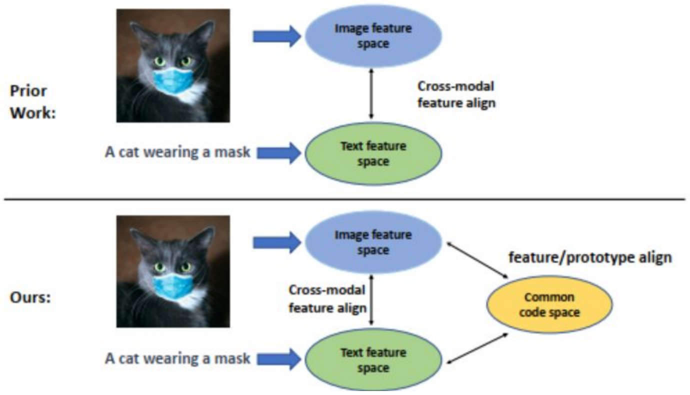
Multi-modal alignment using representation codebook

Duan, J., Chen, L., Tran, S., Yang, J., Xu, Y., Zeng, B., & Chilimbi, T.

Presented by Muntasir Wahed

Motivation

- Aligning feature representations of multi-modal models
- Bridging early fusion models and late fusion models
- Improve intra-modality alignment



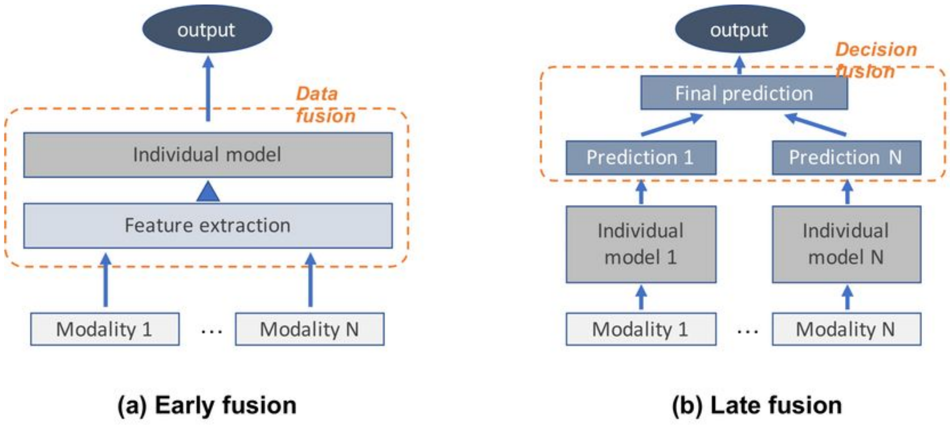


Image credit: Kim, Gyeongho, et al. "A multimodal deep learning-based fault detection model for a plastic injection molding process." IEEE Access 9 (2021): 132455-132467.

Contrastive Learning

- **Contrasts** every sample with all samples in the **minibatch**
- **Positive**: Different **views** of the same image
- **Negative**: All other samples in the minibatch

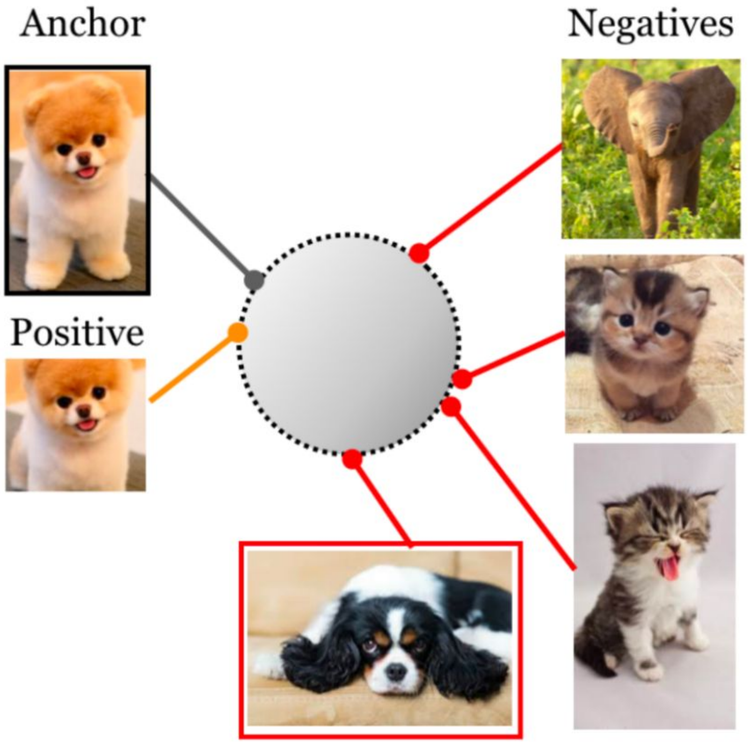


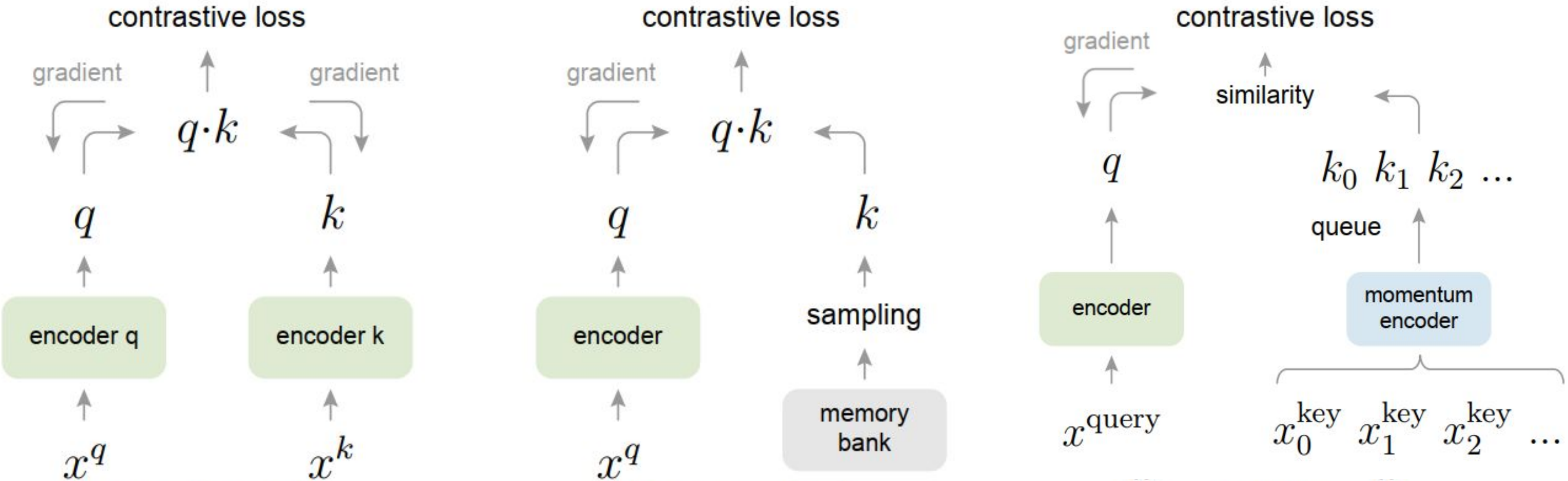
Image credit: Khosla, Prannay, et al. "Supervised contrastive learning." Advances in neural information processing systems 33 (2020): 18661-18673.

Momentum Contrast (MoCo) - Motivation

- Contrastive learning requires a large amount of negative samples
  - Large batch size - constrained by GPU memory
  - Memory bank - stale representations
- Maintain a queue of embeddings instead, evolving over time

$$\theta_k \leftarrow m\theta_k + (1 - m)\theta_q$$

Momentum Contrast (MoCo)



(a) end-to-end (b) memory bank (c) MoCo

Image credit: He, Kaiming, et al. "Momentum contrast for unsupervised visual representation learning." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.

Momentum Contrast (MoCo)

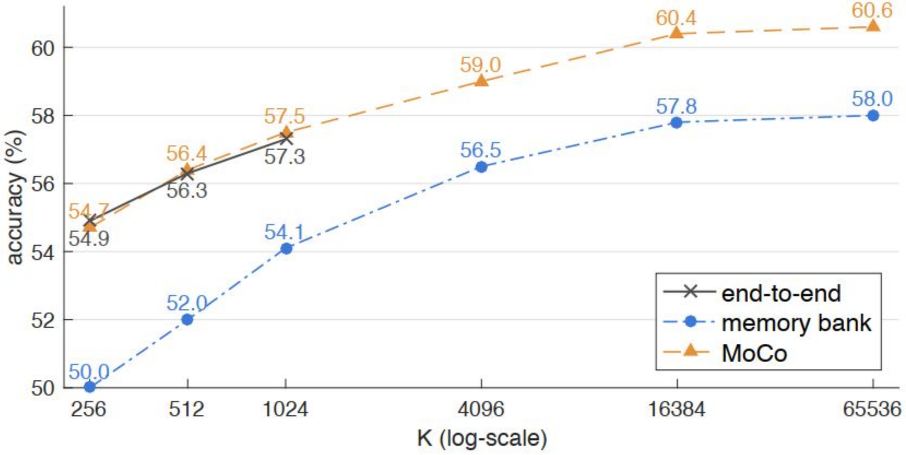
- Different view of the same image as query and key for the positive logit
- Back propagation only happens for the query
- Negative logits extracted from the queue

Algorithm 1 Pseudocode of MoCo in a PyTorch-like style.

```
# f_q, f_k: encoder networks for query and key
# queue: dictionary as a queue of K keys (CxK)
# m: momentum
# t: temperature

f_k.params = f_q.params # initialize
for x in loader: # load a minibatch x with N samples
   x_q = aug(x) # a randomly augmented version
   x_k = aug(x) # another randomly augmented version

   q = f_q.forward(x_q) # queries: NxC
   k = f_k.forward(x_k) # keys: NxC
   k = k.detach() # no gradient to keys

   # positive logits: Nx1
   l_pos = bmm(q.view(N,1,C), k.view(N,C,1))

   # negative logits: NxK
   l_neg = mm(q.view(N,C), queue.view(C,K))

   # logits: Nx(1+K)
   logits = cat([l_pos, l_neg], dim=1)

   # contrastive loss, Eqn.(1)
   labels = zeros(N) # positives are the 0-th
   loss = CrossEntropyLoss(logits/t, labels)

   # SGD update: query network
   loss.backward()
   update(f_q.params)

   # momentum update: key network
   f_k.params = m*f_k.params+(1-m)*f_q.params

   # update dictionary
   enqueue(queue, k) # enqueue the current minibatch
   dequeue(queue) # dequeue the earliest minibatch
```

bmm: batch matrix multiplication; mm: matrix multiplication; cat: concatenation.

Scalability



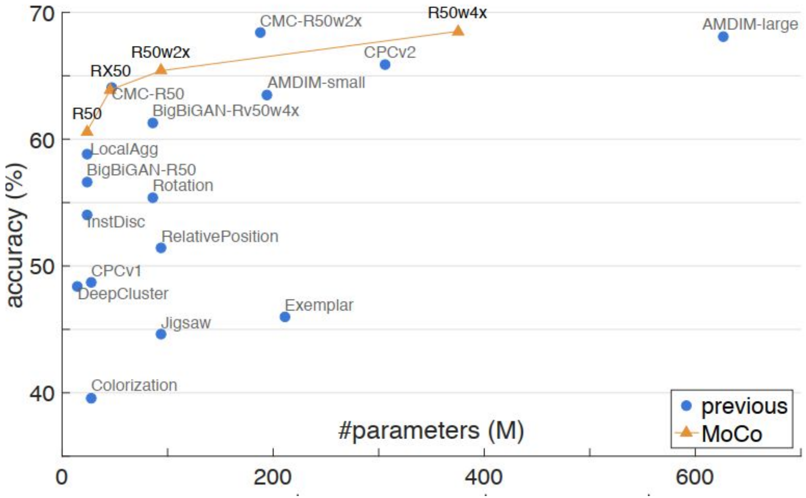


Image credit: He, Kaiming, et al. "Momentum contrast for unsupervised visual representation learning." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020.

Align Before Fuse (ALBEF) - Motivation

- Address the limitations of late fusion models
  - The image and text embeddings in their own spaces
  - Use of annotation-expensive and compute-expensive object detector
  - The datasets are inherently noisy, and existing pre-training objectives such as MLM may overfit

Image Text Contrastive Learning (ITC) Loss

- g_v and g_w are linear transformations that map the [CLS] embeddings to normalized lower-dimensional (256-d) representations
- two queues to store the most recent M image-text representations, the normalized features denoted by $g'_v(v'_{cls})$ and $g'_w(w'_{cls})$

$$s(I,T) = g_v(\boldsymbol{v}_{\text{cls}})^\top g'_w(\boldsymbol{w}'_{\text{cls}}) \qquad s(T,I) = g_w(\boldsymbol{w}_{\text{cls}})^\top g'_v(\boldsymbol{v}'_{\text{cls}}).$$

$$p_m^{\text{i2t}}(I) = \frac{\exp(s(I,T_m)/\tau)}{\sum_{m=1}^M \exp(s(I,T_m)/\tau)}, \qquad p_m^{\text{t2i}}(T) = \frac{\exp(s(T,I_m)/\tau)}{\sum_{m=1}^M \exp(s(T,I_m)/\tau)}$$

$$\mathcal{L}_{\text{itc}} = \frac{1}{2}\mathbb{E}_{(I,T)\sim D}\big[\mathrm{H}(\boldsymbol{y}^{\text{i2t}}(I), \boldsymbol{p}^{\text{i2t}}(I)) + \mathrm{H}(\boldsymbol{y}^{\text{t2i}}(T), \boldsymbol{p}^{\text{t2i}}(T))\big]$$

Masked Language Modeling (MLM) Loss

- Predict ground-truth labels of masked text tokens.

$$\mathcal{L}_{\mathrm{mlm}} = \mathbb{E}_{(I,\hat{T})\sim D}\mathrm{H}(\boldsymbol{y}^{\mathrm{msk}}, \boldsymbol{p}^{\mathrm{msk}}(I, \hat{T}))$$

Image Text Matching (ITM) Loss

- [CLS] token used as the joint representation of the image-text pair.
- Use a fully connected layer to predict the matching probability.

$$\mathcal{L}_{\mathrm{itm}} = \mathbb{E}_{(I,T)\sim D}\mathrm{H}(\boldsymbol{y}^{\mathrm{itm}}, \boldsymbol{p}^{\mathrm{itm}}(I,T))$$

ALBEF Pre-training

Training Objective

$$\mathcal{L} = \mathcal{L}_{\text{itc}} + \mathcal{L}_{\text{mlm}} + \mathcal{L}_{\text{itm}}$$

Momentum Distillation

- ITC and MLM penalize all negative predictions regardless of their correctness
- Modify the loss functions to learn from pseudo-targets generated by the momentum model instead
- a weighted combination of the original loss and the KL-divergence between the model’s prediction and the pseudo-targets

Align Before Fuse (ALBEF) - Benefits

- Aligns the image and text embeddings to improve cross-modal learning
- Improves the unimodal encoders to better understand the semantic meaning of images and texts
- A common low-dimensional space to embed images and texts
    - facilitates extraction of informative samples through our contrastive hard negative mining
- Model not penalized for producing reasonable outputs different from the web annotation, resulting in more stable learning

Codebook Learning with **Dis**tillation (CODIS)

- Inspired by ALBEF
  - Consider both intra and cross modal alignment in L_{ica}
- Multimodal codebook learning
  - Learnable codebook for both modalities
  - Predict codebook assignment using either text or image
- Teacher-student distillation
  - Guides the codebook learning
  - Improves unimodal and cross-modal alignment

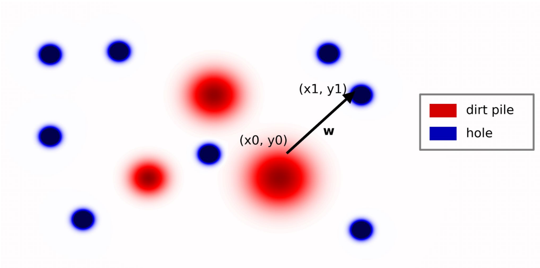
Relation to Prior Work

- A hybrid between the late-fusion and early-fusion works
    - ALBEF [1] is also doing something similar
- Codebook used by BEiT [2] and SOHO [3] to quantize the visual space
    - Contrary to them, this work quantized the join output space
- The loss function inspired by SwAV [4]
    - SwAV contrasts one view of the image with the assigned cluster of the same image
    - This paper contrasts across modalities

[1] Junnan Li, Ramprasaath R Selvaraju, Akhilesh Deepak Gotmare, Shafiq Joty, Caiming Xiong, and Steven Hoi. Align before fuse: Vision and language representation learning with momentum distillation. arXiv preprint arXiv:2107.07651, 2021.
[2] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. arXiv preprint arXiv:2106.08254, 2021.
[3] Zhicheng Huang, Zhaoyang Zeng, Yupan Huang, Bei Liu, Dongmei Fu, and Jianlong Fu. Seeing out of the box: End-toend pre-training for vision-language representation learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 12976–12985, 2021.
[4] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised learning of visual features by contrasting cluster assignments. arXiv preprint arXiv:2006.09882, 2020.

Optimal Transport

- Map one distribution to another distribution
- n! combinations available for two discrete distributions consisting of n items each
- Find the most optimal (with least cost) solution to this matching problem



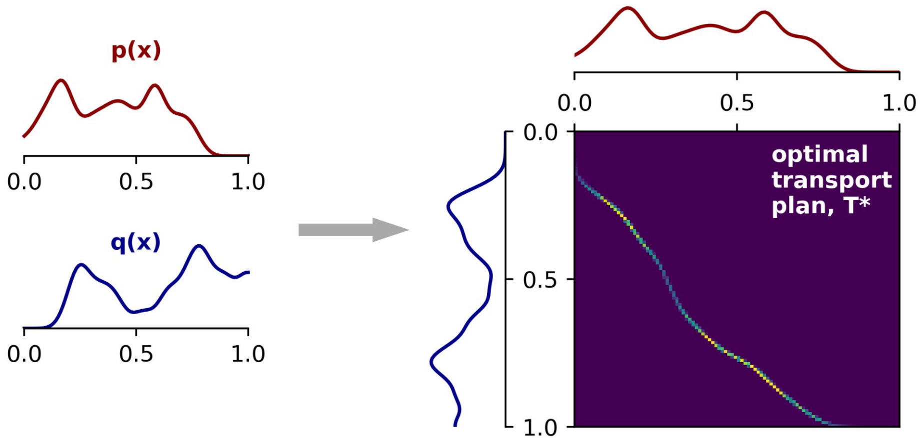


Image credit: http://alexhwilliams.info/itsneuronalblog/2020/10/09/optimal-transport/

Optimal Transport (cont.)

- Tries to minimize the optimal transport distance between prototypes and features
- Maps each feature with a prototype
- Sparse solution, with at most (2r − 1) (r = max(N, K) non-zero elements

Algorithm 2 IPOT Algorithm.

Input: distance/similarity matrix $\mathbf{Z}$, $\mathbf{C}$, ϵ, probability vectors $\boldsymbol{\mu}$, $\boldsymbol{\nu}$
$\boldsymbol{\sigma} = \frac{1}{n}\mathbf{1_n}$, $\mathbf{T}^{(1)} = \mathbf{1}\mathbf{1}^\top$
$D_{ij} = d(\boldsymbol{z}_i, \boldsymbol{c}_j)$,$\mathbf{A}_{ij} = \mathrm{e}^{-\frac{\mathbf{D}_{ij}}{\epsilon}}$
for $t = 1, 2, 3 \ldots$ **do**
  $\mathbf{Q} = \mathbf{A} \odot \mathbf{T}^{(t)}$ // $\odot$ is Hadamard product
  for $k = 1, 2, 3, \ldots K$ **do**
    $\boldsymbol{\delta} = \frac{\boldsymbol{\mu}}{n\mathbf{Q}\boldsymbol{\sigma}}$, $\boldsymbol{\sigma} = \frac{\boldsymbol{\nu}}{n\mathbf{Q}^\top\boldsymbol{\delta}}$
  end for
  $\mathbf{T}^{(t+1)} = \mathrm{diag}(\boldsymbol{\delta})\mathbf{Q}\mathrm{diag}(\boldsymbol{\sigma})$
end for
Return $\mathbf{T}$

\# samples

\# codebooks

cost/distance

$$\mathcal{L}_{\mathrm{ot}} = \min_{\mathbf{T}\in\Pi(\mathbf{u},\mathbf{v})} \sum_{i=1}^{\boxed{N}} \sum_{j=1}^{\boxed{K}} \boxed{\mathbf{T}_{ij}} \cdot \boxed{d(\boldsymbol{z}_i^m, \boldsymbol{c}_j)} = \min_{\mathbf{T}\in\Pi(\mathbf{u},\mathbf{v})} \langle \mathbf{T}, \mathbf{D} \rangle$$

optimal transport plan

Multimodal Codebook Learning

- Codebook (prototypes)
  - Encodes image and text into a joining embedding space
- Optimal Transport, T, used as ground-truth signals

$$\mathcal{L}_{\text{t2p}}(\mathbf{Z}_t, \mathbf{C}, \mathbf{T}_{i2p}) = H(\mathbf{P}_{t2p}, \mathbf{T}_{i2p}),$$

$$\mathcal{L}_{\text{i2p}}(\mathbf{Z}_v, \mathbf{C}, \mathbf{T}_{t2p}) = H(\mathbf{P}_{i2p}, \mathbf{T}_{t2p}), \tag{2}$$

$$\mathbf{P}_{t2p} = \textbf{SoftMax}(\mathbf{Z}_t\mathbf{C}/\gamma), \mathbf{P}_{i2p} = \textbf{SoftMax}(\mathbf{Z}_v\mathbf{C}/\gamma)$$

Codebook Loss

- Both the text-to-prototype (L_{t2p}) loss and image-to-prototype (L_{i2p}) loss chain features from both modalities
- When calculating the transport plan, use the teacher encoders
- Losses back propagated to both the codebook and the student encoders

$$\begin{aligned}\mathcal{L}_{\text{code}} &= \mathcal{L}_{\text{ot}}(\mathbf{Z}_v^m, \mathbf{C}) + \mathcal{L}_{\text{ot}}(\mathbf{Z}_t^m, \mathbf{C}) \\ &+ \mathcal{L}_{\text{t2p}}(\mathbf{Z}_t, \mathbf{C}, \mathbf{T}_{t2p}) + \mathcal{L}_{\text{i2p}}(\mathbf{Z}_v, \mathbf{C}, \mathbf{T}_{i2p})\end{aligned}$$

Teacher-student Distillation Learning

- Store features from teacher encoders z_v^m and z_t^m in memory queues Q_v and Q_t.
- Pseudo negatives are sampled from the queues.
- Also use the teacher encoders to provide soft distillation targets, y_{i2t}, y_{t2i}, y_{t2t}, y_{i2i}.
- Teacher encoders are updated using momentum.

$$f_t = \alpha f_t + (1-\alpha) f_s, g_t = \alpha g_t + (1-\alpha) g_s$$

$$\boldsymbol{p}_{t2i}(T) = \exp \frac{\boldsymbol{z}_t \boldsymbol{z}_v^{m\top}}{\gamma} / \sum_{\boldsymbol{z}_v^{m'} \in \mathbf{Q}_v} \exp \frac{\boldsymbol{z}_t \boldsymbol{z}_v^{m'\top}}{\gamma}$$

$$\boldsymbol{p}_{i2t}(I) = \exp \frac{\boldsymbol{z}_v \boldsymbol{z}_t^{m\top}}{\gamma} / \sum_{\boldsymbol{z}_t^{m'} \in \mathbf{Q}_t} \exp \frac{\boldsymbol{z}_v \boldsymbol{z}_t^{m'\top}}{\gamma}$$

$$\boldsymbol{p}_{i2i}(I) = \exp \frac{\boldsymbol{z}_v \boldsymbol{z}_v^{m\top}}{\gamma} / \sum_{\boldsymbol{z}_v^{m'} \in \mathbf{Q}_v} \exp \frac{\boldsymbol{z}_v \boldsymbol{z}_v^{m'\top}}{\gamma}$$

$$\boldsymbol{p}_{t2t}(T) = \exp \frac{\boldsymbol{z}_t \boldsymbol{z}_t^{m\top}}{\gamma} / \sum_{\boldsymbol{z}_t^{m'} \in \mathbf{Q}_t} \exp \frac{\boldsymbol{z}_t \boldsymbol{z}_t^{m'\top}}{\gamma}$$

Training Objective

- Simultaneously optimize the codebook and the student encoders
- L_{MLM} conditioned on both surrounding text tokens and image representations
- For L_{itm}, sample one negative text/image using contrastive similarity distribution.

$$\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{mlm}} + \mathcal{L}_{\text{itm}} + \mathcal{L}_{\text{ica}} + \mathcal{L}_{\text{code}}$$

Experimental Setup (Downstream Tasks)

- Image-Text Retrieval
  - Zero-shot
  - After-finetuning
- Visual Question Answering (VQA)
- Visual Reasoning ($NLVR^2$)
- Visual Entailment (SNLI-VE)

Experimental Results (Zero-Shot)

Method	MSCOCO (5K)						Flickr30K (1K)					
	Text Retrieval			Image Retrieval			Text Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
ImageBERT [36]	44.0	71.2	80.4	32.3	59.0	70.2	70.7	90.2	94.0	54.3	79.6	87.5
Unicoder-VL [24]	-	-	-	-	-	-	64.3	85.8	92.3	48.4	76.0	85.2
UNITER [8]	-	-	-	-	-	-	80.7	95.7	98.0	66.2	88.4	92.9
ViLT [22]	56.5	82.6	89.6	40.4	70.0	81.1	73.2	93.6	96.5	55.0	82.5	89.8
CLIP [37]	58.4	81.5	88.1	37.8	62.4	72.2	88.0	98.7	99.4	68.7	90.6	95.2
ALIGN [21]	58.6	83.0	89.7	45.6	69.8	78.6	88.6	98.7	**99.7**	75.7	93.8	**96.8**
ALBEF 4M [25]	68.6	89.5	94.7	50.1	76.4	84.5	90.5	98.8	**99.7**	76.8	93.7	96.7
Ours	**71.5**	**91.1**	**95.5**	**53.9**	**79.5**	**87.1**	**91.7**	**99.3**	99.8	**79.7**	**94.8**	97.3

Experimental Results (Finetuning)

Method	MSCOCO (5K)						Flickr30K (1K)					
	Text Retrieval			Image Retrieval			Text Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
ImageBERT [36]	66.4	89.8	94.4	50.5	78.7	87.1	87.0	97.6	99.2	73.1	92.6	96.0
UNITER [8]	65.7	88.6	93.8	52.9	79.9	88.0	87.3	98.0	99.2	75.6	94.1	96.8
VILLA [14]	-	-	-	-	-	-	87.9	97.5	98.8	76.3	94.2	96.8
OSCAR [28]	70.0	91.1	95.5	54.0	80.8	88.5	-	-	-	-	-	-
ViLT [22]	61.5	86.3	92.7	42.7	72.9	83.1	83.5	96.7	98.6	64.4	88.7	93.8
UNIMO [27]	-	-	-	-	-	-	89.7	98.4	99.1	74.6	93.4	96.0
SOHO [20]	66.4	88.2	93.8	50.6	78.0	86.7	86.5	98.1	99.3	72.5	92.7	96.1
ALBEF 4M [25]	73.1	91.4	96.0	56.8	81.5	89.2	94.3	**99.4**	99.8	82.8	96.7	**98.4**
Ours	**75.3**	**92.6**	**96.6**	**58.7**	**82.8**	**89.7**	**95.1**	**99.4**	**99.9**	**83.3**	96.1	97.8

Experimental Results (VQA, NVLR^2, SNLI-VE)

Method	VQA		NLVR^2		SNLI-VE	
	test-dev	test-std	dev	test-P	val	test
VisualBERT [26]	70.80	71.00	67.40	67.00	-	-
LXMERT [43]	72.42	72.54	74.90	74.50	-	-
12-in-1 [32]	73.15	-	-	78.87	-	76.95
UNITER [8]	72.70	72.91	77.18	77.85	78.59	78.28
ViLT [22]	70.94	-	75.24	76.21	-	-
OSCAR [28]	73.16	73.44	78.07	78.36	-	-
VILLA [14]	73.59	73.67	78.39	79.30	79.47	79.03
ALBEF 4M [25]	74.54	74.70	80.24	80.50	80.14	80.30
Ours	**74.86**	**74.97**	**80.50**	**80.84**	**80.47**	**80.40**

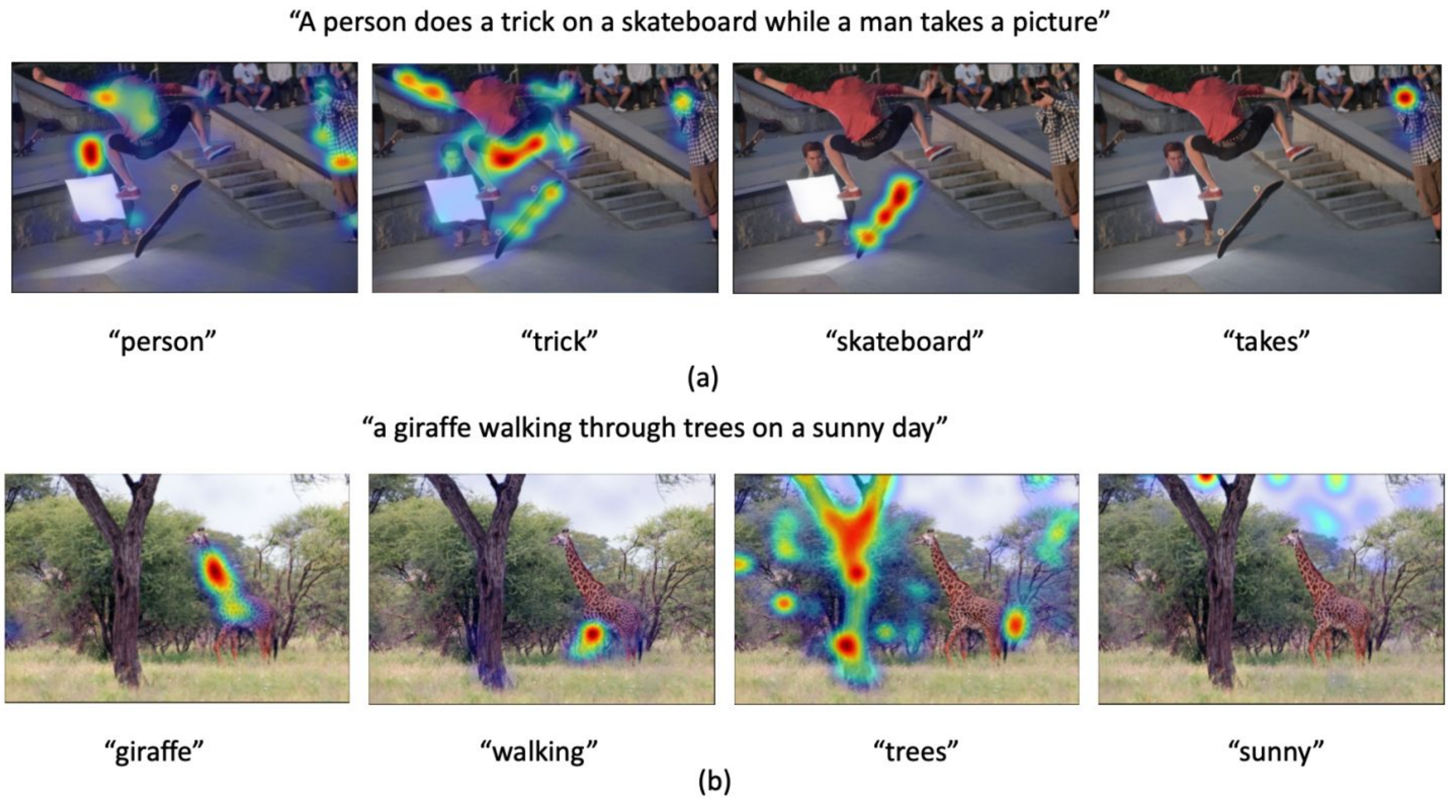
Ablation Studies

Objective functions	MSCOCO (5K)						Flickr30K (1K)					
	Text Retrieval			Image Retrieval			Text Retrieval			Text Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
a: MLM+ITM+ITC (cross align)	68.60	89.50	94.70	50.10	76.40	84.50	84.90	97.20	99.00	68.18	88.58	93.02
b: MLM+ITM+ITC (intra + cross)	69.86	89.48	94.42	50.52	77.02	85.17	85.80	96.80	98.10	69.70	89.60	93.48
a + codebook (teacher feature)	70.74	89.54	94.88	51.39	77.86	85.60	86.00	97.00	98.20	70.18	90.66	94.44
b + codebook (student feature)	71.12	89.62	94.78	51.40	77.42	85.53	86.30	96.90	98.30	70.34	90.00	93.84
b + codebook (teacher feature)	**71.10**	**90.60**	**95.10**	**52.10**	**78.00**	**85.90**	**86.70**	**97.30**	98.70	**71.40**	**90.82**	**94.62**

Ablation Studies

	TR@1	TR@5	TR@10	IR@1	IR@5	IR@10
ALBEF	55.70	81.92	88.78	41.08	69.01	78.86
0.5x codebook	58.66	83.9	90.64	43.74	72.10	81.58
2.0x codebook	59.02	84.46	91.06	43.62	71.69	81.12
3K codewords	58.96	84.28	90.98	44.66	72.31	81.68
500 codewords	55.52	81.68	89.28	41.53	68.75	78.43
Ours	59.38	84.04	91.20	44.71	72.63	81.69

Qualitative Analysis



Strengths

- Proposes intra-modal alignment to further improve cross-modal alignment
  - Ablation studies show that it improves the performance significantly
- The proposed teacher-student distillation framework works well
  - the slowly evolving teacher encoder helps the training process
- Strong results across multiple experiments against state-of-the-art baselines
- GRAD-CAM visualization is very interesting

Weaknesses

- Updating all the encoders simultaneously
  - Can lead to unpredictable oscillations
- Various issues with optimal transport
  - Why optimal transport instead of a simpler clustering algorithm?
  - Not clear if each codebook has only one image and vice versa
- Issues with notation.
  - Assumes too much about reader's prior knowledge.
  - Prior concepts used in the paper not explained properly
  - Missing notations for the algorithm for Optimal Transport
- Some minor errors in the tables

Future Works

- Instead of aligning the embeddings in a single layer, we can experiment with aligning them over multiple layers.
    - This might have the effect of aligning the embeddings at different semantic levels.
- Using the codebooks, we can sample hard negatives for the L_{itm} loss.

Discussion

- What is the reason for using optimal transport?
- Why do you think the intra-modal alignment is helping improve the results?