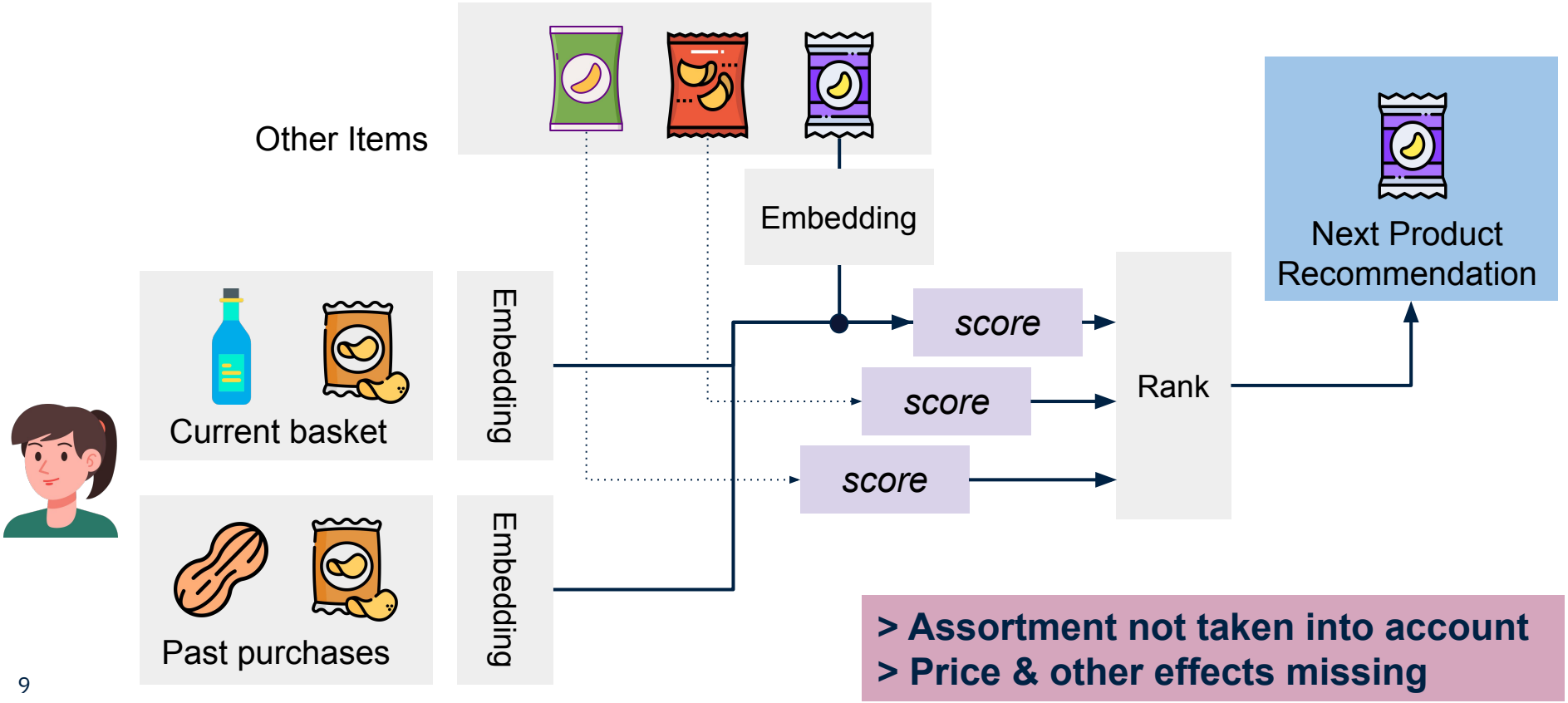


> Does not scale well
> Need to control for basket size

– Internal –

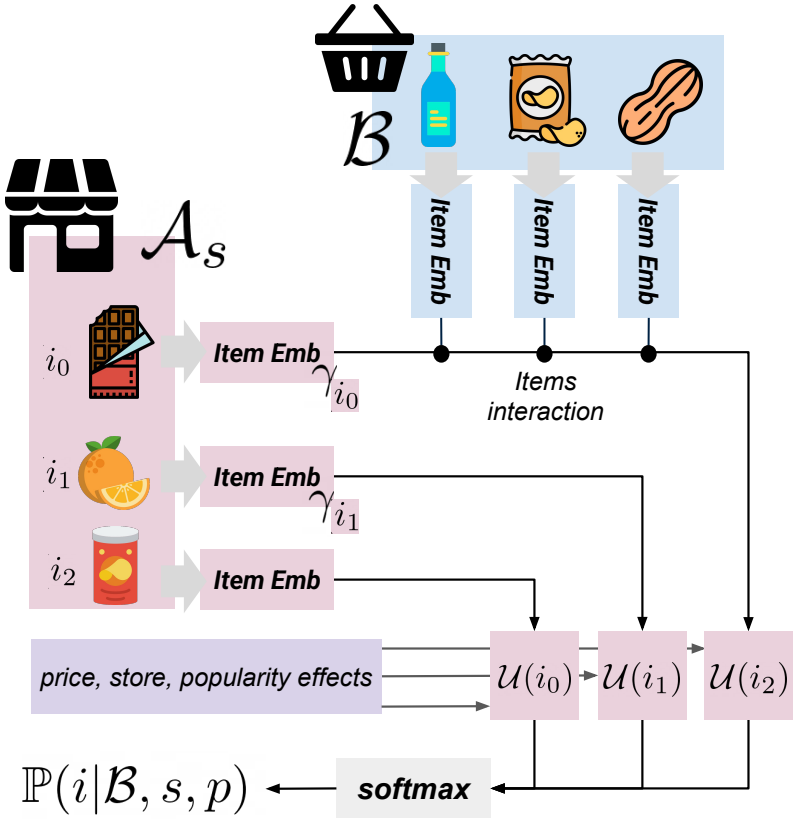
ARTEFACT

Representation Learning & Recommender Systems



– Internal –

Our proposed approach



$$U(i|\mathcal{B}, s, p) =$$

α_i *itemwise intercept*

$+ < \beta_i, \delta_s > \cdot \log p(i)$ *store x price effect*

$$+ \frac{1}{|\mathcal{B}|} \sum_{j \in \mathcal{B}} < \gamma_i, \gamma_j >$$ *item x basket interaction*

> Prediction of the next purchased item
> Conditional utility formalutation

– Internal –

Our proposed approach: training

Conditional Probability:

$$\mathbb{P}(i|\mathcal{B}, s, p) = \frac{e^{\mathcal{U}(i|\mathcal{B},s,p)}}{\sum_{j \in \mathcal{A} \setminus \mathcal{B}} e^{\mathcal{U}(j|\mathcal{B},s,p)}}$$

- Complexity reduced from the 2^N baskets to N items
- Still computationally expensive

Training Loss :

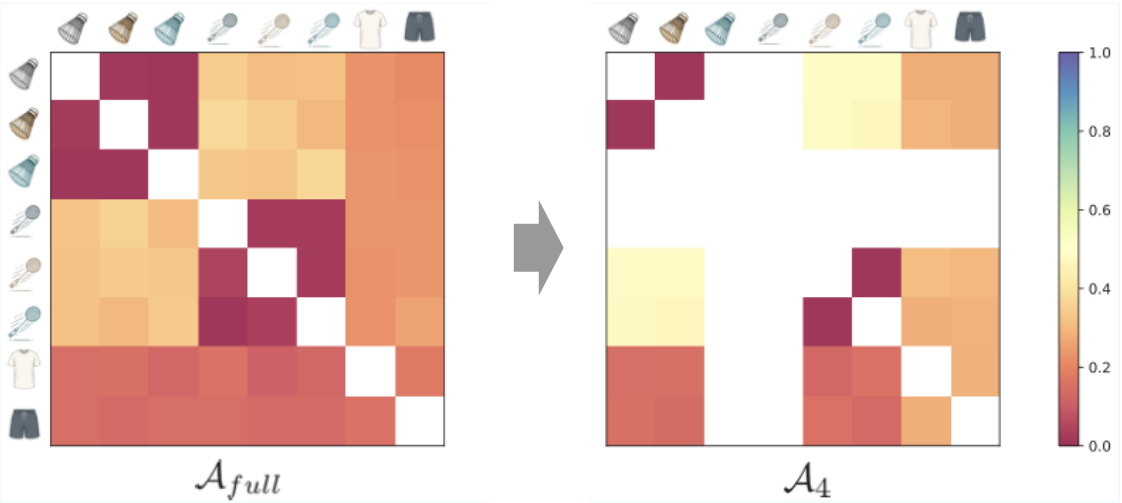
$$\mathcal{L} = -\sum_{k=1}^{N} \sum_{i \in \mathcal{B}_k} \left(\log \sigma \left[\mathcal{U}(i|\mathcal{B}_k \backslash i, s_k, p_k) \right] + \sum_{j \in \mathcal{NS}} \log \left(\sigma \left[\mathcal{U}(j|\mathcal{B}_k \backslash i, s_k, p_k) \right] \right) \right)$$

positive sample: "leave one item out"

$\mathcal{NS} \subset \mathcal{A}_k$ *negative samples, selected within the assortment*

– Internal –

Synthetic Experiments



Ground Truth Interactions

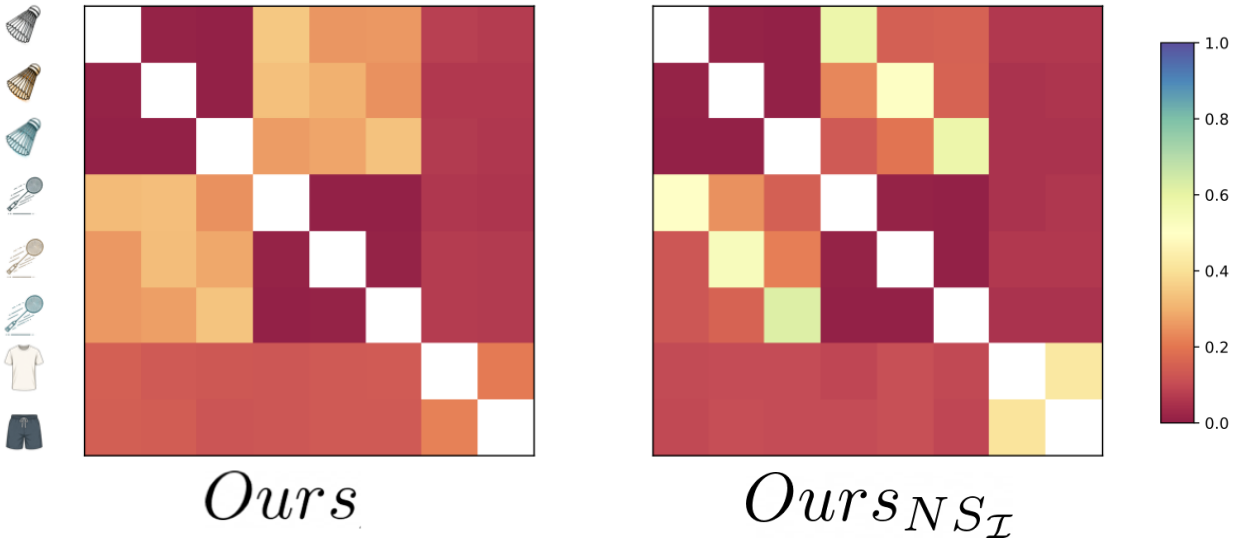
Generalization to sub-assortment

Basket Sampling

– Internal –

ARTEFACT

Results & comparison with other models



$Ours$ $Ours_{NS_{\mathcal{I}}}$

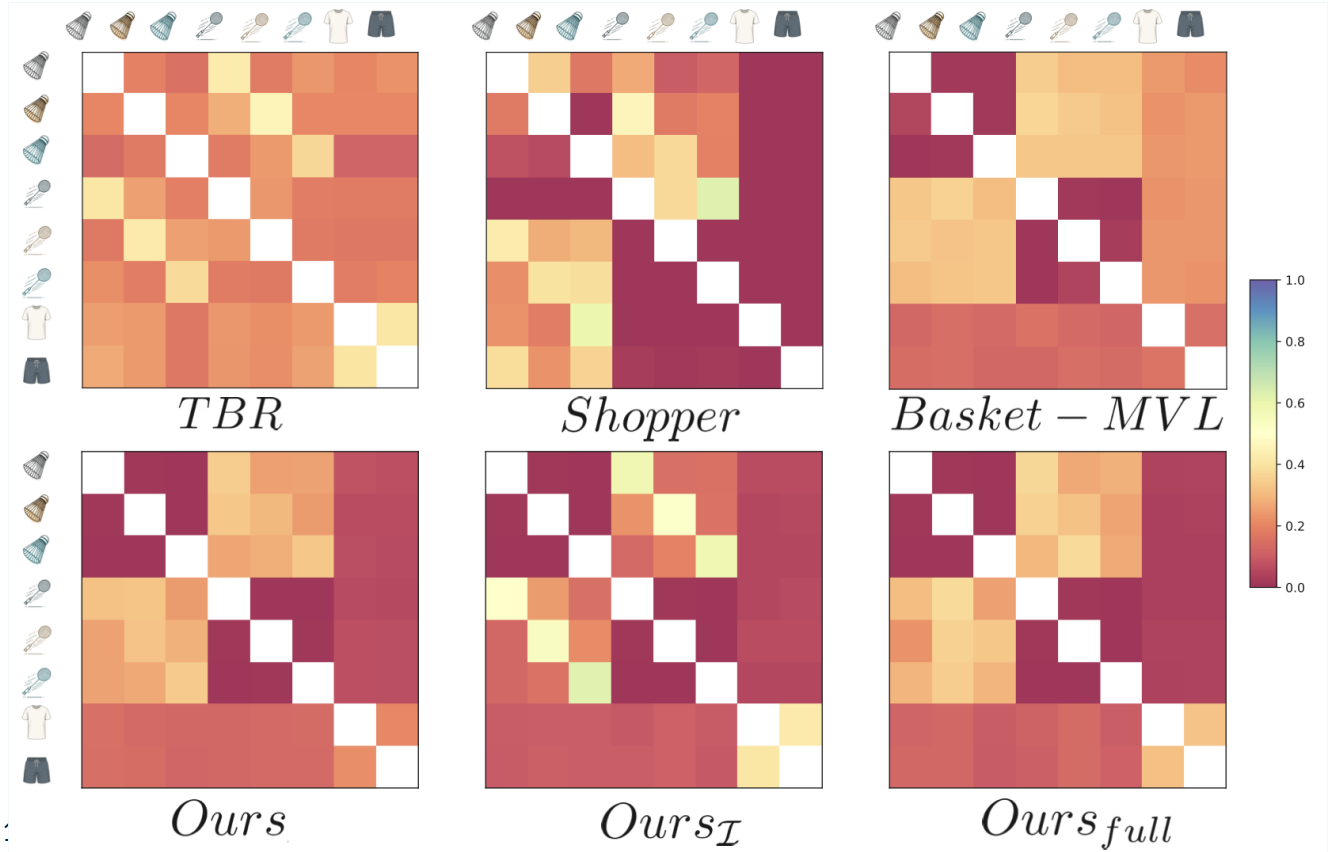
> Observation of $\mathbb{P}(i|j)$ for the different models

> Most models pick interactions signal may be degraded

> Taking into account assortment during training is definitely key even though it brings computational complexity

– Internal –

Results & comparison with other models



> Observation of $\mathbb{P}(i|j)$ for the different models

> Most models pick interactions signal may be degraded

> Taking into account assortment during training is definitely key even though it brings computational complexity

– Internal –

Results & comparison with other models

Runtime & memory consumption

	100 Items		1000 Items	
	Process Time	Fits in RAM	Process Time	Fits in RAM
Shopper	100s	✓	-	×
Python-Shopper	110s	✓	620s	✓
Basket-MVL	3,000s	✓	-	×
TBR	550s	✓	1020s	✓
Ours	80 s	✓	360s	✓

Performance

Model	Shopper[4]	Basket-MVL[2]	TBR[3]	Ours$_{full}$	Ours$_{\mathcal{I}}$	Ours
$NLL(\mathcal{D}_{full})$	1.75	1.69	1.82	**1.25**	1.43	1.26
$NLL(\mathcal{D}_4)$	1.49	1.41	1.55	0.97	1.07	**0.95**
$MRR(\mathcal{D}_{full})$	0.32	0.41	0.28	**0.42**	0.39	0.41
$MRR(\mathcal{D}_4)$	0.45	0.60	0.45	**0.62**	0.59	0.61

– Internal –

Results on a real world dataset: Fortenova



FORTENOVA GROUP

#Referenced Items	#Shops	#Baskets	#Assortments	Avg Basket Size
6,000+	600+	100,000,000+	10,000+	12.4

> 4 years of data

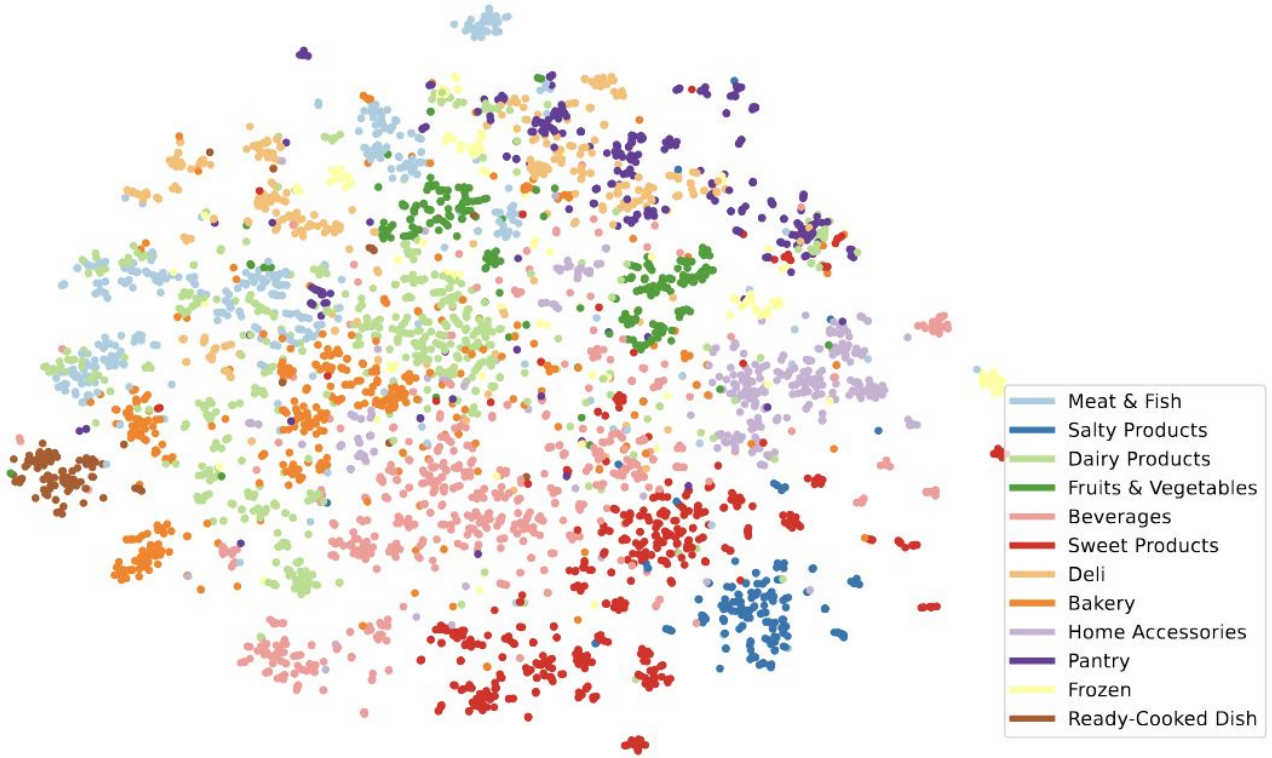
> Daily changes of assortment (by shop)

> Scalability definitely important

– Internal –

ARTEFACT

Results on a real world dataset: Fortenova



> 2D projection of interactions Embeddings

> T-SNE Algorithm

> Uncovers "clusters" of item of same category

> Definitely learns meaningful information

– Internal –

ARTEFACT

Results on a real world dataset: Fortenova

Soda A **0.5L**		Soda A **2L**	
Water A **0.5L**	Water B **1L**	Soda B **2L**	Soda C **2L**
Soda B **0.5L**	Water C **0.5L**	Water A **1.5L**	Sweets A **500g**
Soda C **0.5L**	Energy Drink **0.5L**	Soda D **1.5L**	Water B **1L**
Soda D **0.5L**	Strong Alcohol **0.1L**	Light Alcohol **0.5L**	Soda E **2L**
Water D **0.5L**	Ice Cream **0.16L**	Sweets B **100g**	Soda F **2L**
Tomato sauce A		Vanilla powder A	
Pasta A	Pasta B	Margarin A	Chocolate A
Bread A	Pizza Dough A	Chocolate B	Chocolate C
Pasta C	Other Sauce A	Baking Powder A	Sugar A
Toasts A	Other Sauce B	Baking Powder B	Margarin B
Bread B	Other Sauce C	Whipped Cream A	Margarin C

Detection of most-complementary products

– Internal –

Thank you for your Attention !



Test the model and star us on GitHub