Matching

Lecture 7: CV models

Nikita Bozhedomov

Moscow Institute of Physics and Technology

Autumn 2023

Plan

- Types of pictures
- CV in the Matching pipeline
- Past and current research

Types of pictures

Images

Trash Predictor

Clear

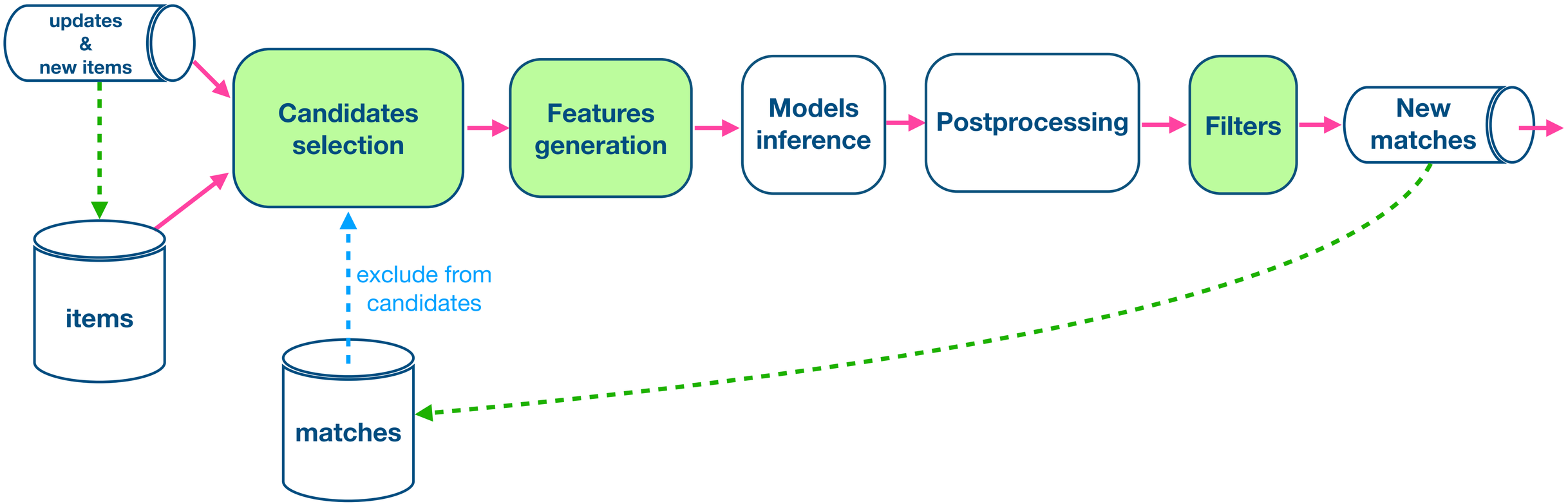
Main

Additional

Trash

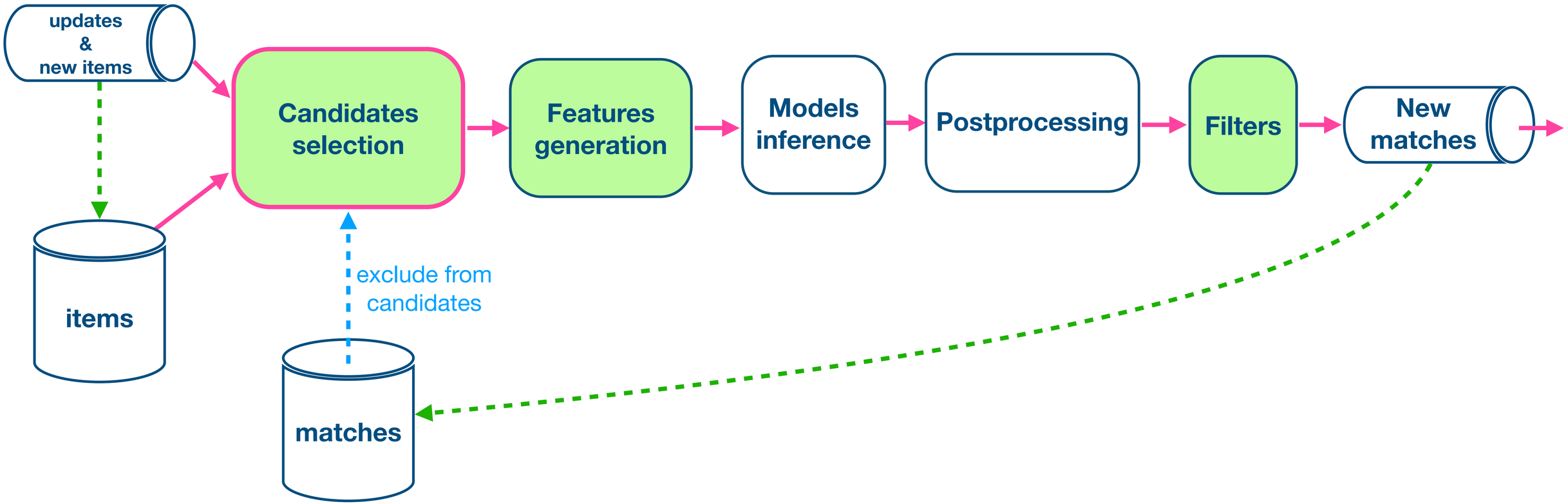
Matching Pipeline

High level design

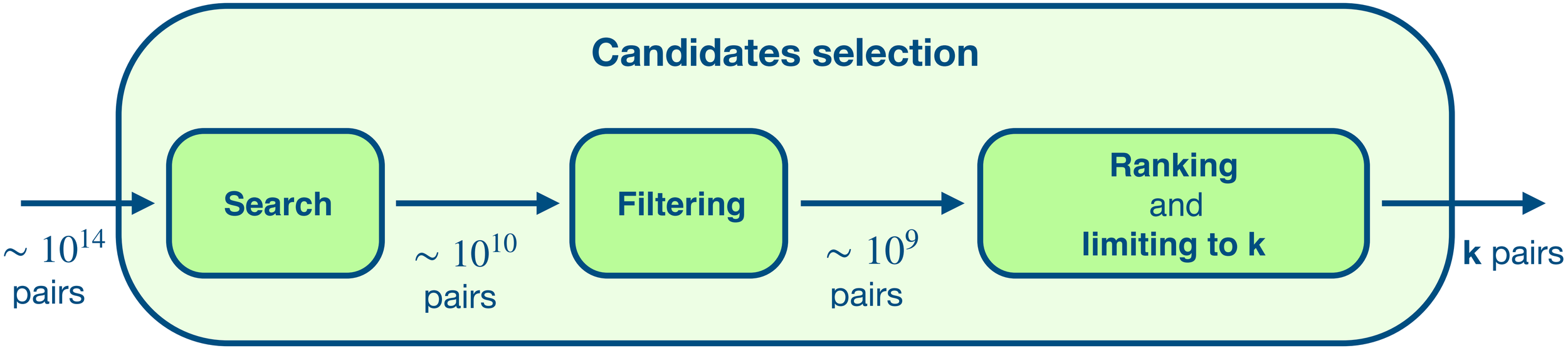


Matching Pipeline

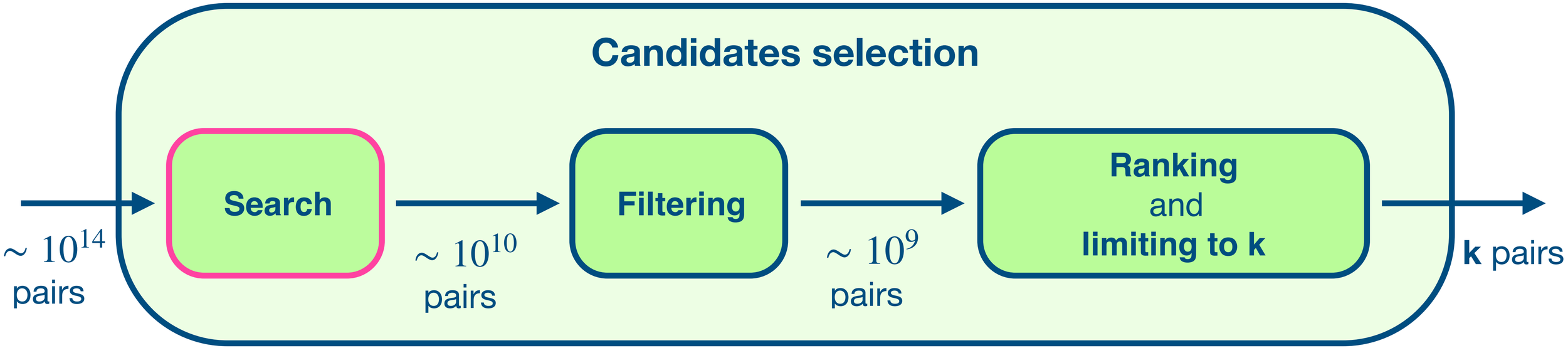
High level design



Candidates selection stages

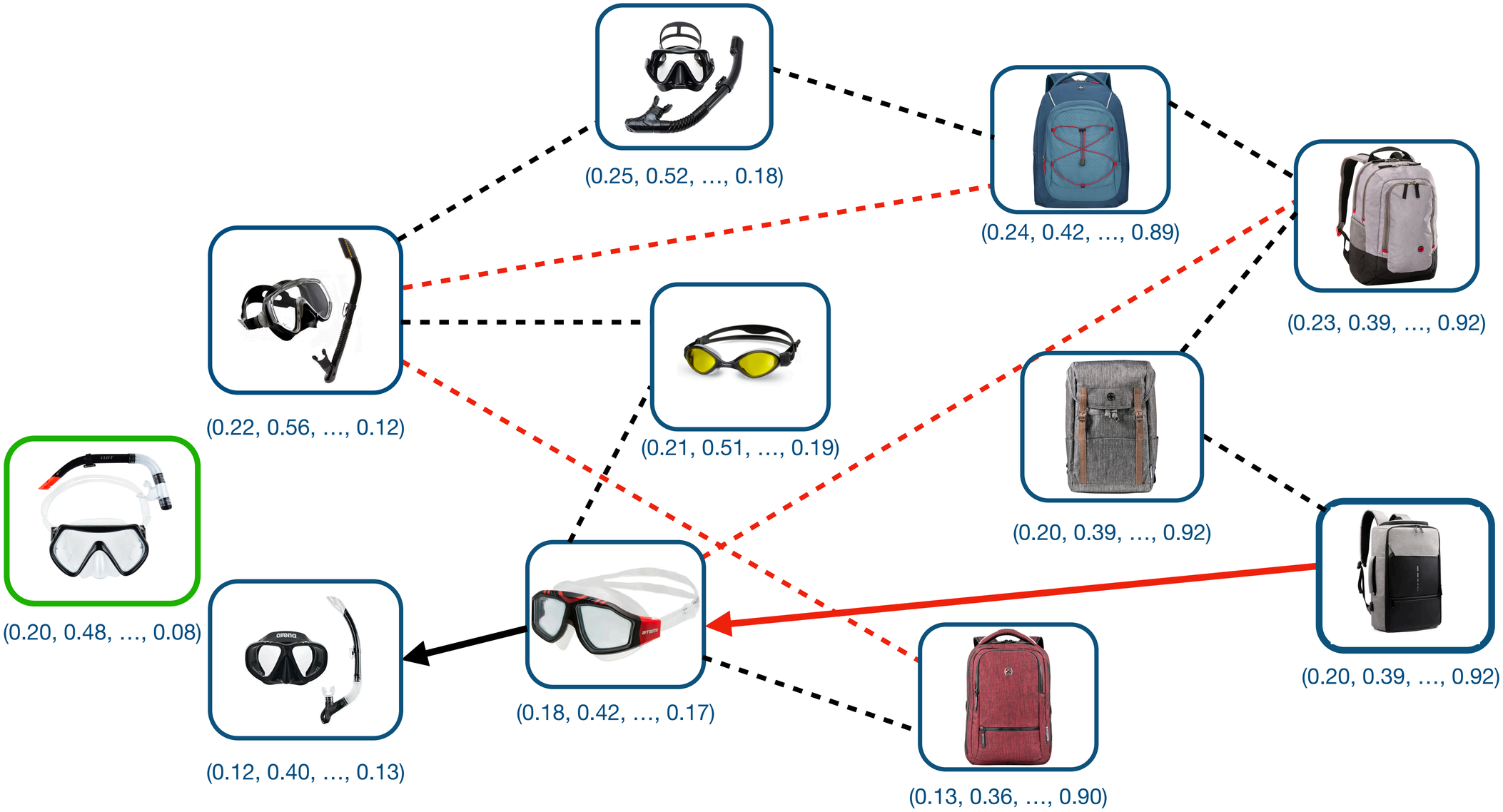


Candidates selection stages



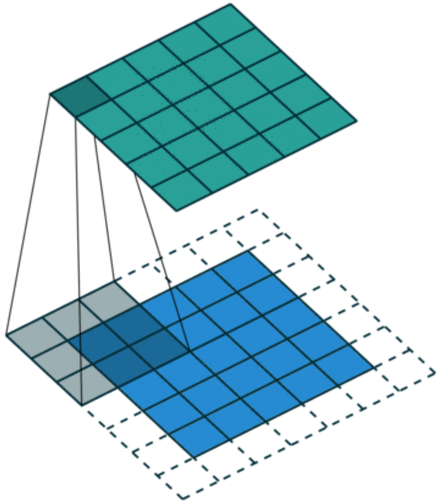
Search

HNSW (Hierarchical Navigable Small World)

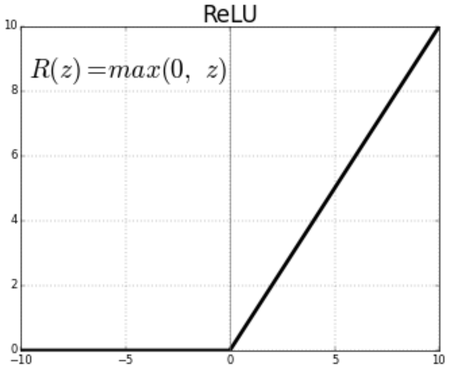


Neural Networks

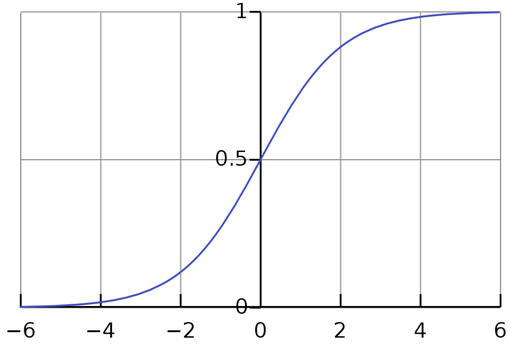
Base layers



Convolution



Nonlinear activation: ReLU



$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Nonlinear activation: Sigmoid

$$\mu_{\mathcal{B}} \leftarrow \frac{1}{m}\sum_{i=1}^{m} x_i$$

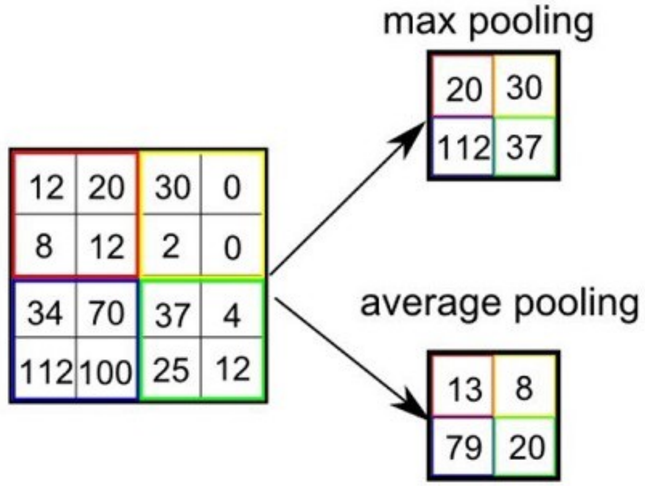
$$\sigma_{\mathcal{B}}^2 \leftarrow \frac{1}{m}\sum_{i=1}^{m}(x_i - \mu_{\mathcal{B}})^2$$

$$\widehat{x}_i \leftarrow \frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}}$$

$$y_i \leftarrow \gamma \widehat{x}_i + \beta \equiv \mathbf{BN}_{\gamma,\beta}(x_i)$$

Batch Normalization

[S. Ioffe , C. Szegedy, 2015]



Pooling

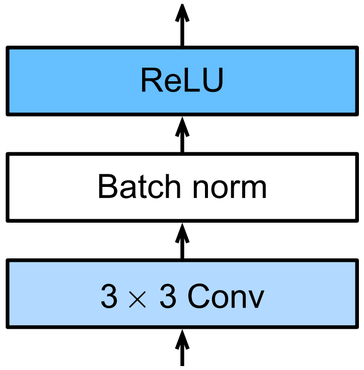
$$\sigma : \mathbb{R}^K \mapsto (0,1)^K$$

$$\sigma(\mathbf{z})_i = \frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}}$$

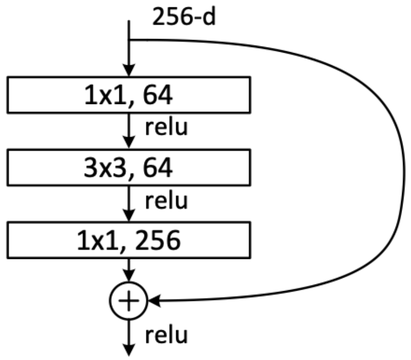
Softmax

Neural Networks

ResNet. Architecture



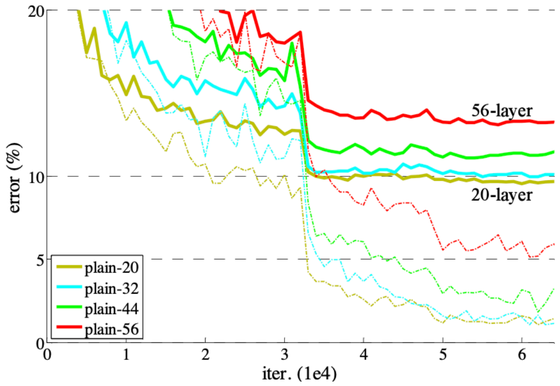
Part of residual-block

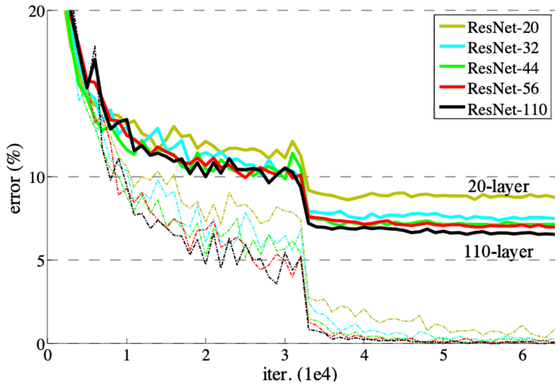


Residual-block

layer name	output size	18-layer	34-layer	50-layer	101-layer	152-layer
conv1	112×112	7×7, 64, stride 2				
conv2_x	56×56	3×3 max pool, stride 2				
		$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 64 \\ 3\times3, 64 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 64 \\ 3\times3, 64 \\ 1\times1, 256 \end{bmatrix}\times3$
conv3_x	28×28	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 128 \\ 3\times3, 128 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times4$	$\begin{bmatrix} 1\times1, 128 \\ 3\times3, 128 \\ 1\times1, 512 \end{bmatrix}\times8$
conv4_x	14×14	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 256 \\ 3\times3, 256 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times6$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times23$	$\begin{bmatrix} 1\times1, 256 \\ 3\times3, 256 \\ 1\times1, 1024 \end{bmatrix}\times36$
conv5_x	7×7	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times2$	$\begin{bmatrix} 3\times3, 512 \\ 3\times3, 512 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$	$\begin{bmatrix} 1\times1, 512 \\ 3\times3, 512 \\ 1\times1, 2048 \end{bmatrix}\times3$
	1×1	average pool, 1000-d fc, softmax				
FLOPs		1.8×10^9	3.6×10^9	3.8×10^9	7.6×10^9	11.3×10^9

Architecture





[K. He et al., 2015]

Neural Networks

ResNet. Training

Dataset

	old	new
positives	0.7M	13.3M
negatives		3.9M

NT-Xent loss

[Chen T. et al., 2020]

$$\ell_{i,j} = -\log \frac{\exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_j)/\tau)}{\sum_{k=1}^{2N} \mathbb{1}_{[k \neq i]} \exp(\text{sim}(\boldsymbol{z}_i, \boldsymbol{z}_k)/\tau)}$$

$$\text{sim}(\mathbf{u}, \mathbf{v}) = \mathbf{u}^T\mathbf{v}/||\mathbf{u}||||\mathbf{v}||$$

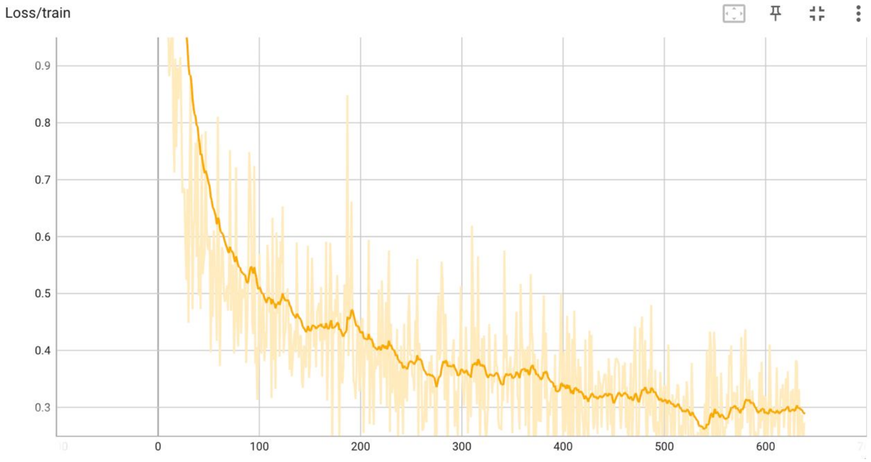
Optimization

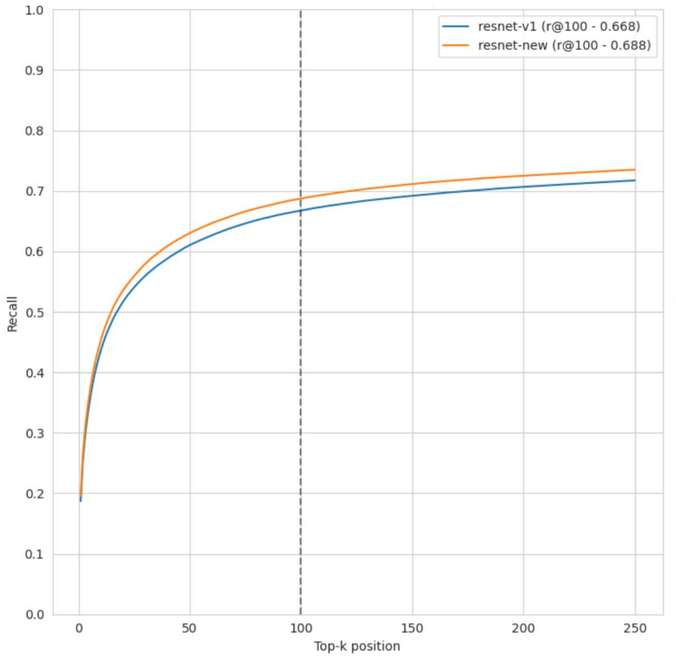
Stochastic Gradient Discent

$$w := w - \eta \nabla Q(w) = w - \frac{\eta}{n} \sum_{i=1}^{n} \nabla Q_i(w)$$

$$lr = \eta = 0.01, \quad batch\_size = n = 128$$

$$n\_process = 4, \quad ratio = 1\frac{GPU}{process}$$





Neural Networks

Previous research

$$BN \rightarrow AdaptiveGradientClipping$$

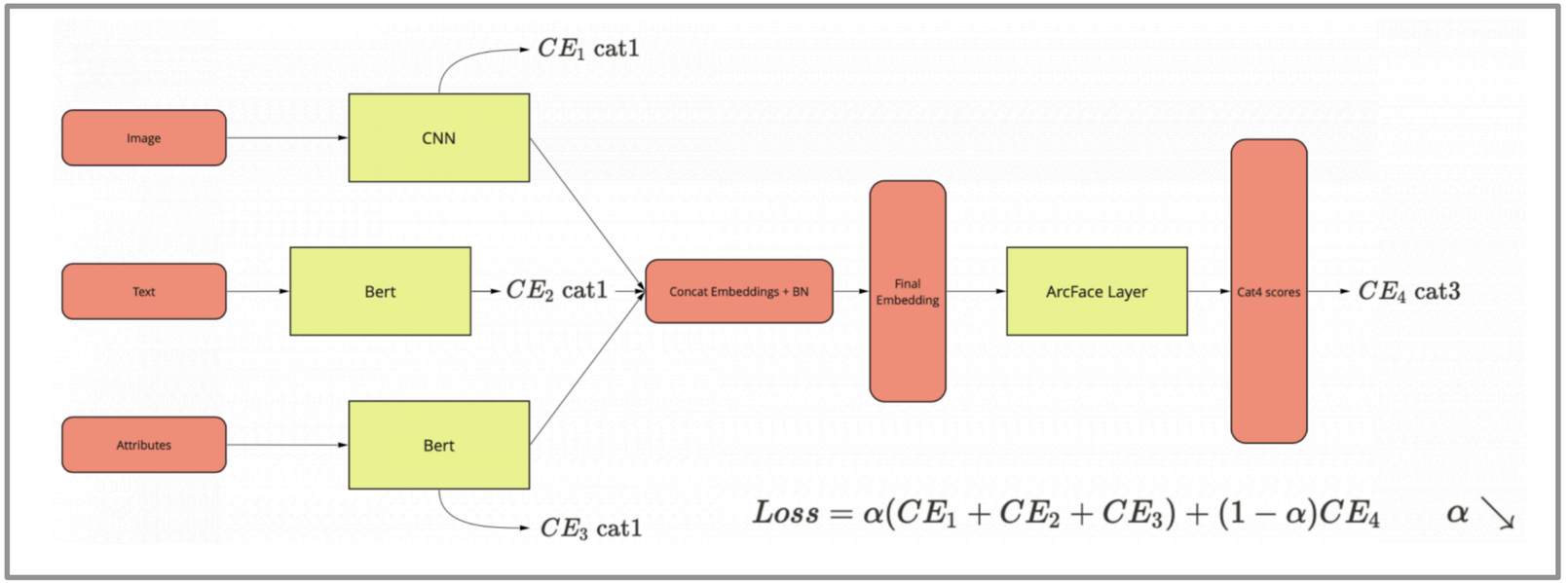
$\|W^l\|_F = \sqrt{\sum_{i,j}(W^l_{i,j})^2}$ - Frobenius norm

$$G_i^\ell \rightarrow \begin{cases} \lambda \frac{\|W_i^\ell\|_F^\star}{\|G_i^\ell\|_F} G_i^\ell & \text{if } \frac{\|G_i^\ell\|_F}{\|W_i^\ell\|_F^\star} > \lambda, \\ G_i^\ell & \text{otherwise.} \end{cases}$$

$$\|W_i\|_F^\star = \max(\|W_i\|_F, \epsilon)$$

Normalized-Free ResNets

[A. Brock et al., 2021]



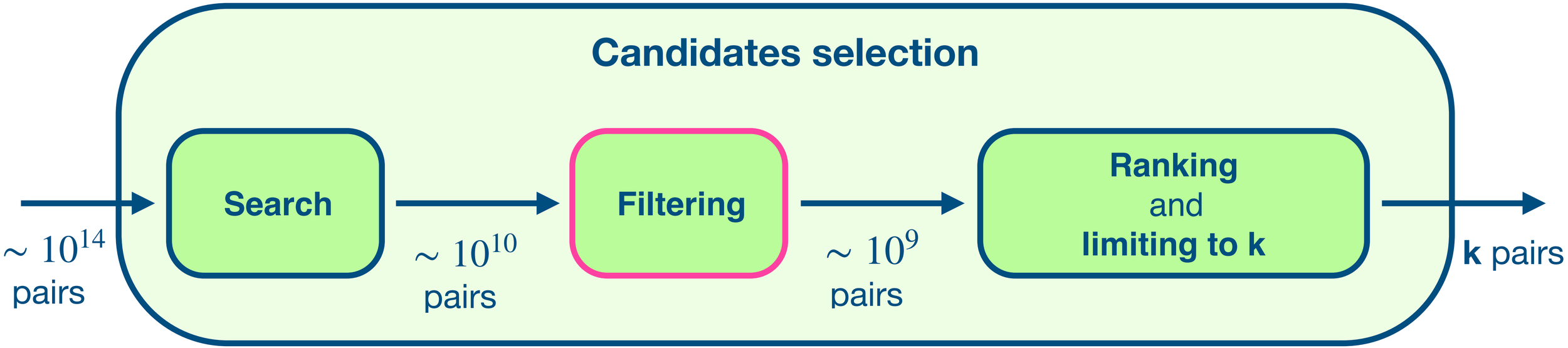
Prod2Vec

[Ozon Matching, 2021]

$$L_3 = -\log \frac{e^{s\cos(\theta_{y_i}+m)}}{e^{s\cos(\theta_{y_i}+m)} + \sum_{j=1,j\neq y_i}^{N} e^{s\cos\theta_j}}$$

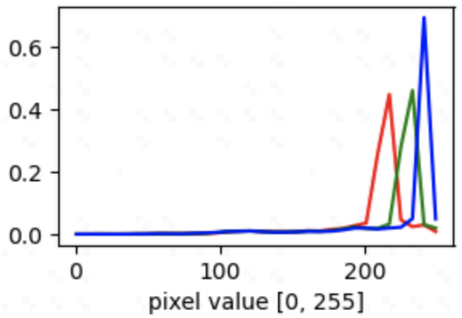
ArcFace loss
[J. Deng et al., 2015]

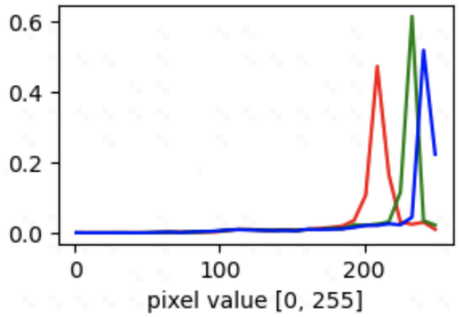
Candidates selection stages

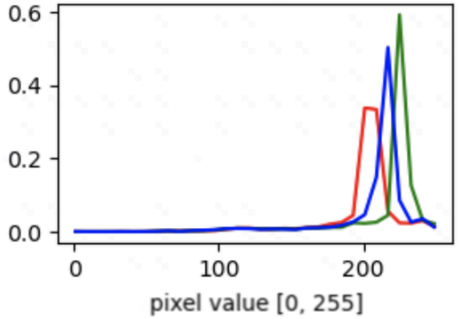


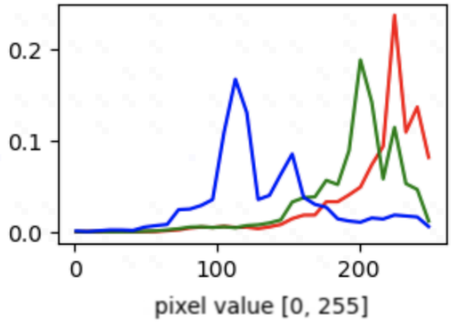
Filtering

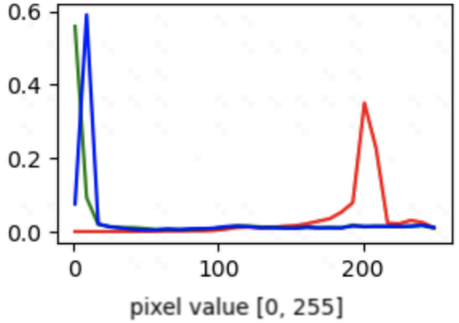
Filter by color









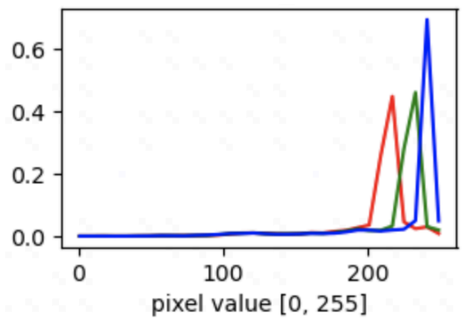


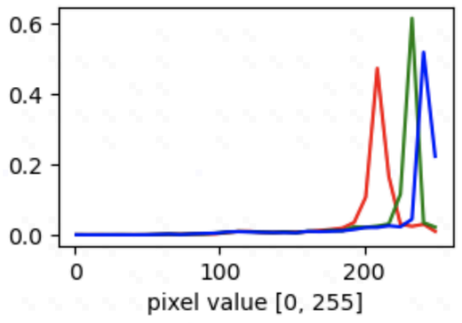
Jeffreys divergence = $D_{\mathrm{KL}}(P \parallel Q) + D_{\mathrm{KL}}(Q \parallel P)$

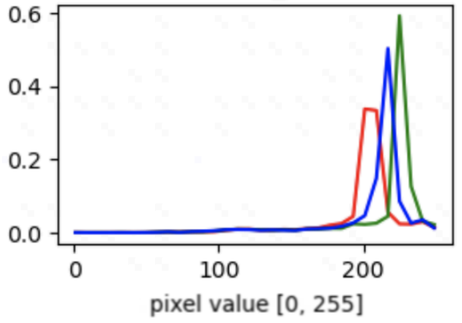
$$D_{\mathrm{KL}}(P \parallel Q) = \sum_{x \in \mathcal{X}} P(x) \log\left(\frac{P(x)}{Q(x)}\right)$$

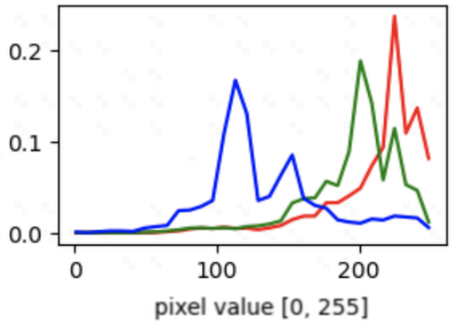
Filtering

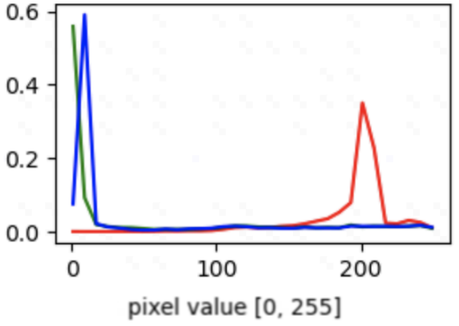
Filter by color











Divergence = 0.34

Divergence = 2.07

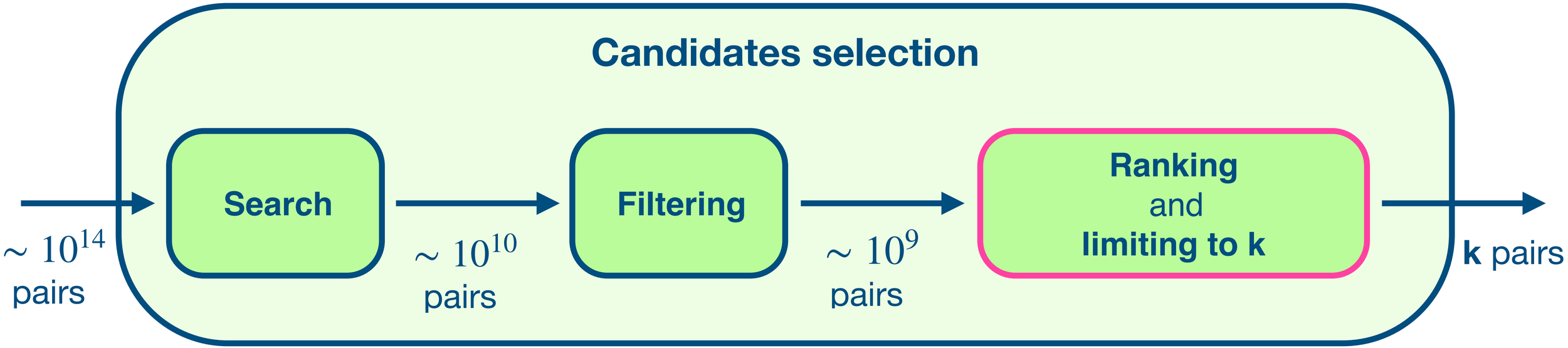
Divergence = 2.72

Divergence = 5.91

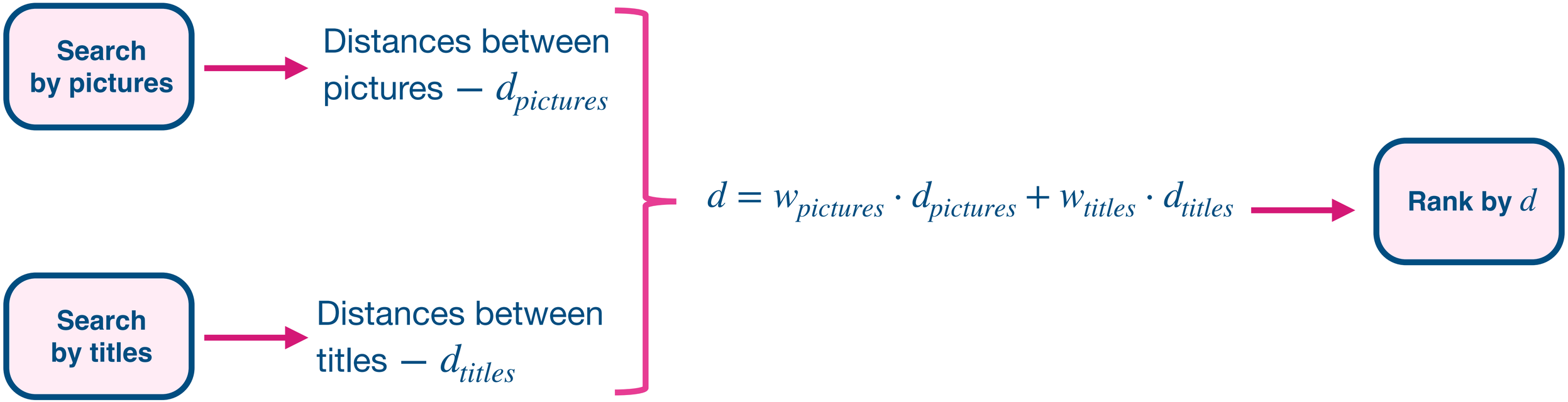
? Possible match

X Not match

Candidates selection stages



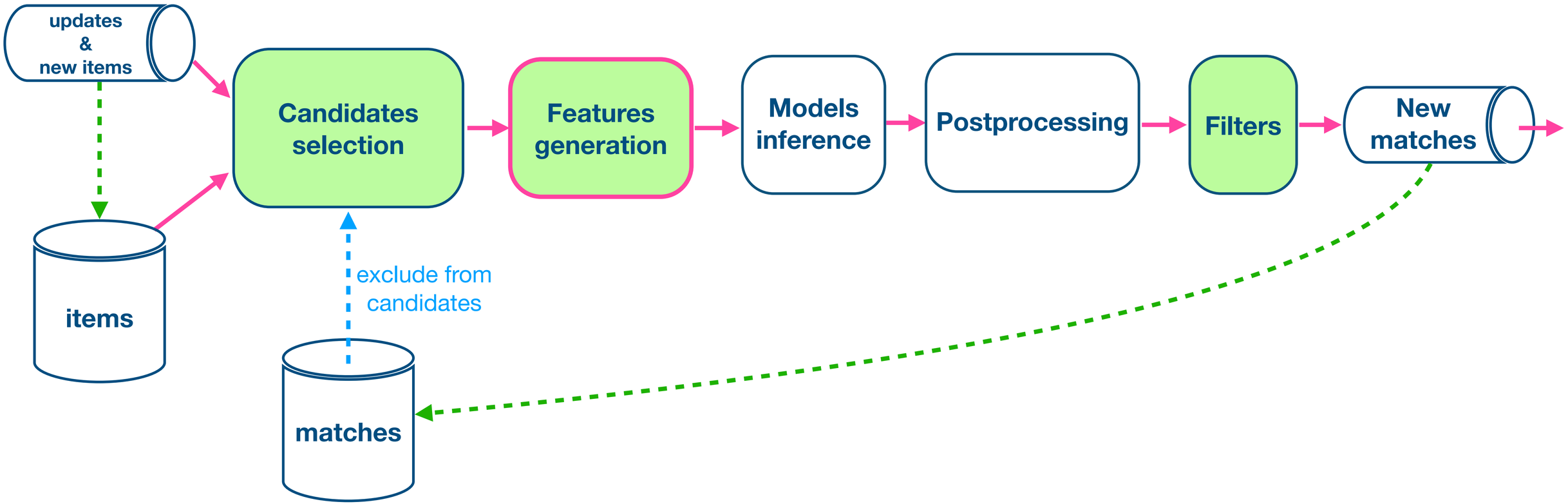
Ranking



$d_{pictures}$ - cosine distance between ResNet50 embeddings

Matching Pipeline

High level design



Features

Embeddings based

Ozon - Competitor matrix

Pairwise distance matrix						
	5	7	4	1	3	2
	6	6	5	4	5	6
	0	2	4	5	4	2
	2	0	6	7	5	3
	2	3	1	2	1	0
	3	6	4	5	4	4

Statistics

Ozon to Ozon →
1. Mean
2. Min

Ozon to Comp →
1. All to All
2. Main to All
3. All to Main
4. Max-min aggregations
5. Position-based

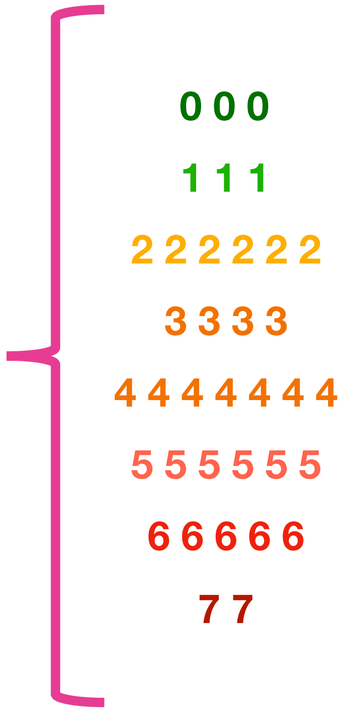
Comp to Comp →
1. Mean
2. Min

Features

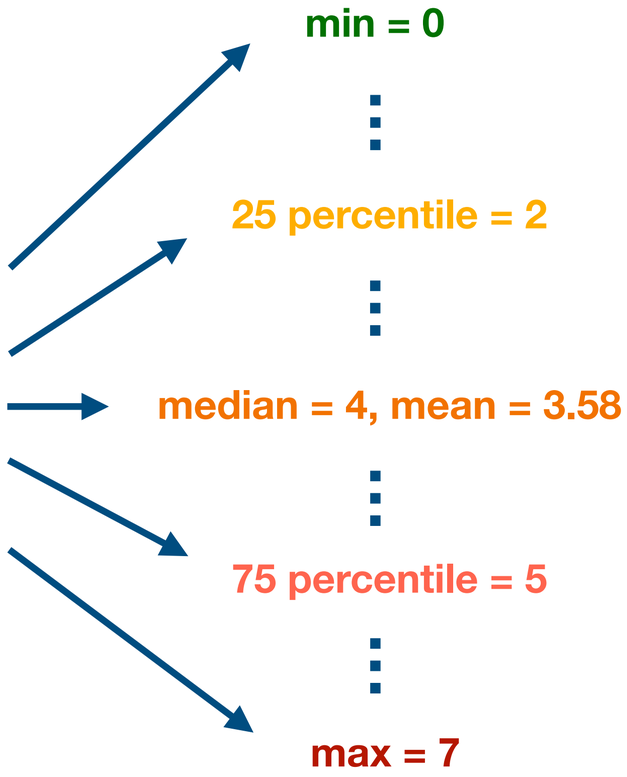
Embeddings based

Ozon - Competitor matrix

Pairwise distance matrix						
	5	7	4	1	3	2
	6	6	5	4	5	6
	0	2	4	5	4	2
	2	0	6	7	5	3
	2	3	1	2	1	0
	3	6	4	5	4	4

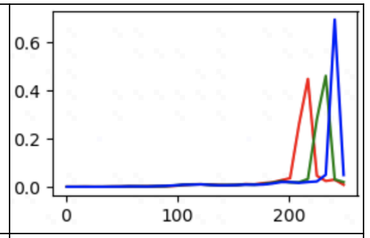
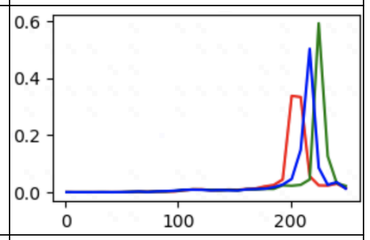
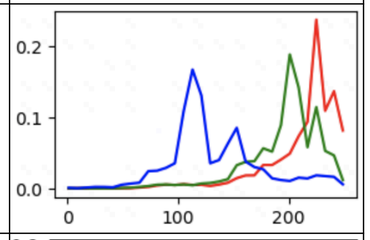
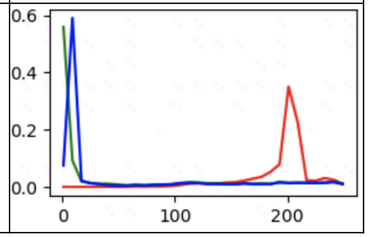


Statistics



Features

RGB-densities based

Image	RGB-densities
	
	
	
	

Jeffreys divergence: $D_{\mathrm{KL}}(P \parallel Q) + D_{\mathrm{KL}}(Q \parallel P)$

Hellinger distance: $\frac{1}{\sqrt{2}} \left\| \sqrt{P} - \sqrt{Q} \right\|_2 .$

L2-norm of diff: $\|P - Q\|_2$

Weighted distance between peaks:

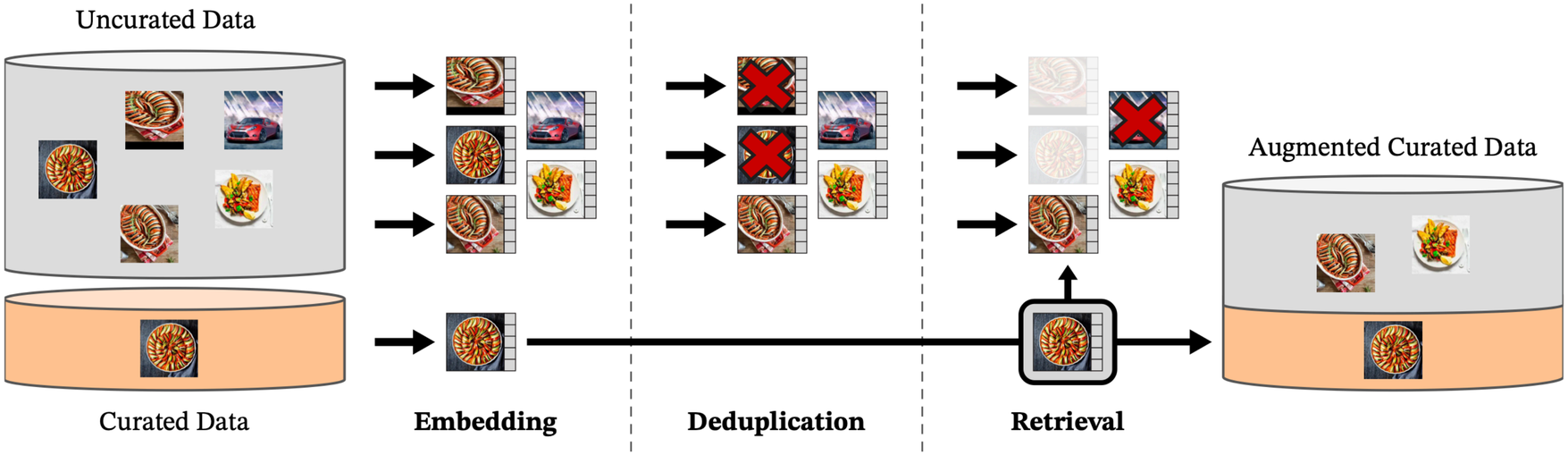
$$i_P = argmax(P), \quad i_Q = argmax(Q)$$

$$\frac{max(P[i_P], Q[i_Q])}{min(P[i_P], Q[i_Q])} * \frac{|i_P - i_Q|}{bins(P)}$$

Neural Networks

Current research

DINOv2
[M. Oquab et al., 2023]



Neural Networks

Current research

Print On Demand

Neural Networks

Current research

Background + info

Thanks for your attention