Relaxing Contrastiveness in Multimodal Representation Learning
SUPPLEMENTARY MATERIALS

Zudi Lin[1*] Erhan Bas[2*] Kunwar Yashraj Singh[3] Gurumurthy Swaminathan[3] Rahul Bhotika[4*]
[1]Amazon Alexa Science [2]Scale AI [3]AWS AI Labs [4]Optum Labs

In this document, we describe more implementation details (Sec. 1), introduce a generalized vector embedding (Sec. 2), and present additional quantitative results (Sec. 3).

1. Implementation Details

Data augmentation. At training time, we augment images with random horizontal flip $p = 0.5$, rotation ($[-20, 20]$ degrees), maximum fractional translation (0.1 of image width and height), random size scaling ($[0.75, 1.25]$) as spatial transformations with linear interpolation to sample images. We used color jittering (brightness$=0.4$, contrast$=0.4$), Gaussian Blur (kernel_size=5, sigma=$[0.1, 3.0]$) as pixel augmentations. For the MIMIC-CXR [3] experiments, we stack the gray-scale images along the channel dimension to make them RGB images and use the ImageNet [1] statistics to normalize the images. There is no augmentation at test time. We also do not add any augmentation for text.

Dataset split for transfer learning. In Sec. 4.3 we described the linear evaluation and finetuning results on the CheXpert [2] dataset for *multi-label* classification. Specifically, this dataset contains 223414 training images and 234 validation images. Since the official test images are not publicly released, we follow ConVIRT [7] and split the official training set into 218414/5000 as the new training and validation sets, and use the official validation set as the test set. The AUC scores are calculated for each of the eight classes and averaged as the final metric.

Text embedding selection. For the MIMIC-CXR [3] and CheXpert retrieval [7] experiments, we use the max-pooling result over output tokens as the representation of any given text input. This implementation is consistent with ConVIRT [7], and ConVIRT authors have reported that max-pooling achieves the overall best performance in comparison with the `[CLS]` token and other pooling strategies. For the Flickr30K [4] experiments, we follow the official CLIP [5] implementation, which uses a *causual* attention mask so that each token only has attention to the tokens before it. Therefore the output at the `[EOS]` token is used as the sentence embedding.

2. Generalized Vector Embedding

As described in the method part (Sec. 3), the InfoNCE loss forces negative pairs to be anti-correlated in the embedding space. Our proposed ReCo loss alleviates the optimization target and lets negative pairs be orthogonal or negatively correlated, preserving more flexibility and diversity in the embedding space and leading to significantly improved multimodal retrieval performance on two different datasets. However, both InfoNCE and our ReCo optimize positive pairs to be correlated, which geometrically means image and text embeddings are on the same line with the same direction (*i.e.*, cosine similarity is 1). This can post conflicts when having semantically different sentences describe the same image, a common situation in datasets like MIMIC-CXR [3]. Although the model does not arrive at the condition given a reasonably large dataset in practice, a better embedding space by design to incorporate more flexibility would be favorable. Therefore we propose a ***generalized*** embedding that instead of representing each sample as a single vector, we can represent it as a 2D parallelogram in the high-dimensional space using two vectors (called a 2-*blade*) so that if the parallelograms representing two samples reside on the same 2D plane, they are aligned. We follow the definition in exterior algebra and use the wedge product[1] of two vectors to define a 2-blade $\boldsymbol{u} = u_1 \wedge u_2$. Then the inner product between two 2-blades is defined as:

$$\langle \boldsymbol{u}, \boldsymbol{v} \rangle = \langle u_1 \wedge u_2, v_1 \wedge v_2 \rangle \triangleq \begin{vmatrix} \langle u_1, v_1 \rangle & \langle u_1, v_2 \rangle \\ \langle u_2, v_1 \rangle & \langle u_2, v_2 \rangle \end{vmatrix} \quad (1)$$

which is the *determinant* of the pairwise inner product matrix. Then the cosine similarity between two 2-blades is

$$\cos\theta = \frac{\langle u_1 \wedge u_2, v_1 \wedge v_2 \rangle}{\sqrt{\langle u_1 \wedge u_2, u_1 \wedge u_2 \rangle \langle v_1 \wedge v_2, v_1 \wedge v_2 \rangle}} \quad (2)$$

Similar to vectors, blades also have directions as $u \wedge v = -(v \wedge u)$ (*i.e.*, anticommutative). Thus the maximum and

*Work was done when affiliated with AWS AI Labs.

[1]It is also known as exterior product or outer product in some literature.

Table 1. Performance of baseline models. We report image-image and text-image retrieval results on the CheXpert 8×200 datasets [7]. InfoNCE-N, $N \in \{32, 64, 96\}$ denotes the model training with a mini-batch size of N using InfoNCE loss. Our InfoNCE-32 is the re-implementation of ConVIRT [7]. We use **InfoNCE-64** as the baseline model for other experiments in this work as it outperforms both ConVIRT and InfoNCE-32. The mean and standard deviation (σ) of our results are calculated from 4 runs. Please note that in Table 1 and Figure 5 we reported the *standard error* (SE), which is defined as SE$= \sigma/\sqrt{n}$ ($n = 4$ is the number of observations).

Method	Image-Image Retrieval			Text-Image Retrieval			Average
	Prec@5	Prec@10	Prec@50	Prec@5	Prec@10	Prec@50	
Random	12.5	12.5	12.5	12.5	12.5	12.5	12.5
ImageNet	14.8	14.4	15.0	–	–	–	–
Results reported in Zhang et al. [7]							
Caption-Transformer	29.8	28.0	23.0	–	–	–	–
Caption-LSTM	34.8	32.9	28.1	–	–	–	–
Contrastive-Binary	38.8	36.6	29.7	15.5	14.5	13.7	24.8
ConVIRT	45.0	42.9	35.7	60.0	57.5	48.8	48.3
Our baseline models with different batch sizes							
InfoNCE-32	41.0 (±1.4)	39.7 (±1.0)	33.6 (±1.1)	63.0 (±2.7)	57.9 (±2.1)	48.1 (±1.6)	47.2 (±1.1)
InfoNCE-64	43.3 (±0.9)	40.2 (±1.3)	35.0 (±0.4)	63.7 (±2.7)	59.2 (±2.4)	50.1 (±1.7)	**48.6** (±1.1)
InfoNCE-96	42.8 (±1.3)	40.9 (±1.1)	34.9 (±0.6)	62.0 (±3.5)	58.1 (±2.2)	49.3 (±1.1)	48.0 (±0.7)

Table 2. Impact of embedding dimensions in InfoNCE models. We report image and text retrieval results on the on CheXpert retrieval and show that the performance is stable across a range of different embedding dimensions (512 to 1536). Except for the output dimensions of the projection MLP, all other implementation details are identical to InfoNCE-64 (embedding dimension is 512) in Table 1.

Embedding Dim.	Image-Image Retrieval			Text-Image Retrieval			Average
	Prec@5	Prec@10	Prec@50	Prec@5	Prec@10	Prec@50	
256	42.8 (±2.7)	39.9 (±2.3)	34.3 (±1.4)	61.8 (±1.8)	59.5 (±2.8)	49.6 (±2.2)	48.0 (±0.9)
512	43.3 (±0.9)	40.2 (±1.3)	35.0 (±0.4)	63.7 (±2.7)	59.2 (±2.4)	50.1 (±1.7)	48.6 (±1.1)
768	43.9 (±2.0)	41.2 (±1.4)	35.5 (±0.9)	63.2 (±2.0)	59.5 (±2.6)	49.0 (±2.6)	48.7 (±1.1)
1024	43.2 (±1.3)	41.2 (±1.4)	35.3 (±0.3)	63.8 (±3.4)	58.9 (±2.2)	48.5 (±1.1)	48.5 (±1.3)
1280	43.0 (±1.6)	41.1 (±1.5)	35.0 (±0.3)	62.2 (±2.9)	59.5 (±2.4)	49.3 (±2.8)	48.4 (±1.1)
1536	44.3 (±1.6)	41.5 (±0.7)	34.9 (±0.6)	62.9 (±2.3)	59.1 (±2.9)	49.0 (±1.9)	48.6 (±1.2)
1792	41.3 (±1.1)	39.8 (±1.3)	34.9 (±1.1)	61.3 (±4.1)	59.1 (±2.9)	48.6 (±1.6)	47.5 (±0.8)
2048	43.7 (±2.1)	41.1 (±1.9)	35.0 (±1.0)	61.4 (±2.5)	57.5 (±1.5)	48.5 (±1.9)	47.8 (±0.2)

minimum of cosine are achieved when two blades reside on the same subspace but have identical or opposite directions. We can then extend the inner product and cosine to k-blades for a D dimensional embedding space, with $k < D$:

$$\langle u_1 \wedge \cdots \wedge u_k, v_1 \wedge \cdots \wedge v_k \rangle = |A| \tag{3}$$

where A is a $D \times D$ matrix and $A_{ij} = \langle u_i, v_j \rangle$ is the inner product of two vectors. Then the general cosine similarity between two k-blades is defined as

$$\cos\theta = \frac{|A|}{\sqrt{|B||C|}} \tag{4}$$

where $B_{ij} = \langle u_i, u_j \rangle$, $C_{ij} = \langle v_i, v_j \rangle$. Geometrically, the original cosine similarity is determined by the angle between two vectors, where the only degree of freedom is that changing the length (norm) of the vectors does not influence the cosine value. However, for 2-blades, besides changing the norm of vectors that forms a 2-blade, rotating a parallelogram on the 2D plane also does not change the cosine similarity as the cosine value is determined by the angle between two *planes* they reside on. For the general k-blade setting, the cosine value is determined by the angle between two k-dimensional subspace on which they reside, which provides even more freedom as transformations of the k-blade in the k-dimensional subspace do not change the cosine score. Our proposed representation generalizes the vector embedding to facilitate a similarity measure that preserves more diversity. The blade representation can be realized with minimal modification to existing architectures. We only increase the number of output units

Table 3. Impact of λ in our proposed ReCo (Eqn. 3) with an embedding dimension of 512 (vector representation). The optimal overall performance (averaged over image-image and text-image scores) is 51.5 with $\lambda = 0.6$, which improves the InfoNCE baseline by 2.9% in absolute precision. Besides, for a wide range of $\lambda \in [0.1, 0.8]$, our proposed ReCo outperforms the InfoNCE model. Please note that in Table 1 and Figure 5 we reported the *standard error* (SE), which is defined as SE$= \sigma/\sqrt{n}$ ($n = 4$ is the number of observations).

	Image-Image Retrieval			Text-Image Retrieval			Average
	Prec@5	Prec@10	Prec@50	Prec@5	Prec@10	Prec@50	
InfoNCE	43.3 (±0.9)	40.2 (±1.3)	35.0 (±0.4)	63.7 (±2.7)	59.2 (±2.4)	50.1 (±1.7)	48.6 (±1.1)
Weight λ in $\mathcal{L}_{\mathcal{MA}}$							
0.1	43.8 (±1.5)	41.8 (±1.4)	36.3 (±0.9)	65.6 (±2.2)	60.2 (±1.5)	51.9 (±1.8)	50.0 (±1.1)
0.2	44.9 (±1.8)	42.7 (±1.9)	37.0 (±0.7)	66.1 (±1.3)	62.0 (±2.6)	52.1 (±1.6)	50.8 (±0.3)
0.3	44.4 (±1.9)	42.8 (±1.2)	36.9 (±1.3)	66.0 (±3.4)	62.8 (±1.9)	53.4 (±2.2)	51.1 (±0.8)
0.4	44.6 (±2.5)	42.7 (±1.0)	37.0 (±1.5)	65.5 (±3.2)	62.5 (±2.1)	51.6 (±1.9)	50.6 (±0.8)
0.5	**45.9** (±0.9)	43.9 (±1.0)	**37.4** (±0.8)	64.1 (±2.8)	60.7 (±0.9)	52.8 (±1.8)	50.8 (±0.8)
0.6	45.6 (±1.4)	**44.1** (±1.8)	35.7 (±1.1)	**67.4** (±3.8)	62.8 (±1.9)	53.1 (±0.9)	**51.5** (±1.1)
0.7	45.3 (±1.4)	42.7 (±0.8)	35.3 (±0.8)	66.0 (±2.0)	61.5 (±0.6)	**53.9** (±1.5)	50.8 (±0.4)
0.8	44.5 (±1.4)	42.9 (±1.4)	34.4 (±0.9)	67.0 (±5.0)	**63.5** (±4.5)	52.8 (±2.0)	50.9 (±1.3)

Table 4. Impact of λ in $\mathcal{L}_{\mathcal{OC}}$ (Eqn. 6, the loss of orthogonal constraint) with an embedding dimension of 512 (vector representation). The optimal overall performance (averaged over image-image and text-image retrieval scores) is 51.3 with $\lambda = 0.15$, which improves the InfoNCE baseline by 2.7 in absolute precision, and is slightly lower than our ReCo loss (51.5). However, we have discussed in Sec. 4.4 that $\mathcal{L}_{\mathcal{OC}}$ decreases the performance on Flickr30K [4], while our ReCo can consistently improve the performance on both datasets.

	Image-Image Retrieval			Text-Image Retrieval			Average
	Prec@5	Prec@10	Prec@50	Prec@5	Prec@10	Prec@50	
InfoNCE	43.3 (±0.9)	40.2 (±1.3)	35.0 (±0.4)	63.7 (±2.7)	59.2 (±2.4)	50.1 (±1.7)	48.6 (±1.1)
Weight λ in $\mathcal{L}_{\mathcal{MA}}$							
0.01	39.3 (±1.4)	37.3 (±1.5)	33.6 (±0.8)	46.8 (±2.0)	42.7 (±1.7)	38.7 (±1.5)	39.7 (±0.4)
0.05	43.4 (±2.8)	42.2 (±1.2)	37.3 (±1.0)	62.5 (±1.7)	59.0 (±1.9)	50.3 (±2.4)	49.1 (±0.4)
0.10	45.0 (±1.5)	43.3 (±0.4)	37.1 (±1.0)	64.9 (±1.9)	60.9 (±1.9)	**52.1** (±1.4)	50.5 (±0.5)
0.15	**46.1** (±0.6)	**44.2** (±0.8)	**38.0** (±1.4)	**65.9** (±2.6)	61.8 (±2.2)	51.9 (±0.9)	**51.3** (±0.9)
0.20	44.2 (±1.5)	42.9 (±1.4)	37.6 (±0.8)	65.3 (±1.6)	**62.8** (±2.4)	**52.1** (±2.1)	50.8 (±1.0)
0.25	44.7 (±2.4)	43.6 (±1.6)	37.1 (±1.4)	61.4 (±2.5)	58.4 (±2.6)	50.8 (±2.7)	49.3 (±0.6)
0.30	44.9 (±1.5)	43.5 (±0.8)	36.6 (±0.6)	62.6 (±3.5)	60.1 (±1.9)	51.5 (±1.2)	49.9 (±0.8)
0.50	42.9 (±1.8)	42.0 (±2.8)	35.0 (±1.6)	65.3 (±3.3)	60.4 (±3.1)	51.7 (±1.9)	49.5 (±1.2)

and split them into multiple vectors to form a blade without changing layers before the projection layer. The new formulation of the embedding representation can seamlessly work with both InfoNCE and our proposed ReCo as both losses are directly calculated with the cosine similarity matrices.

3. Additional Experiments

In Figure 5 of the main manuscript, we have reported the comprehensive ablation studies of different hyper-parameters using InfoNCE and our proposed ReCo loss. We only report the average retrieval precision in Figure 5 due to space limitation. The detailed results of those experiments (*i.e.*, image-image and text-image retrieval precisions at different threshold) can be found in Table 1, 2 and 3. The results in those tables are generated with four independent runs with different initial random seeds, and the standard deviation (σ) is shown after the scores. Please note that in Table 1 and Figure 5 of the main manuscript, we reported the *standard error* (SE), which is defined as SE$= \sigma/\sqrt{n}$ ($n = 4$ is the number of observations). In Eqn. 6, we also describe a loss $\mathcal{L}_{\mathcal{OC}}$ that is inspired by the Barlow Twin [6] loss and force orthogonality for negative pairs. We show the detailed results in Table 4.

In this section, we show the additional results that mainly involve the generalized vector representation we proposed in Sec. 2. The experiment setting is identical to Sec. 4.1.

Blade representation with InfoNCE. In Table 5 we show the k-blade ($k \in 1, 2, 3, 4$) representations with two differ-

Table 5. Ablation study of the blade representations in Sec. 2. We tested k-blade representations with 256 and 512 embedding dimensions using the InfoNCE loss function. Please note that the 1-blade settings are identical to the corresponding vector representations.

Configuration	Image-Image Retrieval			Text-Image Retrieval			Average
	Prec@5	Prec@10	Prec@50	Prec@5	Prec@10	Prec@50	
1-blade 256	42.8 (±2.7)	39.9 (±2.3)	34.3 (±1.4)	61.8 (±1.8)	59.5 (±2.8)	49.6 (±2.2)	48.0 (±0.9)
2-blade 256	40.7 (±2.5)	41.0 (±1.5)	35.1 (±0.8)	64.3 (±2.2)	61.3 (±1.4)	50.2 (±1.7)	48.8 (±0.6)
3-blade 256	42.5 (±2.2)	40.2 (±1.5)	35.0 (±0.8)	64.3 (±4.4)	60.9 (±3.7)	50.9 (±0.7)	49.0 (±1.0)
4-blade 256	42.1 (±1.6)	40.3 (±0.5)	35.7 (±0.8)	60.6 (±1.0)	58.4 (±0.7)	48.7 (±1.8)	47.6 (±0.5)
1-blade 512	43.3 (±0.9)	40.2 (±1.3)	35.0 (±0.4)	63.7 (±2.7)	59.2 (±2.4)	50.1 (±1.7)	48.6 (±1.1)
2-blade 512	44.3 (±2.0)	41.6 (±1.2)	35.3 (±0.6)	64.8 (±3.0)	61.1 (±2.6)	49.9 (±1.0)	49.5 (±0.7)
3-blade 512	41.7 (±1.5)	41.5 (±1.1)	35.9 (±0.3)	66.1 (±5.0)	61.4 (±1.6)	51.2 (±0.6)	49.7 (±1.3)
4-blade 512	43.0 (±2.0)	41.5 (±1.6)	35.7 (±1.4)	63.5 (±2.9)	60.4 (±0.6)	49.1 (±1.7)	48.9 (±0.4)

Table 6. Impact of λ in our proposed ReCo loss (Eqn. 3) with **2-blade** representations (Eqn. 1) of dimension 512. The optimal overall performance (averaged over image-image and text-image scores) is 51.0 with $\lambda = 0.8$ and 1.6. Our ReCo with a wide range of λ outperforms the InfoNCE baseline. However, we also notice that the performance using ReCo with the blade representation is slightly lower then using our proposed ReCo along (Table 3).

	Image-Image Retrieval			Text-Image Retrieval			Average
	Prec@5	Prec@10	Prec@50	Prec@5	Prec@10	Prec@50	
2-blade 512	44.3 (±2.0)	41.6 (±1.2)	35.3 (±0.6)	64.8 (±3.0)	61.1 (±2.6)	49.9 (±1.0)	49.5 (±0.7)
Weight λ in $\mathcal{L}_{\mathcal{MA}}$							
0.2	44.3 (±1.5)	43.2 (±0.9)	36.7 (±1.0)	63.6 (±4.0)	60.8 (±3.2)	51.2 (±1.5)	50.0 (±0.7)
0.4	46.8 (±1.0)	45.5 (±1.0)	38.0 (±1.4)	63.5 (±2.5)	59.6 (±1.0)	49.7 (±2.4)	50.5 (±0.6)
0.6	46.6 (±1.5)	44.5 (±0.9)	36.9 (±0.6)	63.1 (±2.8)	61.3 (±3.6)	52.1 (±1.3)	50.8 (±1.1)
0.8	45.8 (±1.0)	45.3 (±0.2)	36.8 (±0.2)	64.0 (±2.8)	61.7 (±2.2)	52.2 (±2.2)	**51.0** (±1.1)
1.0	46.8 (±1.9)	44.4 (±0.5)	36.3 (±1.0)	63.5 (±3.2)	60.8 (±1.4)	50.3 (±1.2)	50.3 (±1.0)
1.2	46.9 (±1.5)	45.0 (±1.1)	34.9 (±1.1)	63.0 (±1.3)	60.3 (±2.9)	50.0 (±0.7)	50.0 (±0.9)
1.4	45.8 (±2.7)	44.6 (±1.4)	35.9 (±1.5)	63.8 (±1.2)	61.1 (±2.4)	49.8 (±1.9)	50.2 (±1.4)
1.6	45.7 (±1.4)	43.5 (±1.0)	35.3 (±0.9)	66.3 (±0.9)	63.3 (±2.0)	51.7 (±1.1)	**51.0** (±0.8)

ent embedding dimensions (256 and 512). Please note that when $k = 1$, the representation is exactly the corresponding vector representation. The results show that when using the 2-blade-512 setting, the average retrieval performance is 49.5%, which improves the corresponding vector-512 configuration by 0.9% and improves the vector 1024 configuration[2] by 1.0%, showing that the performance gain is *not* because of more output units but our proposed new representation. We also notice that 3-blade-512 can increase the score to 49.7%, which also outperforms the vector setting with an embedding dimension of 1536 (same output units). However, compared with the 2-blade-512 setting, the 3-blade-512 model improves the text-image retrieval results more but decreases the image-image retrieval scores. Therefore we suggest using the 2-blade-512 setting as it can consistently improve both text-image and image-image retrieval performance. We will also explore the explanation for such observations in our future work.

[2]Vector-1024 and 2-blade-512 settings have exactly the same model architecture, which is a fairer comparison.

Blade representation with ReCo. Since both the proposed blade representation and our ReCo loss can improve the performance, we also test the combined effect of both components. When combing ReCo with both 2-blade-512 (Table 6) and 3-blade-512 (Table 7) configurations we described above, we notice that ReCo can improve the performance by 1.5% with an a appropriate negative weight λ. However, we also notice that the results combining blade with ReCo are slightly lower than the results using ReCo only (51.5%). Our focus in the follow-up research will be to understand the interplay of the generalized vector representation and our proposed ReCo loss.

Table 7. Impact of λ in our proposed ReCo loss (Eqn. 3) with **3-blade** representations (Eqn. 1) of dimension 512. The optimal overall performance (averaged over image-image and text-image scores) is 51.2 with $\lambda = 1.2$. Our proposed ReCo with a wide range of λ outperforms the InfoNCE baseline. We notice that the results with 3-blade are slightly better than the 2-blade case. However, we also notice that the performance using ReCo with the blade representation is slightly lower than using our proposed ReCo along (Table 3).

	Image-Image Retrieval			Text-Image Retrieval			Average
	Prec@5	Prec@10	Prec@50	Prec@5	Prec@10	Prec@50	
3-blade 512	41.7 (±1.5)	41.5 (±1.1)	35.9 (±0.3)	66.1 (±5.0)	61.4 (±1.6)	51.2 (±0.6)	49.7 (±1.3)
Weight λ in $\mathcal{L}_{\mathcal{MA}}$							
0.2	44.4 (±1.9)	43.1 (±0.8)	36.9 (±0.9)	63.8 (±3.4)	61.1 (±2.6)	50.8 (±2.1)	50.0 (±0.8)
0.4	48.2 (±1.3)	43.9 (±1.3)	36.3 (±0.7)	65.1 (±5.2)	61.3 (±3.6)	51.9 (±1.4)	51.1 (±1.8)
0.6	45.2 (±2.2)	44.1 (±1.7)	35.8 (±0.6)	65.5 (±2.5)	62.1 (±2.7)	50.3 (±1.6)	50.5 (±0.7)
0.8	46.3 (±2.1)	44.0 (±1.8)	35.7 (±1.3)	64.9 (±2.2)	62.5 (±1.6)	51.5 (±1.0)	50.8 (±0.8)
1.0	47.1 (±2.3)	44.5 (±1.6)	36.3 (±0.7)	64.3 (±1.0)	61.5 (±2.1)	50.7 (±1.3)	50.7 (±0.8)
1.2	50.1 (±1.4)	45.1 (±0.4)	35.8 (±0.9)	64.8 (±0.9)	61.5 (±0.9)	50.2 (±1.6)	**51.2** (±0.4)
1.4	46.9 (±2.1)	44.4 (±1.0)	35.9 (±0.8)	65.1 (±2.2)	60.3 (±1.3)	49.4 (±1.7)	50.3 (±0.4)
1.6	48.0 (±1.6)	45.6 (±0.5)	36.2 (±0.5)	61.5 (±1.2)	59.6 (±1.7)	49.4 (±1.8)	50.0 (±0.5)

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[2] Jeremy Irvin, Pranav Rajpurkar, Michael Ko, Yifan Yu, Silviana Ciurea-Ilcus, Chris Chute, Henrik Marklund, Behzad Haghgoo, Robyn Ball, Katie Shpanskaya, et al. Chexpert: A large chest radiograph dataset with uncertainty labels and expert comparison. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 590–597, 2019.

[3] Alistair E. W. Johnson, T. Pollard, Seth J. Berkowitz, Nathaniel R. Greenbaum, M. Lungren, Chih ying Deng, R. Mark, and S. Horng. Mimic-cxr, a de-identified publicly available database of chest radiographs with free-text reports. *Scientific Data*, 6, 2019.

[4] Bryan A. Plummer, Liwei Wang, Christopher M. Cervantes, Juan C. Caicedo, J. Hockenmaier, and S. Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. *International Journal of Computer Vision*, 123:74–93, 2015.

[5] Alec Radford, Jong Wook Kim, Chris Hallacy, A. Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[6] J. Zbontar, L. Jing, Ishan Misra, Y. LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *ICML*, 2021.

[7] Yuhao Zhang, Hang Jiang, Yasuhide Miura, Christopher D. Manning, and Curtis P. Langlotz. Contrastive learning of medical visual representations from paired images and text, 2020.