Presented by:

Alphin
Tarang

Matryoshka Representation Learning

Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham Kakade, Prateek Jain, Ali Farhadi

Motivation

Embeddings used everywhere.

Cost grows with:
dimension d
dataset size N
labels L

At web scale, utilization cost overshadows feature computation cost.

Problem

The Problem

Embedding are rigid and fixed in size

One Size for everything always

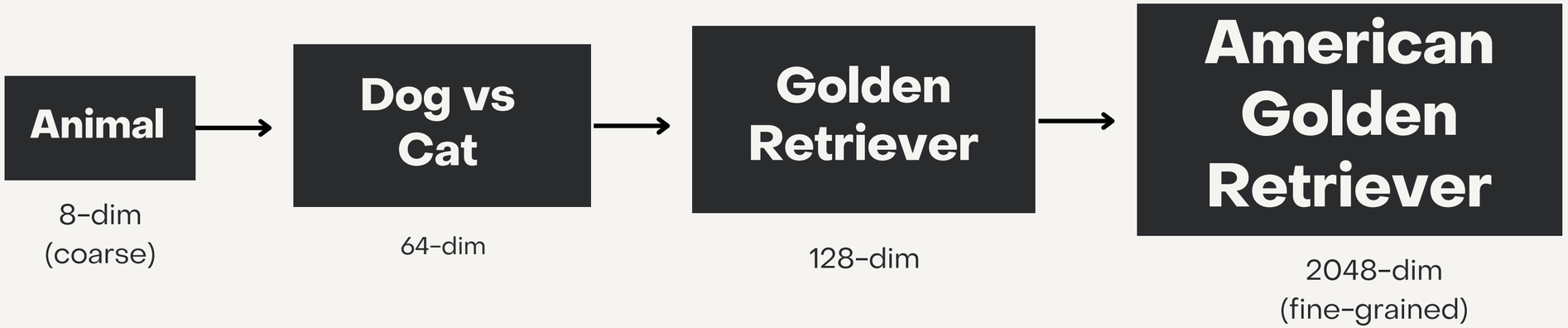
What it means

Cannot adapt to:
- compute
- accuracy requirements
- ‘Single fixed representation’
- ‘No adaptability’

Why it matters

Existing solutions require multiple models, expensive forward passes, or significant accuracy drops

Background/ Intuition

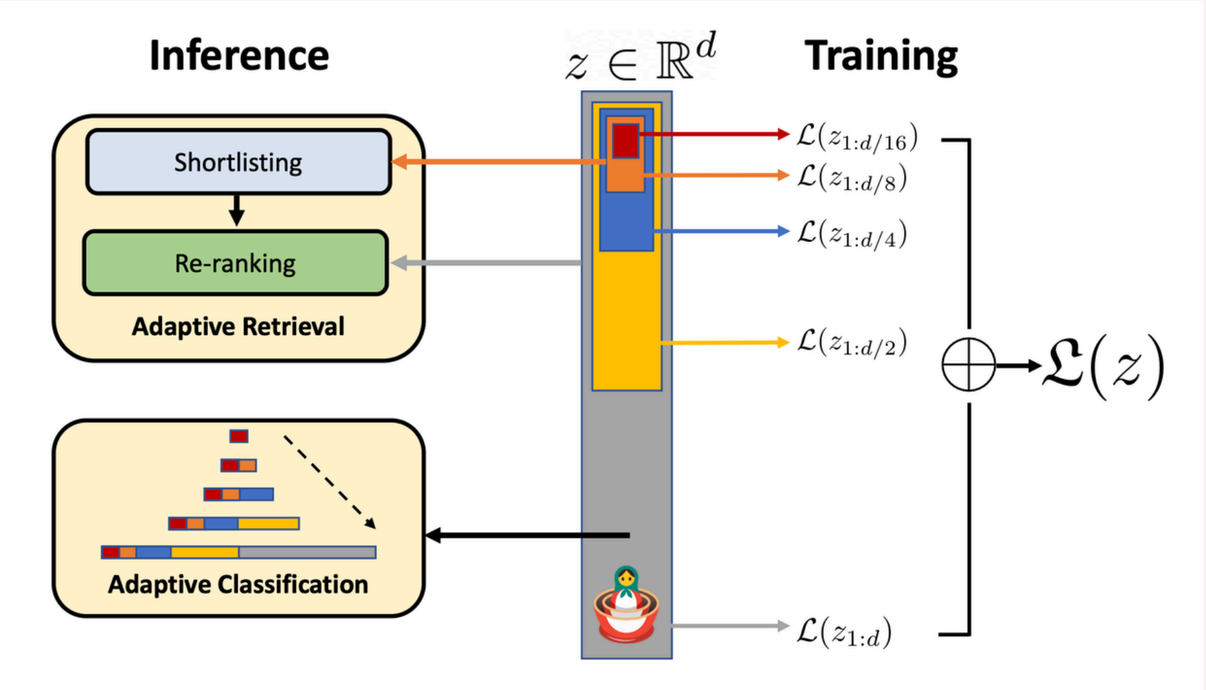


- Learned embeddings power search, classification, and retrieval is nearly in every modern ML system.
- At web scale, the cost of using embeddings grows linearly with dimension — making smaller embeddings for easy queries much cheaper.
- Standard models produce only one fixed-size embedding, forcing every query to pay the full cost regardless of difficulty.

Can one model produce embeddings that work well at any size?

Matryoshka Representation Learning

- One embedding → multiple usable sizes.
- First m dims of z form an accurate m-dim representation.
- Like Russian nesting dolls — each size lives inside the next.
- No extra inference cost — just slice the vector.



Formal Definition

Full embedding from neural network

$$z \in \mathbb{R}^d$$

First m dimensions, independently usable

$$z_{1:m} \in \mathbb{R}^m$$

Goal: make every prefix $z_{1:m}$ as good as a dedicated m-dim model

Key Equation

$$\min_{\{W^{(m)}\}_{m\in\mathcal{M}},\,\theta_F} \frac{1}{N}\sum_{i\in[N]}\sum_{m\in\mathcal{M}} c_m \cdot L\left(W^{(m)} \cdot F(x_i;\theta_F)_{1:m}; y_i\right)$$

- One loss per nested size per training example.
- All losses sum together, backbone updated by all simultaneously.

$W^{(m)}$ - Classifier weight matrix for dimension size m

θ_F - All learnable weights inside the backbone network

c_m - Importance weight per dimension (default = 1)

$F(x_i;\theta_F)$ - Run image xi through the neural network

y_i - The correct label for image i

Nested Dimensions

M = {8, 16, 32, 64, 128, 256, 512, 1024, 2048}

- Only |M| ≤ log(d) sizes explicitly optimized.
- ResNet50: d=2048, ViT/BERT: d=768.

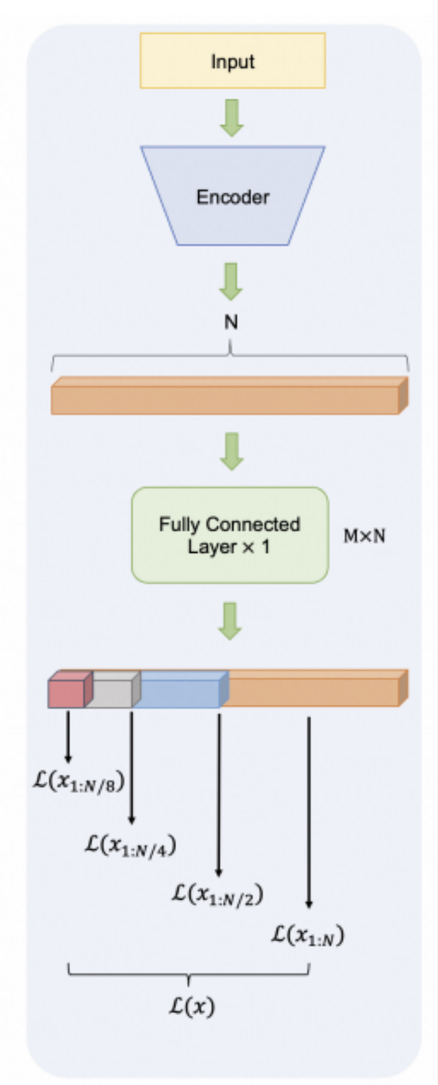
MRL-E

- Standard MRL: separate $W^{(m)}$ per dimension — one classifier per size.
- MRL-E: $W^{(m)} = W_{1:m}$ — one shared matrix, use first m columns.
- Benefit: half the memory cost.
- Cost: within 1% accuracy at most sizes.
- BERT uses MRL-E automatically — weight tying already exists.

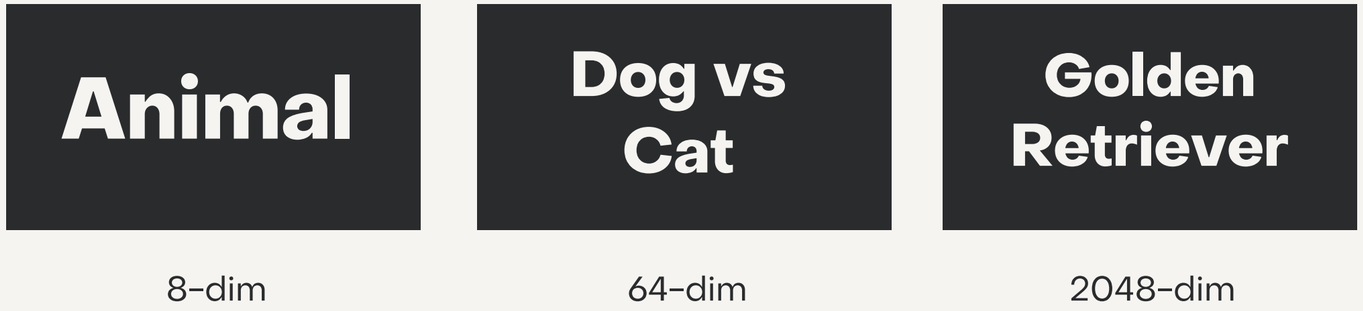
$$W^{(m)} = W_{1:m}$$

Training Pipeline

- The encoder produces a single full-dimensional embedding z for each input.
- The embedding is sliced at each nested size (8, 16, 32 ... 2048), and a separate classifier computes a loss at each size.
- All losses are summed and used in one backward pass to update the entire network simultaneously.



Why it works



- Standard training spreads information randomly across all dimensions, with no guarantee that any prefix is meaningful on its own.

- MRL's nested losses force the model to pack the most important information into early dimensions first, creating a coarse-to-fine hierarchy.

- As a result, 8 dimensions capture broad categories, while higher dimensions progressively add the fine-grained detail needed for harder distinctions.

Datasets & Models

- ResNet50 on ImageNet-1K (supervised vision)
- ViT-B/16 on JFT-300M (web-scale vision)
- ALIGN: ViT + BERT on 1.8B image-text pairs
- BERT on Wikipedia + BooksCorpus (language)
- ImageNet-4K: new benchmark introduced by this paper — 4,202 classes, ~4.2M images

ROBUSTNESS DATASETS

- ImageNet-V2 — 10K images, natural distribution shift
- ImageNet-R — 30K artistic renditions, 200 classes
- ImageNet-A — 7.5K adversarial real-world images
- ImageNet-Sketch — 50K sketches, all 1K classes

Related Work

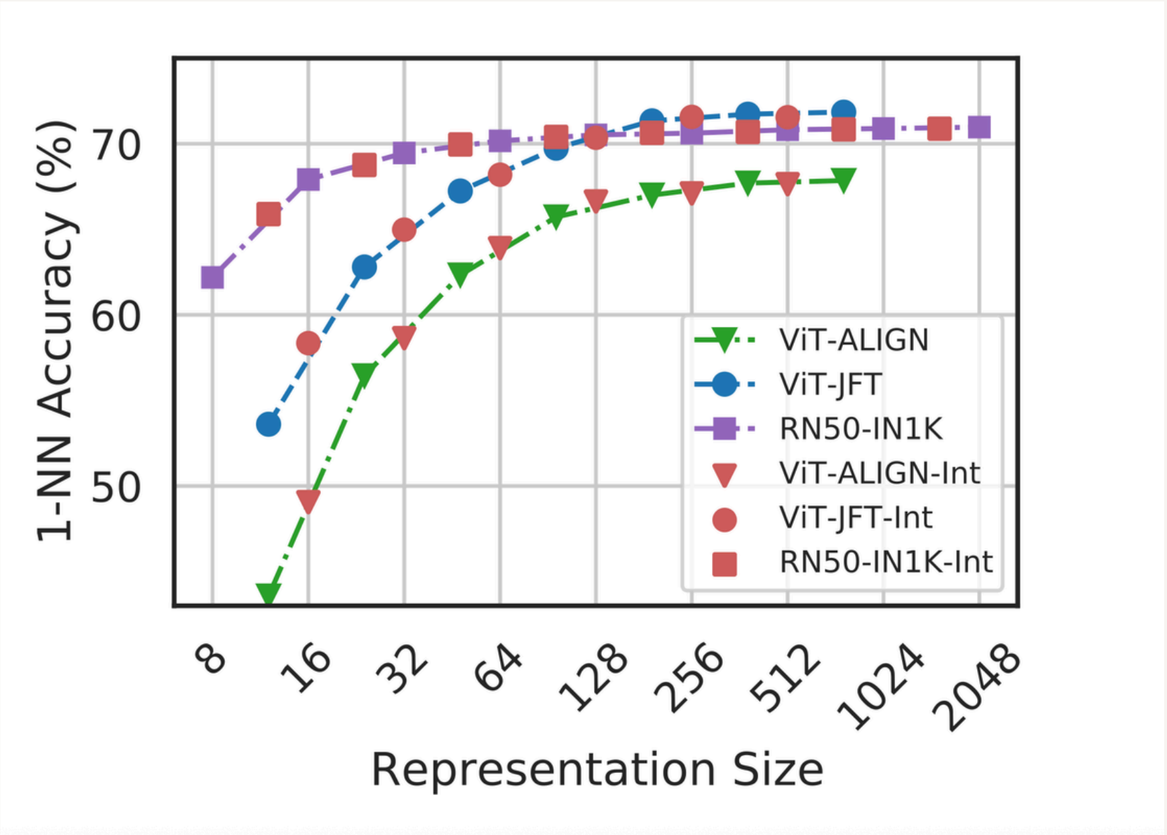
Prior approaches to flexible embeddings each have a critical limitation that MRL is designed to address.

Approach	Problem
Nested Dropout (Rippel et al.)	Optimizes O(d) sizes — too expensive at scale
Slimmable Networks	Needs separate forward pass per size
SVD / PCA	Post-hoc, collapses below 256-dim
ANNS (HNSW etc.)	Tackles N scaling, not d scaling

MRL avoids all of these issues by optimizing only O(log d) nested sizes within a single forward pass, at no additional inference cost.

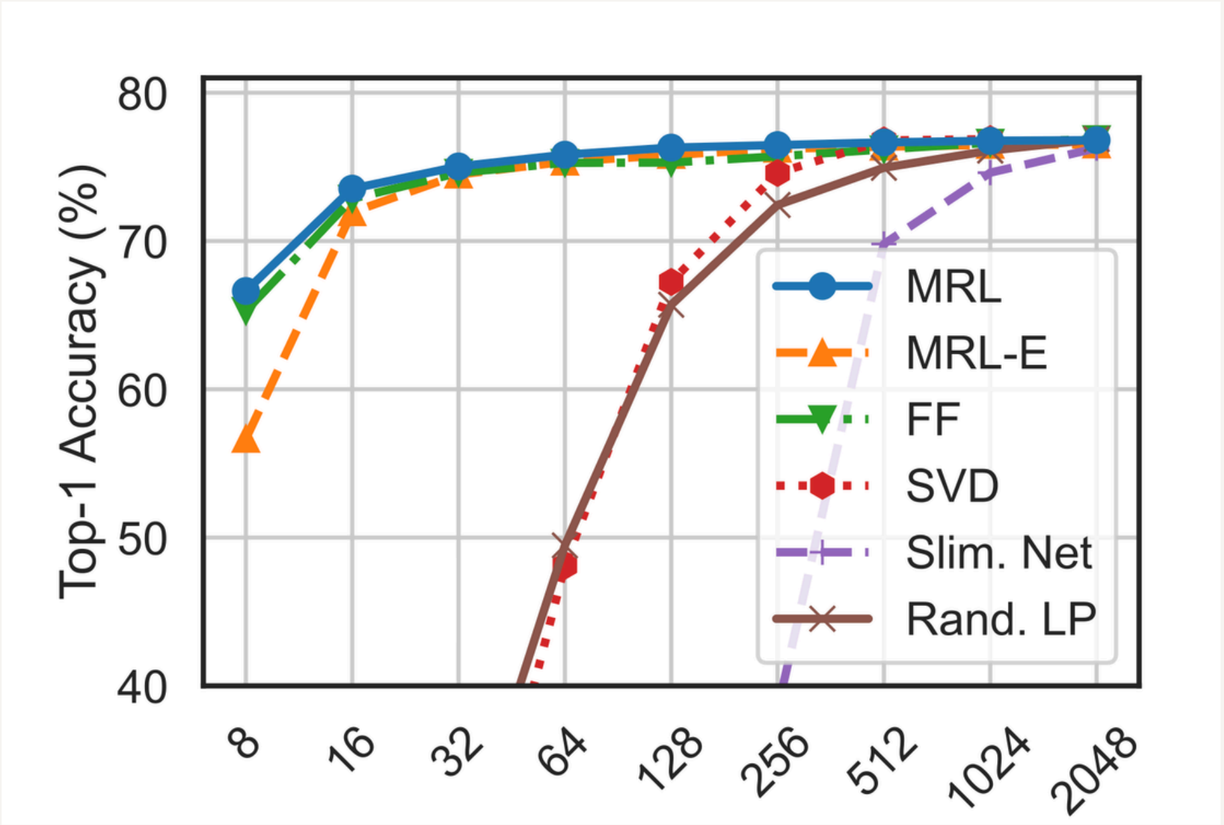
Interpolation

- MRL is explicitly trained at only 9 sizes: {8, 16, 32.....2048}.
- Intermediate sizes such as 24 and 48 dimensions are never directly optimized, yet still perform well.
- For example, 24-dim achieves 68.76% 1-NN accuracy — smoothly between 16-dim (67.91%) and 32-dim (69.46%).
- This means practitioners can use any embedding size they need, not just the ones trained explicitly.



Classification Results

- MRL matches FF at every dimension.
- SVD and Slimmable collapse below 256-dim — MRL holds up.
- MRL-E within 1% across all sizes.



Web-Scale Results

jFT-300M (ViT-B/16)

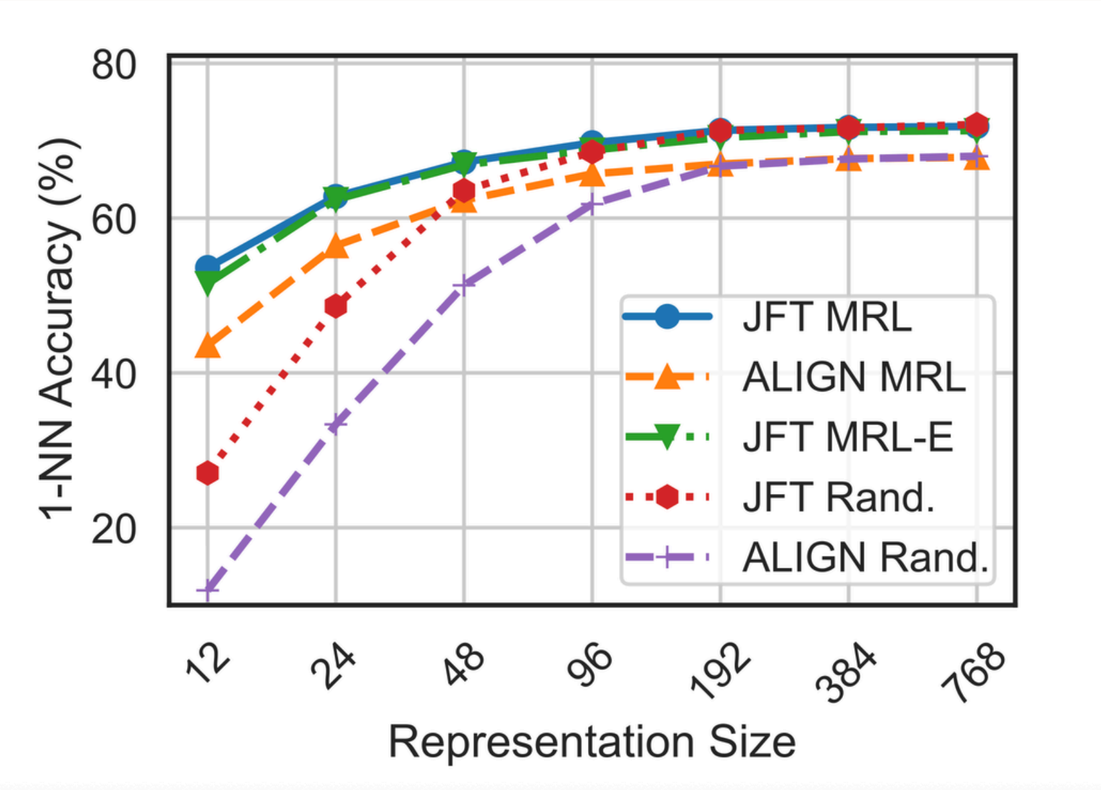
- Outperforms random features at all dims.
- Big gains at lower dimensions.

ALIGN (ViT + BERT, 1.8B pairs)

- Improves accuracy at nearly all dims.
- Zero-shot robustness preserved.

BERT (Wikipedia + BooksCorpus)

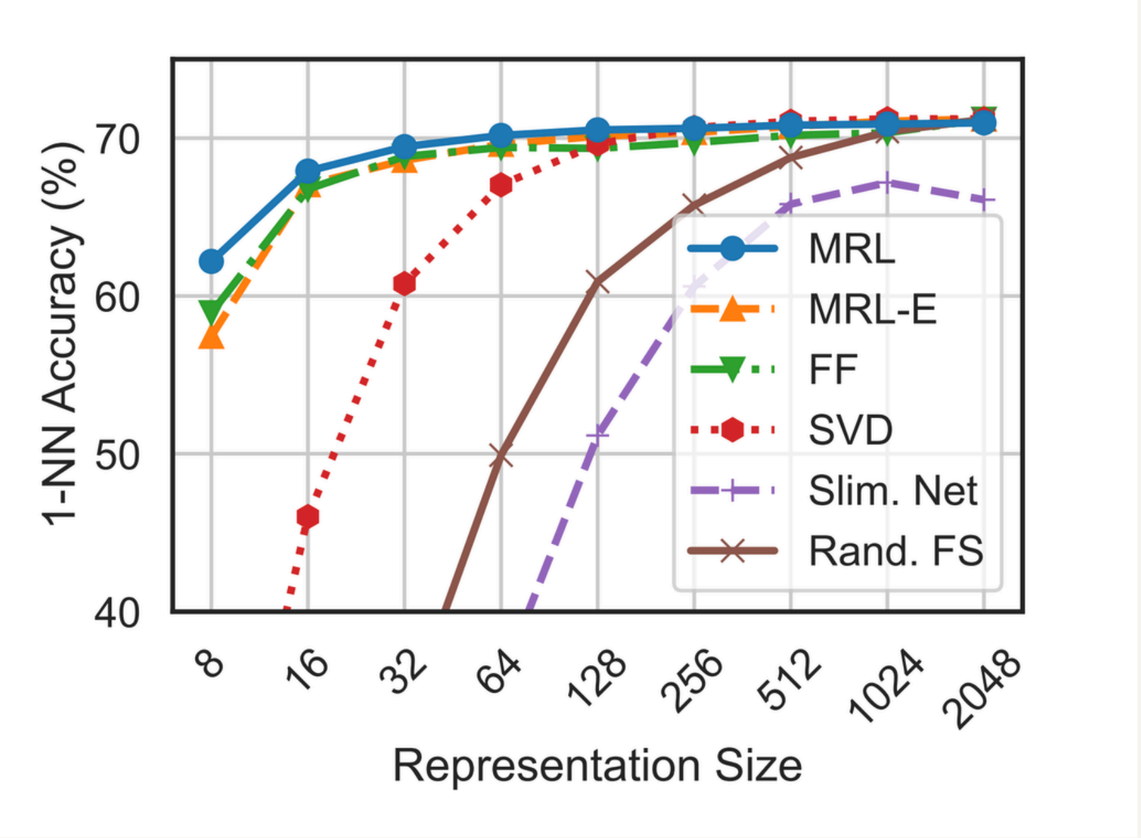
- Within 0.5% of FF at all sizes.
- Weight tying = MRL-E automatically.



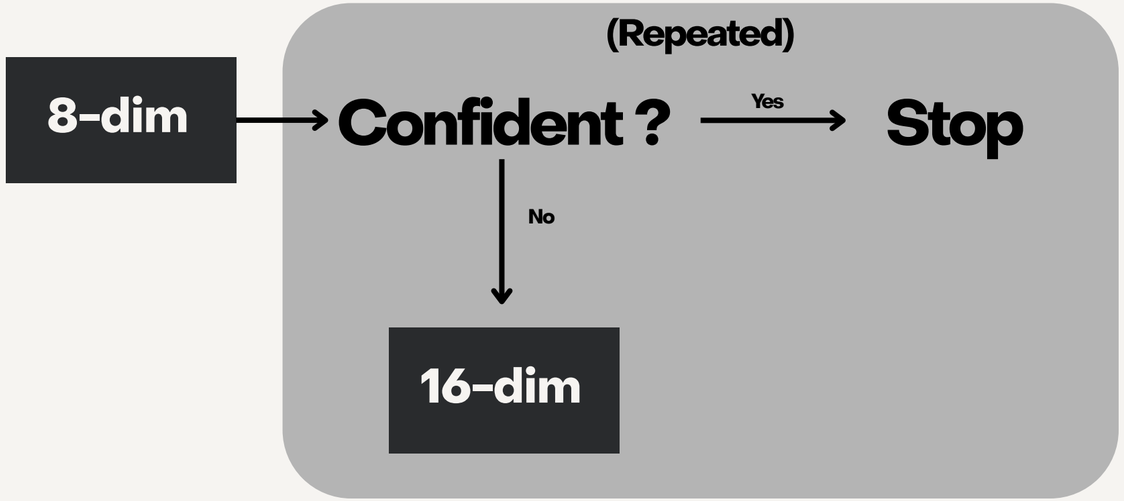
Representation Quality (1-NN)

- 1-NN requires no classifier training, making it a direct measure of raw embedding quality.

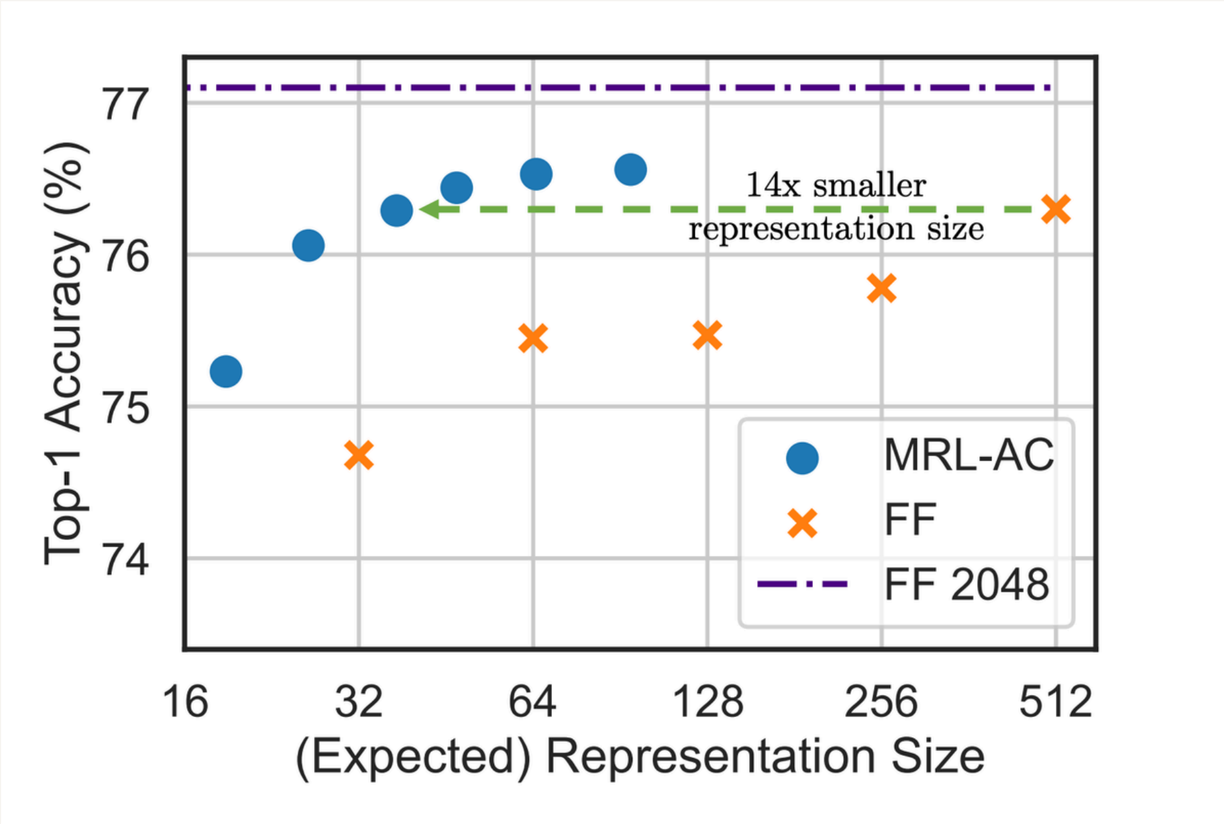
- MRL outperforms fixed-feature baselines by up to 2% at low dimensions — even beating models explicitly trained at those sizes.



Adaptive Classification

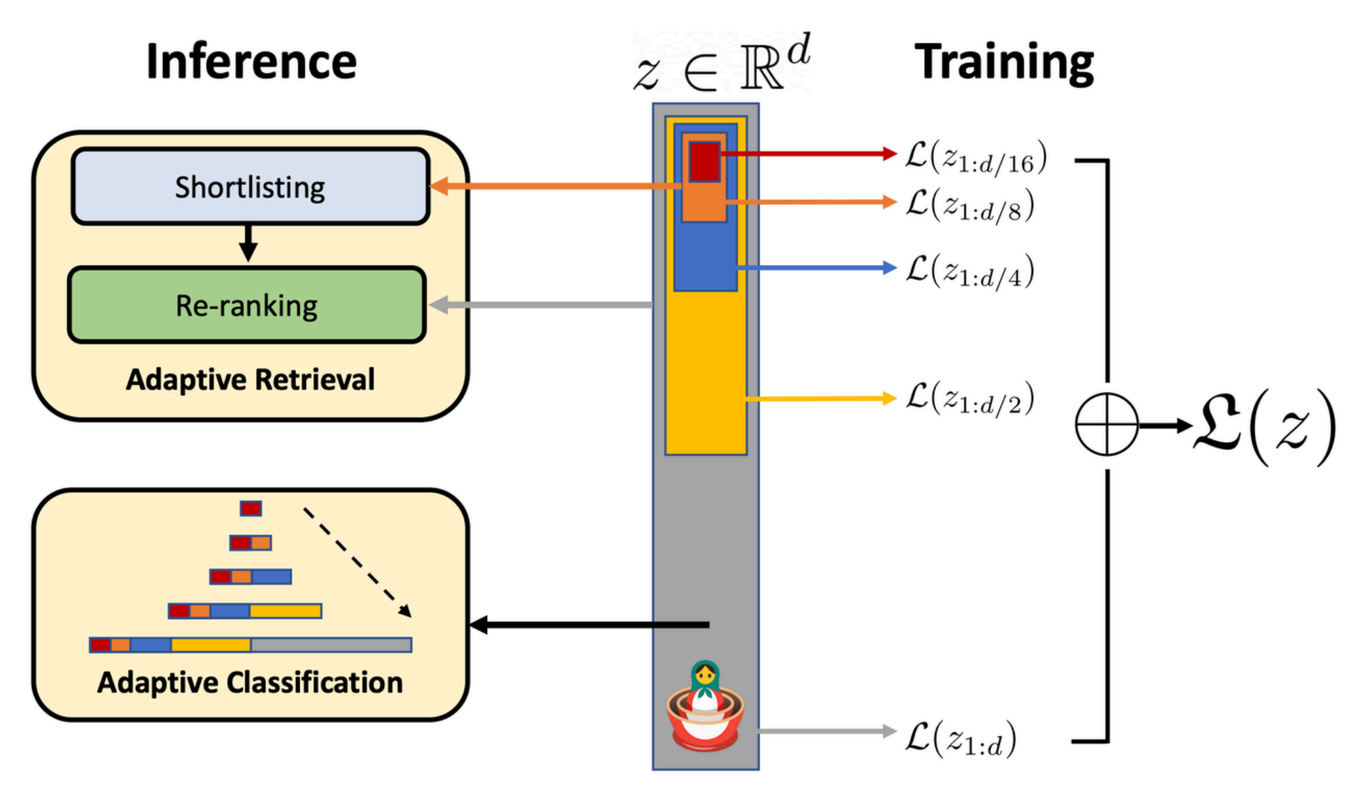


- Result: average dim used = ~37 for 76.3% accuracy.
- Same accuracy as 512-dim FF model but 14× smaller.

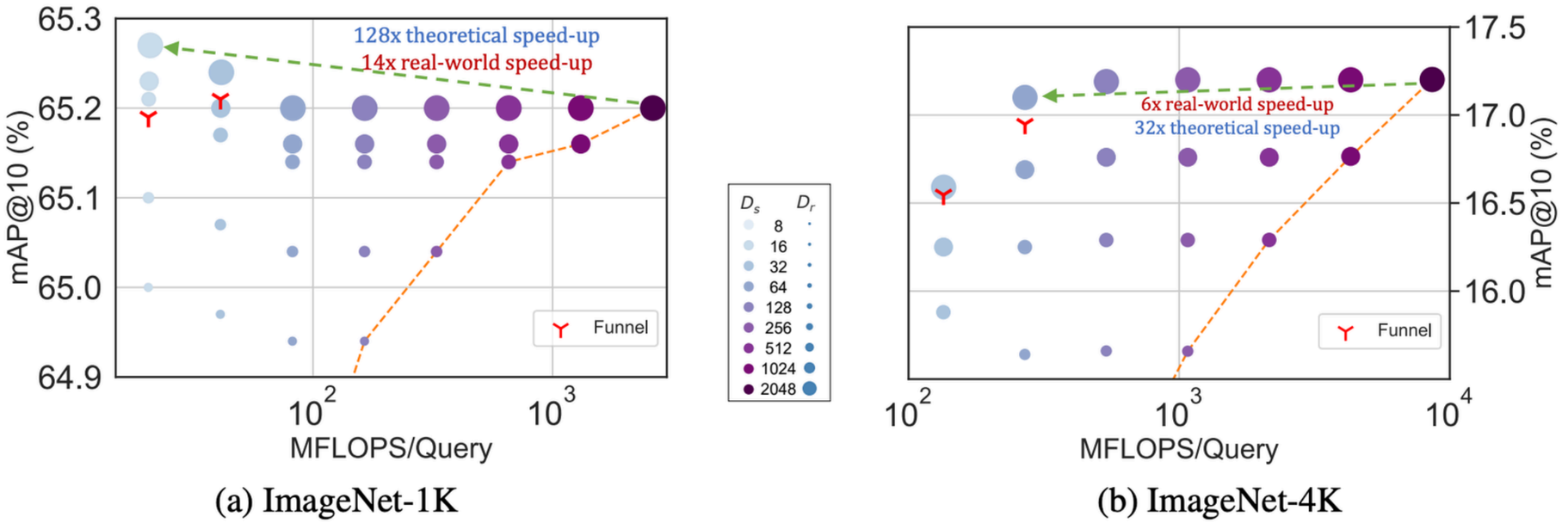


Retrieval Pipeline

- Step 1: search full database using 16-dim (cheap).
- Step 2: rerank top 200 using 2048-dim (accurate).
- Expensive part uses tiny vector, accurate part uses small shortlist.



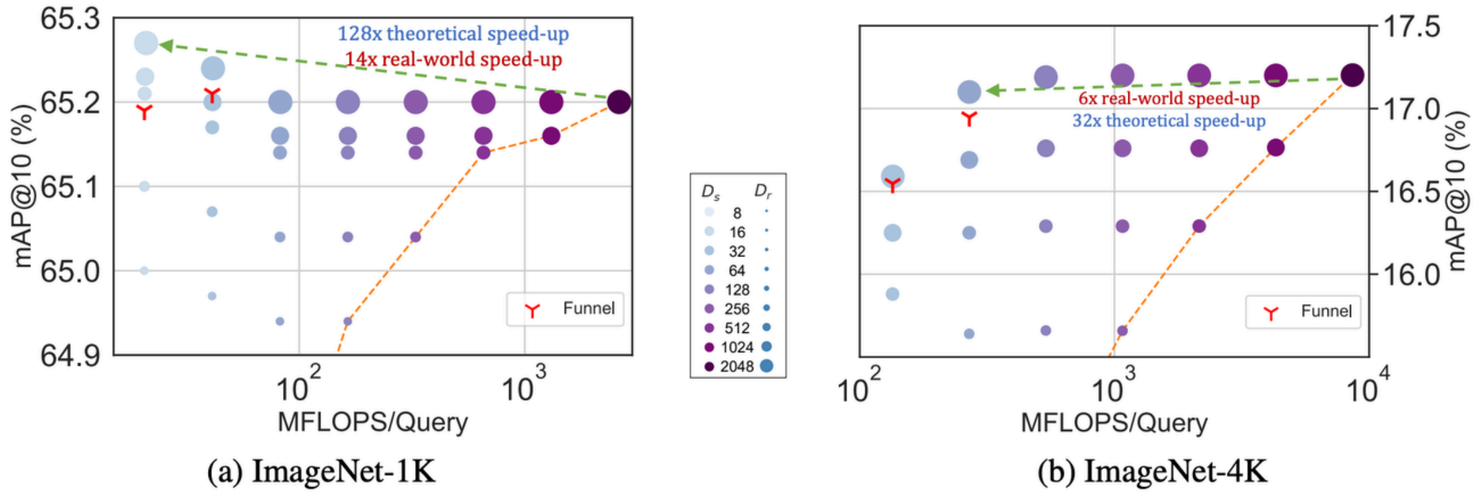
Adaptive Retrieval



- All D_s/D_r combinations fall above the single-shot Pareto line.
- ImageNet-1K: 128× theoretical, 14× real-world speedup.
- ImageNet-4K: 32× theoretical, 6× real-world speedup.

Funnel Retrieval

Funnel retrieval removes the need to manually choose Ds and Dr by using a fixed cascade.



200 candidates → 16-dim
100 candidates → 32-dim
50 candidates → 64-dim
25 candidates → 128-dim
10 candidates → 256-dim → 2048-dim

- Matches full 2048-dim accuracy at 128× lower compute cost on ImageNet-1K.
- Achieves equivalent accuracy at 64× lower cost on ImageNet-4K.

Limitations

- Loss weights (c_m=1) not optimized — tuning helps low dims.
- Dimension choices are heuristic, not learned.
- No learnable search structure on top of MRL yet.
- Full end-to-end joint optimization is future work.

Ablation Studies

FINETUNING

- Works on pretrained models.
- Within 1.5% of full MRL above 64-dim .

LOG vs UNIFORM SPACING

- 8-dim: Log=62.19% Uniform=58.44%.
- Uniform only catches up above 512-dim.

LOSS WEIGHTS

- Minimal cost at higher dims (<0.1%).
- Within 1.5% of full MRL above 64-dim.

MIN GRANULARITY

- Below 8-dim: unstable + unusable.
- 8-dim = validated practical minimum.

Conclusion

- One embedding, every size — train once, deploy flexibly.
- 14× smaller for classification, 14× faster for retrieval.
- Works across vision, language, and vision+language.
- Minimal training overhead, zero inference overhead.
- Complementary to all existing efficiency techniques.

References

[1] A. Kusupati, G. Bhatt, A. Rege, M. Wallingford, A. Sinha, V. Ramanujan, W. Howard-Snyder, K. Chen, S. Kakade, P. Jain, and A. Farhadi, "Matryoshka Representation Learning," in Advances in Neural Information Processing Systems, vol. 35, 2022.

[2] S. Agrawal, "The Russian Doll Revolution: How Matryoshka Learning is Transforming AI Efficiency," Medium, May 29, 2025. [Online]. Available: https://medium.com/@agrawalshrish321/the-russian-doll-revolution-how-matryoshka-learning-is-transforming-ai-efficiency-e04e0f43ec3f [Accessed: Apr. 22, 2026].

[3] B. Zhang, L. Chen, T. Liu, and B. Zheng, "SMEC: Rethinking Matryoshka Representation Learning for Retrieval Embedding Compression," in Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, Nov. 2025, pp. 26209–26222.

QUESTIONS?

EXTRA SLIDES

Robustness

DATASETS: V2 · R · A · Sketch

- MRL matches FF at every dimension
- ImageNet-A: +0.6% over FF (20% relative improvement)
- Out-of-domain retrieval: +3% mAP@10 over FF

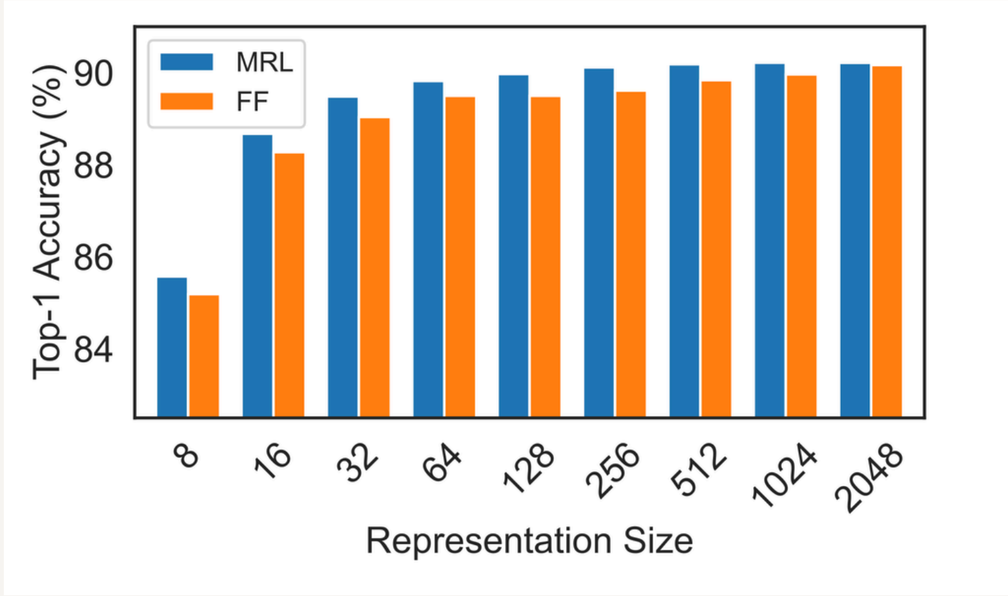
Dataset	MRL	FF
V2	64.93%	64.69%
A	3.59%	2.93%
R	35.07%	37.10%
Sketch	23.70%	24.05%

Few-Shot & Long-Tail

- Few-shot: MRL matches FF at all shot numbers and all sizes
- At 1-shot, 32-dim = 2048-dim accuracy — small vectors enough with few examples
- Long-tail (FLUID benchmark): +2% on novel tail classes
- Higher dims needed only when training data is scarce

Superclass Accuracy

- ~85% superclass accuracy even at 8-dim
- Fine-grained drops sharply below 64-dim
- "Garment": +11% from 8→16 dims alone
- Small dims capture semantics, not noise



Disagreement Across Dimensions

- Some images classify better at 8-dim than 2048-dim
- Perfect routing across dims = +4.6% accuracy gain
- Low dims fail on: cluttered scenes, same-superclass confusion (rock python vs boa)
- MRL doubles as a diagnostic tool for classification difficulty

Additional Insights

- Superclass accuracy stays ~85% even at 8-dim
- Fine-grained accuracy drops fast below 64-dim
- Small dims capture coarse semantics, not noise
- "Garment" superclass: +11% accuracy from 8→16 dims alone