Diversified recommendations of cultural activities with personalized determinantal point processes (DPP)

Carole Ibrahim[1], Hiba Bederina[2], Daniel Cuesta[1], Laurent Montier[1], Cyrille Delabre[1], Jill-Jênn Vie[2]

[1] *pass Culture, 35 rue de Sébastopol, 75001 Paris, France*
[2] *Soda team, Inria Saclay, 1 rue Honoré d'Estienne d'Orves, 91120 Palaiseau, France*

1. The pass Culture App

pass Culture is a French government app launched in 2019 to promote cultural engagement among 3 million young users (ages 15–20). Users receive credits to explore books, cinema, music, theater, and more. However, initial usage was heavily skewed towards familiar media (e.g., manga, cinema).
Our goal: broaden cultural discovery through better, more diverse recommendations by incorporating DPPs in our pipeline.

2. What are Determinantal Point Processes (DPP)?

Determinantal Point Processes are probabilistic models on subsets of items given a similarity kernel K between items. The probability of selecting a subset S from the universe of all items $\Omega = \{1, \dots, n\}$ is:

$$\mathbb{P}(S) \propto \det(K_S)$$

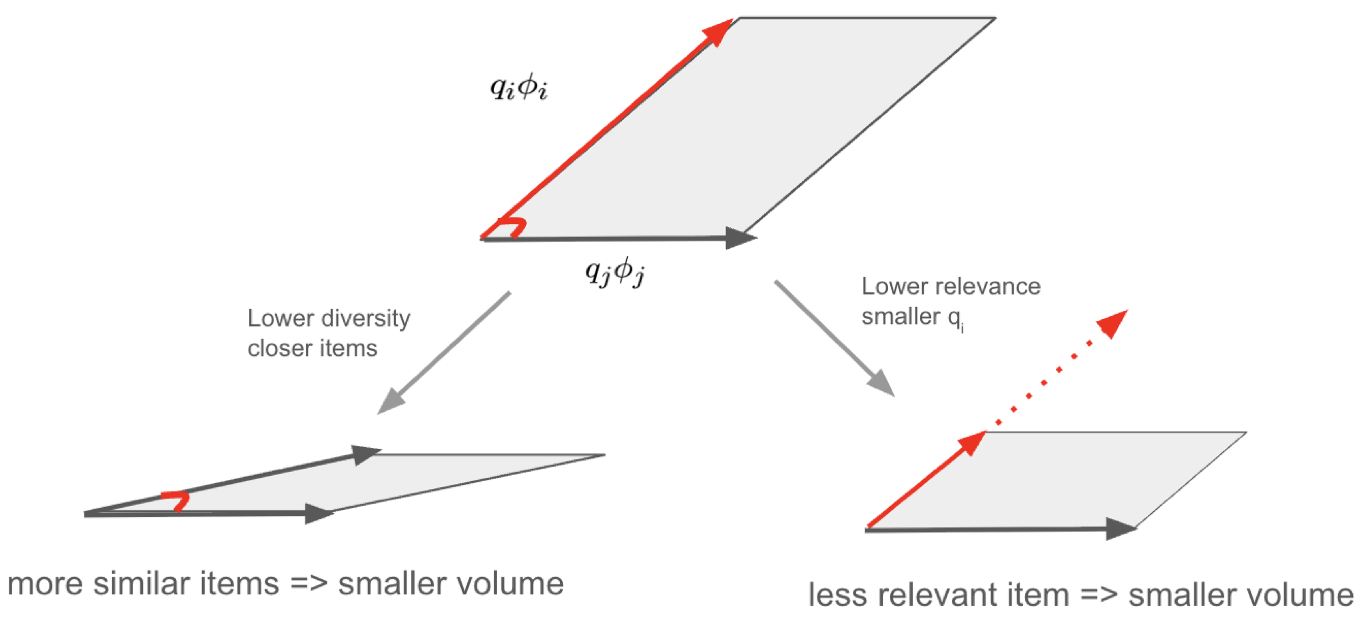
where K_S is the submatrix composed of the rows and columns indexed by S.
We consider a **quality-diversity decomposition** of the kernel:

$$K_{ij} = q_i^\lambda q_j^\lambda \phi_i^\top \phi_j$$

where:

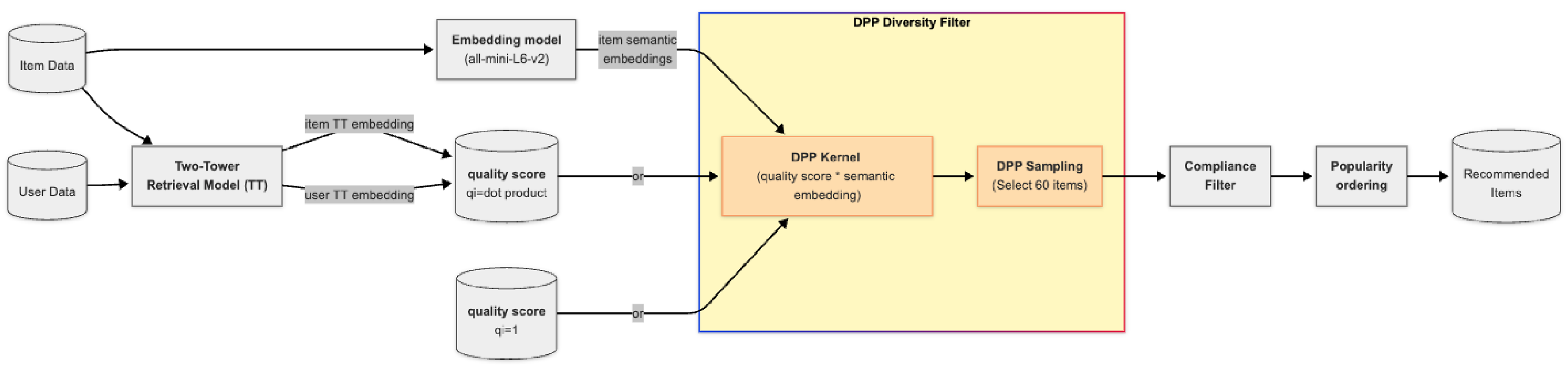
- quality $q_i > 0$ is the predicted relevance of item i
- $\phi_i \in \mathbb{R}^d$ is the semantic embedding of item i, given by all-MiniLM-L6-v2 on the item description.

The more **diverse** the feature vectors, the larger the determinant of K_S (i.e., the square of volume formed by items in S), the higher the probability of selecting S. The higher the **relevance** q_i of an item i to the user, the higher the probability of selecting i.
We use **stochastic DPP sampling**, meaning different recommendations can be served when refreshing the page, enhancing discovery.



3. Integration of DPP sampler in our recommendation pipeline

1. **Two-Tower Retrieval:** Top 1,000 items retrieved via user-item embedding similarity.
2. **DPP Filter:** 60 diverse items sampled using a DPP with personalized q_i relevance scores.
3. **Compliance + Ranking:** Filter out invalid items and rank by popularity.



4. Evaluation metrics

Relevance metrics:

- click rate (%)
- cosine similarity between user and item two-tower embeddings

Diversity:

- volume ratio of recommended items $\text{Vol}(S) = \sqrt{\det K_S}$ to the baseline items' volume
- business diversity metric: sum of novelty scores of each item with a maximum of 6.5 *(2.5 points for a new category, 2 points for a new venue type, 1 point for a new subcategory, and 0.5 points for a new venue or genre)*

5. Offline and Online Evaluation

Offline Evaluation (100K users):

Model	Cosine Similarity	Volume Ratio	Business Diversity
A (Baseline)	**0.525**	1x	2.759
B (DPP with relevance q_i^1)	0.399	24.7x	**3.404**
C (DPP no relevance q_i^0)	0.381	**28.8x**	**3.482**

Online A/B/C Test (400K+ users):

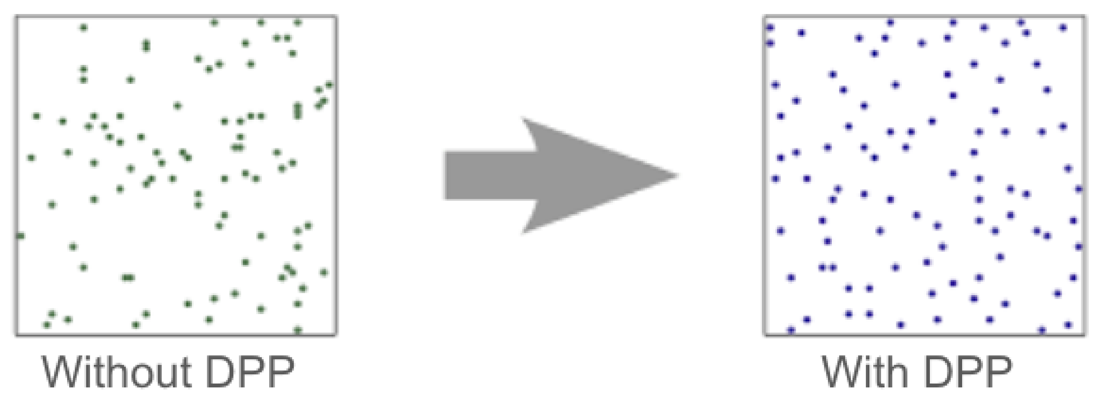
Group	Click Rate	Volume Ratio	Business Diversity
A (Baseline)	**0.54%**	1x	3.132
B (DPP with relevance q_i^1)	0.34%*	12x	**3.512***
C (DPP no relevance q_i^0)	0.29%*	**15.8x**	**3.590***

* statistically significant with $p < 0.001$

6. Key Findings

Adding a DPP filter resulted in:

- **Diversity Boost:** Up to 28.8x increase in volume diversity offline, 15.8x online
- **Business Impact:** 15% improvement in business diversity metrics
- **Trade-off:** Click rates decreased, but cultural exploration increased



7. Next Step: Tuning Relevance-Diversity

Future work includes introducing a trade-off parameter λ to better balance relevance and diversity in the kernel:

$$\log\det K_S = \lambda \underbrace{\sum_{i \in S} \log q_i}_{\text{quality}} + 2 \underbrace{\log \text{Vol}(S)}_{\text{diversity}}$$

By tuning λ, we can control the weight of relevance vs. diversity, potentially finding a Pareto frontier for optimal trade-offs.

Expected Benefits:

- Better balance between relevance and discovery
- Improved overall user satisfaction

8. Implementation Details

Technical Stack:

- Tensorflow Recommenders for the two-tower retrieval
- DPP Sampling: implementation based on **DPPy** library with efficient eigendecomposition. The algorithm complexity is $O(Nd^2 + Nk^3)$ to sample k diverse items among n in the case of a linear kernel and embeddings of size d.
- Real-time: Sub-100ms recommendation latency

8. References

- Kulesza and Taskar. *Determinantal Point Processes for Machine Learning.* Foundations and Trends, 2012.
- Wilhelm et al. *Practical Diversified Recommendations on YouTube*, CIKM 2018.
- Gautier, Guillaume, et al. "DPPy: DPP sampling with Python." Journal of Machine Learning Research 20.180 (2019)

Code:

Article:

My LinkedIn: