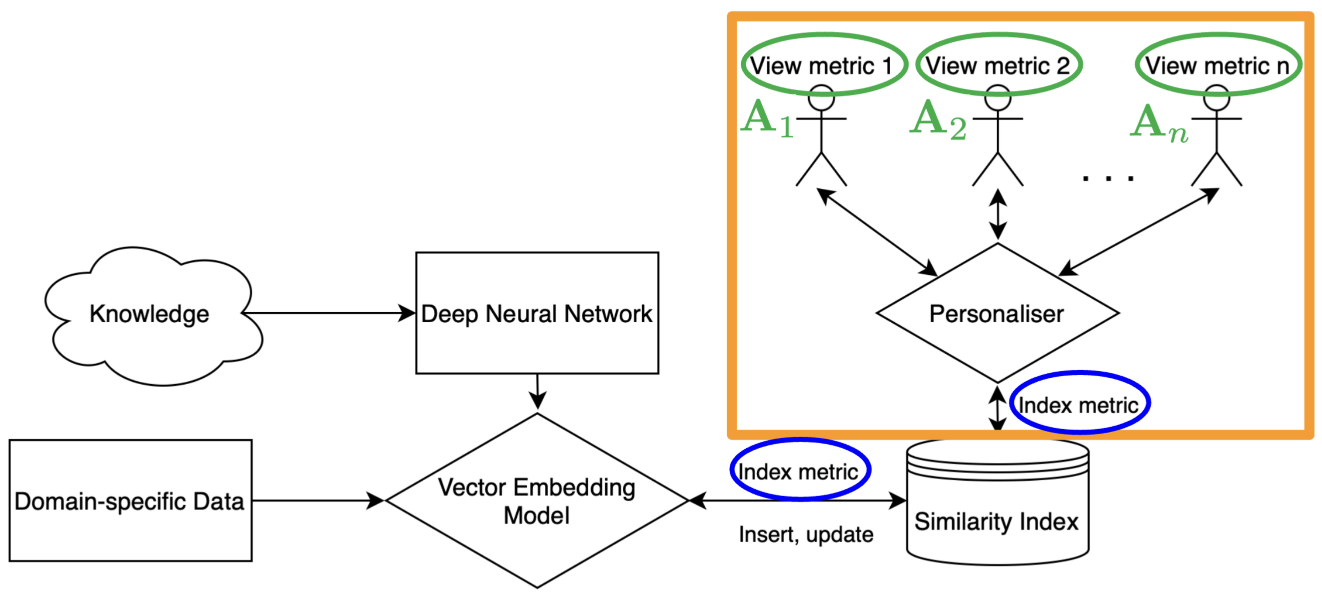
Integrating Relevance Feedback for Effective Personalisation in Vector Search

Matúš Šikyňa, Pavel Zezula
Masaryk University, Brno, Czech Republic

Previous Work & Our Proposal

Towards Personalized Similarity Search for Vector Databases (SISAP24)

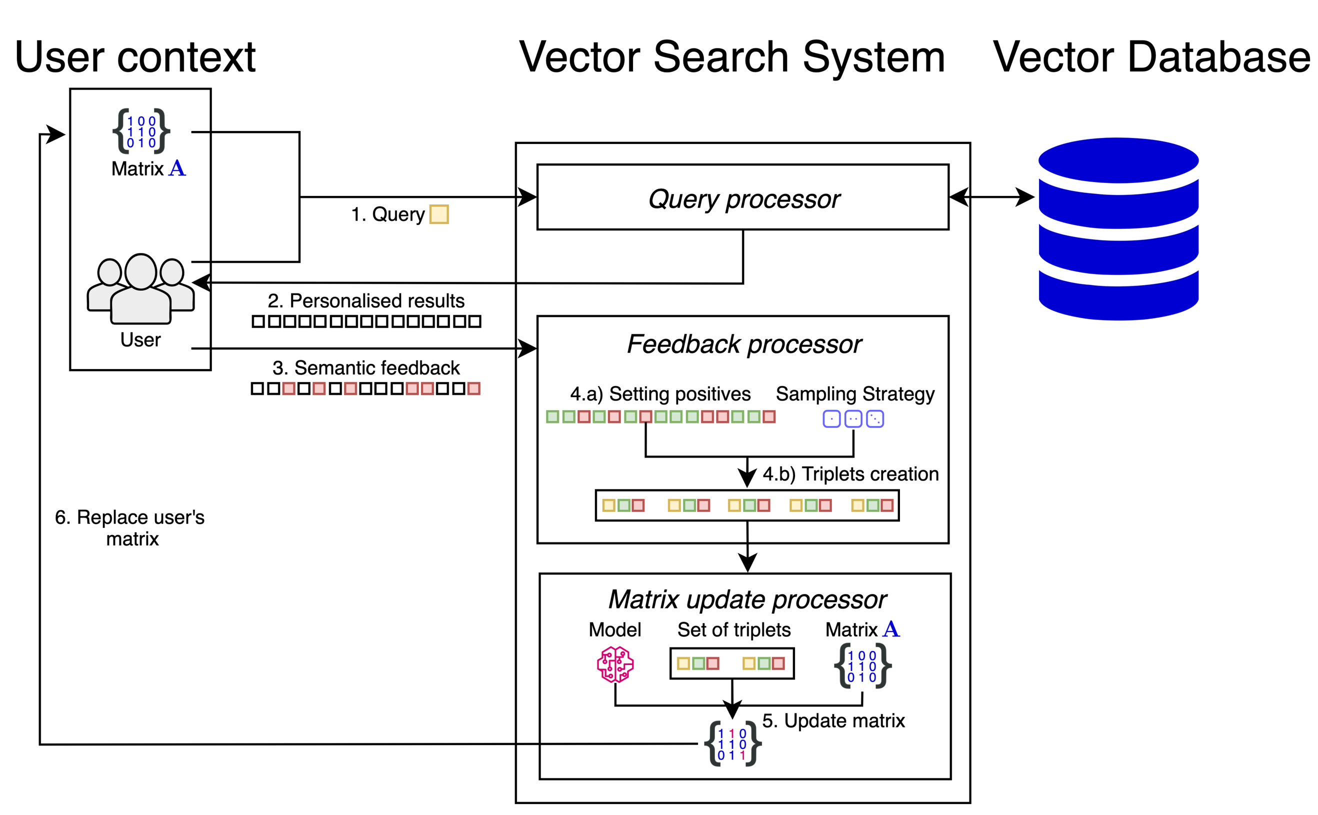
- In vector databases, traditionally, **Euclidean distance** is used for **indexing** and **querying**
- Once vectors are indexed with this **fixed distance**, partitioning and filtering remain the **same for all users**
- This contrasts with human perception of similarity, which is **subjective** and **context-dependent**
- Proposed personalized search engine:
  - **Index metric**: Euclidean distance
  - **View metric**: Mahalanobis distance
    - **A**: **personalized** matrix trained via **user feedback**



  - **Filter** with **efficient Index metric**, and **refine** with **View metric**
  - Uses the **lower bounding relationship** between Euclidean and Mahalanobis distances
  - **Limitations:** utilizes **synthetic feedback** for training the matrices and using two **metric learning** methods: ITML and SDML

Our Approach to Semantic User Feedback

- Integrating **relevance feedback**, where the user selects only **negative** (dissimilar) results; results not selected are **implicitly positive**, creating **triplets** for metric learning
- **Workflow** of the personalised vector search pipeline:



- Triplets are created using various **sampling strategies**:
  - **Strategy 1**: draws randomly one positive and one negative result to form a single triplet, runs in iterations (for more triplets)
  - **Strategy 2**: a randomly chosen negative result is paired with every available positive result, producing a set of triplets
  - **Strategy 3**: every negative result is paired with every positive result, but training is triggered only after feedback from $q \in \{1, 2, 5, 10, 20, 40\}$ queries has been accumulated
- For the practical deployment, we present three **operational scenarios**:
  - **Balanced Precision–Scaling Factor (BPSF)**: balances between a low scaling factor and high retrieval precision
  - **Minimum Scaling Factor with Acceptable Precision (MSAP)**: focuses on minimising the scaling factor to maximise efficiency, while still enforcing an acceptable precision increase (at least by 0.1)
  - **Maximum Precision (MP)**: prioritises highest attainable precision, resulting in maximum personalisation
  - Each scenario has variants: **(i) Baseline (BAS)**, **(ii) Low Learning Time (LLT)** that minimises model learning time, and **(iii) Rank Stability (RS)** that maximises rank correlation coefficients.

Experimental evaluation & results

Dataset, Queries and Models

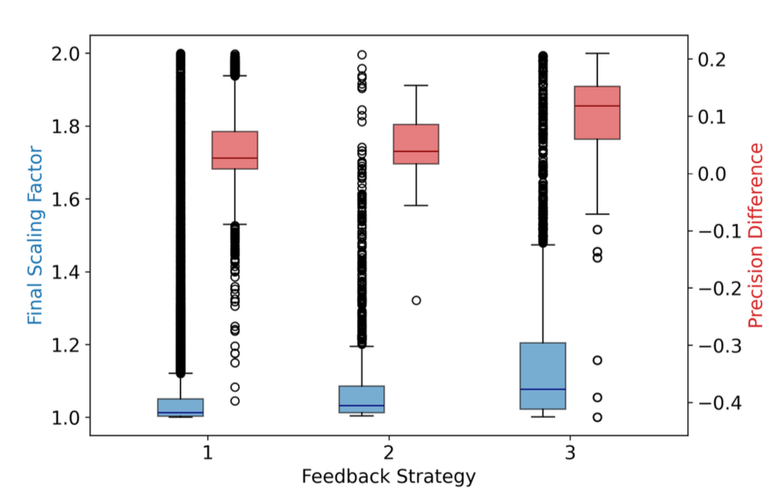
- Used the **Profiset database** of 20M images, each as a **768D CLIP ViT-L/14 embedding** (unit-normalized), enabling text–image similarity search
- Created **280 text queries**; retrieved **top-20 images per query** and manually labeled them relevant/irrelevant to build ground-truth
- Trained and tuned multiple **metric learning models** (OASIS, ITML, POLA, AROMA, OMDML, MLOML, RobustODML, SORS, AdaSORS, OPML) with extensive hyperparameter search

Evaluation Metrics

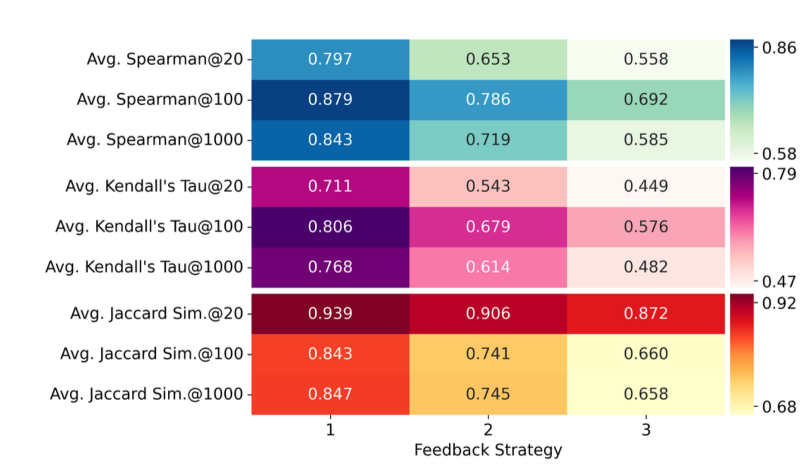
- **Final Scaling Factor (FSF)**: the scaling factor derived after learning the matrix **A** for all queries incrementally
- **Precision Difference (ΔMAP)**: the change in mean-average precision (MAP) obtained when the ranking metric is switched from Euclidean (baseline) to Mahalanobis distance parameterised by the learned user-profile matrix **A** in the ranking function; **A** is first learned using the full query set, and ΔMAP is then evaluated over that same set of queries
- **Average Learning Time (ALT)**: average time (in seconds) needed for integration of a single set of triplets in the metric learning model's learning
- **Average Spearman@k (AS@k)**: the average Spearman rank correlation (across all queries) between two ranked lists – the top k results retrieved with Euclidean distance and the top k results retrieved with the learned Mahalanobis distance – the tested k are: {20, 100, 1000}
- **Average Kendall's Tau@k (AK@k)**: like AS@k, using Kendall's Tau
- **Average Jaccard Sim@k (AJ@k)**: like AS@k, using Jaccard Similarity

Results: Sampling Strategies Comparison

- With a higher number of feedback inputs, the **global ranking using the Mahalanobis distance for ordering deteriorates against the Euclidean distance ranking**
- With a higher number of feedback inputs, both **scaling factors and precisions increase**.



(a) Scaling Factor and Precision



(b) Spearman, Kendall's Tau and Jaccard

Results: Finding Best Models for Operational Scenarios

- **OMDML** dominates when precision or rank stability matter
- **OASIS** excels when learning time is low
- **Final Scaling Factors (FSF)** are low, indicating efficiency of the approach and trained models
- **Precision Differences (ΔMAP)** are high, indicating the effectiveness of the approach and trained models

Table 1: Calculated evaluation measures for the best model given the scenario and its variant. The Model column is in the form: model and hyperparameters. The Feedback column is in the form: feedback strategy and parameters – 1:(Number of pairs, Replacement), 2:(Batch), 3:(Number of queries).

Scenario–Variant	Model	Feedback	FSF	ΔMAP	ALT	AS@1000	AK@1000
BPSF-BAS	OMDML β: 1e-3, C: 4e-3, γ: 7e-3	2 True	1.095	0.203	1.373	0.708	0.532
BPSF-LLT	OASIS C: 7e-2, $enforce\_psd$: False	1 1, False	1.147	0.103	0.001	0.621	0.456
BPSF-RS	OMDML β: 7e-2, C: 1e-3, γ: 1e-4	2 False	1.039	0.134	1.416	0.899	0.738
MSAP-BAS	MLOML nl: 2, γ: 1e-4, act: tanh, λ: 7e-6	3 5	1.017	0.102	18.098	0.863	0.693
MSAP-LLT	OASIS C: 1e-2, $enforce\_psd$: False	1 8, False	1.062	0.106	0.008	0.813	0.636
MSAP-RS	OMDML β: 1.0, C: 7e-4, γ: 4e-4	1 64, True	1.028	0.101	1.768	0.939	0.798
MP-BAS	OMDML β: 7e-1, C: 1e-2, γ: 1e-2	2 False	1.148	0.211	0.712	0.566	0.411
MP-LLT	OASIS C: 7e-2, $enforce\_psd$: False	1 2, True	1.271	0.130	0.003	0.308	0.211
MP-RS	OMDML β: 4e-2, C: 1e-2, γ: 1e-4	2 True	1.077	0.193	1.104	0.788	0.609