Dynamically-Scaled Deep Canonical Correlation Analysis: Supplementary Material

Tomer Friedlander and Lior Wolf

Tel Aviv University, Tel Aviv, Israel
tomerf1@mail.tau.ac.il, wolf@cs.tau.ac.il

Table of Contents

The supplementary material is organized as follows:

1 Implementation Details

Following the literature of the CCA baselines, all neural networks used in this paper are fully-connected networks. For a given dataset, we train all neural network-based transformation functions with the same architecture initialized by the same seed (217). This seed was chosen randomly. Each fully connected layer is followed by a batch normalization layer [1] with no affine parameters and is then followed by the ReLU activation function [5]. The last layer is followed by a batch normalization layer, but not by an activation function. The number of layers and neurons of each mapping function for each dataset is given in Table I, where we follow [7] for MNIST and XRMB. The training is done for the same number of epochs (100, 500, 1k, 200 and 200 for MNIST, XRMB, Flickr8k, Flickr30k and IAPR TC-12) on batches (750 samples per batch for MNIST and XRMB, 1024 for Flickr8k, Flickr30k and IAPR TC-12) using the RMSprop optimizer [6] with a learning rate of $1e^{-3}$ and with a weight decay regularization of $1e^{-5}$.

After each training epoch, the loss of the current checkpoint of the model is computed. The learnable parameters that achieve the lowest loss on the validation set are chosen as the parameters of the model.

The hyperparameters of each model are tuned by evaluating, on the validation set, the total canonical correlation and recall measurements for the total canonical correlation and retrieval experiments, respectively.

Regarding our proposed DS-DCCA, we model the conventionally static layers of the mapping functions with neural networks, which are identical to the networks modelling the mapping functions in the network-based baselines, i.e.

Dataset	$\mathbf{x}_1/\mathbf{x}_2$	Layers of $\mathbf{f}_1/\mathbf{f}_2$
MNIST	L/R	800,800,50/800,800,50
XRMB	ACO/ART	1800,1800,112/1200,1200,112
Flickr$_*$, IAPR	IMG/TXT	1024,512,128/1024,512,128

Table I: The neural network architectures of the mapping functions used for each dataset. L=left, R=right, ACO=acoustic, ART=articulatory, IMG=image, TXT=text

according to Table I. In addition, we model the scaling networks with fully connected layers followed by a batch normalization and a ReLU activation function. The last layer, which outputs the scaling factors, is not followed by an activation function. The number of layers and neurons of the scaling networks are chosen out of $\{(128, c), (256, c), (256, 128, c)\}$, where c is equal to the total number of conventional static parameters to be scaled. The number of epochs defining the warm-up period, T, is set to 50.

The regularization coefficients of the linear CCA were tuned in the logarithmic-scaled range of $[10^{-8}, 10^{2}]$. For FKCCA and NKCCA, following prior works [2,4], a rank-6,000 approximation of the kernel matrices is used. The regularization coefficients of DCCA are chosen out of $\{1e^{-6}, 1e^{-4}, 1e^{-2}\}$. DCCAE was trained with the best performing regularization coefficient of DCCA and the reconstruction coefficient is chosen out of $\{1e^{-3}, 1e^{-2}, 1e^{-1}, 1\}$. For NCCA, the dimensions of the input views are reduced by PCA, as suggested by [4]. In particular, each view of XRMB was reduced to 20% of its original dimension, as done in the original NCCA study. For training NCCA on MNIST and Flickr8k, the dimensions of each view were reduced to the minimal number of PCA components, which captures the variance of at least 80% of the raw data. Regarding Soft-CCA, the stochastic regularization coefficient (λ_{SDL}) and the running-average coefficient (α) of each view are chosen out of $\{1, 2, 4, 5, 10, 15, 20\}$ and $\{0.85, 0.9, 0.95, 0.97\}$, respectively. Regarding ℓ_0-CCA, the gates' regularization term and the noise standard deviation were chosen out of $\{0.01, 0.1, 0.5\}$ and $\{0.1, 0.5, 0.75, 1\}$, respectively. The stochastic gates of ℓ_0-CCA were initialized by sampling a Gaussian distribution with a mean of 0.5 and a standard deviation of 0.01, as done in the official implementation of the stochastic gate.

In the retrieval experiment, both Ranking-CCA and our DS-Ranking CCA were trained for 500 (200) epochs on Flickr8k (Flickr30 and IAPR TC-23). The running-average coefficient of the CCA layer of the Ranking-CCA and the margin used in the pairwise ranking loss (m in Equation 7) were chosen out of $\{0.85, 0.9, 0.95, 0.97\}$ and $\{0.4, 0.5, 0.6, 0.7, 0.8\}$, respectively.

2 Multiple Runs With Different Initial Seeds

2.1 Total Canonical Correlation

In order to examine the sensitivity of the models to the initial seed, we train each neural network based model with its best hyper-parameters (as obtained on the validation set, without considering the test data) for 5 different initial seeds (217, 673, 730, 182, 427). The initial seeds were chosen randomly. For each measurement, we report the average, standard deviation, maximum and minimum values. In addition, we provide the p-value computed in t-test vs. our proposed DS-DCCA. The total canonical correlation statistics are provided in tables II, III and IV for MNIST, XRMB and Flickr8k.

Evidently, for all three datasets (MNIST, XRMB and Flickr8k), our proposed dynamically-scaled Deep CCA (DS-DCCA) learns representations of the two views, which are more canonically correlated than the baselines on average of the 5 different runs. In fact, the minimum result out of the 5 runs of our proposed dynamically-scaled models outperforms the maximum obtained results of the baselines in most of the experiments. The very low p-values emphasize that the leading results of our DS-DCCA are statically significant.

2.2 Cross-Modality Retrieval

We conduct a similar experiment for testing the sensitivity of the leading cross-modality retrieval models to the initialization. In particular, we train the best performing compared baseline, Ranking CCA (R.CCA), and the full-configuration ($\mathbf{z}$ & $\mathbf{x}$) of our proposed DS-R.CCA for 15 different initial seeds (217, 673, 730, 182, 491, 945, 991, 550, 297, 190, 179, 886, 125, 665, 769). For each measurement, we report the average, standard deviation, maximum and minimum values. In addition, we provide the p-values computed in paired t-test vs. our proposed DS-R.CCA. The p-values for Flickr8k, Flickr30k and IAPR TC-12 are provided in table V, the means and standard deviations are provided in table VI and the maximum and minimum results are provided in table VII.

Evidently, our proposed dynamically-scaled Ranking-CCA (DS-R.CCA) achieves better recall rates than the conventionally-scaled Ranking-CCA for the three datasets (Flickr8k, Flickr30k and IAPR TC-12) on average of the 15 runs. The low p-values emphasize that the results are statistically significant.

3 Scales Visualization

For the purpose of visualizing the scales generated by the scaling networks ($\mathbf{h}_1$ and $\mathbf{h}_2$), we provide in Figure 1 the t-SNE plots [3] of the scaling networks' outputs for the two views (left and right halves) of MNIST's test set. The obtained outputs are well clustered according to the input's digit, which is a label information that is not provided during training (the task is to match the two halves of the same digit).

Model	MNIST			
	MEAN±STD	P-VALUE	MAX	MIN
Upper Bound	50.00±0.000	-	50.00	50.00
CCA	29.05±0.000	7.53E-12	29.05	29.05
FKCCA	41.68±0.018	2.22E-10	41.71	41.66
NKCCA	45.01±0.013	5.06E-08	45.04	45.00
DCCA	46.70±0.012	1.48E-06	46.72	46.69
DCCAE	46.71±0.012	5.02E-06	46.73	46.70
NCCA	37.37±0.000	8.29E-11	37.37	37.37
Soft-CCA	44.59±0.117	1.37E-06	44.71	44.42
Soft-HGR	46.81±0.019	5.79E-06	46.83	46.78
ℓ_0-CCA	47.09±0.019	6.10E-06	47.12	47.07
DS-DCCA	**47.49**±0.044	-	**47.56**	**47.44**

Table II: Total canonical correlation statistics for MNIST

Model	XRMB			
	MEAN±STD	P-VALUE	MAX	MIN
Upper Bound	112.00±0.000	-	112.00	112.00
CCA	16.02±0.000	1.50E-15	16.02	16.02
FKCCA	95.17±0.073	1.99E-10	95.30	95.11
NKCCA	103.31±0.052	4.93E-10	103.35	103.25
DCCA	108.66±0.031	5.78E-08	108.69	108.62
DCCAE	108.66±0.014	1.51E-08	108.68	108.64
NCCA	107.31±0.004	9.95E-10	107.31	107.30
Soft-CCA	83.65±3.183	4.54E-05	87.32	79.02
Soft-HGR	106.01±0.049	4.37E-09	106.05	105.92
ℓ_0-CCA	108.66±0.010	1.03E-08	108.68	108.65
DS-DCCA	**110.84**±0.027	-	**110.88**	**110.81**

Table III: Total canonical correlation statistics for XRMB

Model	Flickr8k			
	MEAN±STD	P-VALUE	MAX	MIN
Upper Bound	128.00±0.000	-	128.00	128.00
CCA	41.60±0.000	3.81E-09	41.60	41.60
FKCCA	38.15±0.253	9.92E-10	38.57	37.94
NKCCA	67.35±0.067	1.54E-07	67.45	67.29
DCCA	67.72±0.441	6.31E-07	68.24	67.05
DCCAE	67.65±0.365	1.14E-06	68.11	67.14
NCCA	52.95±0.071	9.07E-09	53.01	52.84
Soft-CCA	60.52±0.468	9.03E-08	61.24	59.94
Soft-HGR	55.18±2.886	1.33E-05	57.97	50.45
ℓ_0-CCA	72.60±0.394	1.24E-06	73.24	72.18
DS-DCCA	**86.04**±0.499	-	**86.76**	**85.53**

Table IV: Total canonical correlation statistics for Flickr8k

Dataset	Task	Model	P-VALUE		
			R_1	R_5	R_{10}
Flickr8k	I → T	R.CCA	7.64E-05	1.12E-06	2.51E-07
		DS-R.CCA	-	-	-
	T → I	R.CCA	2.75E-06	1.50E-08	1.33E-08
		DS-R.CCA	-	-	-
Flickr30k	I → T	R.CCA	6.65E-04	3.89E-07	3.32E-04
		DS-R.CCA	-	-	-
	T → I	R.CCA	1.49E-03	1.89E-05	1.40E-05
		DS-R.CCA	-	-	-
IAPR	I → T	R.CCA	1.77E-02	2.89E-02	3.09E-02
		DS-R.CCA	-	-	-
	T → I	R.CCA	2.04E-02	3.81E-04	4.36E-04
		DS-R.CCA	-	-	-

Table V: Recall rates statistics: p-value computed in t-test vs. our DS-R.CCA (I=image, T=text)

Dataset	Task	Model	MEAN±STD		
			R_1	R_5	R_{10}
Flickr8k	I → T	R.CCA	33.2±1.065	66.1±1.078	78.4±0.995
		DS-R.CCA	**35.6**±0.989	**69.3**±0.881	**81.5**±0.887
	T → I	R.CCA	31.8±0.861	64.3±0.876	77.6±0.795
		DS-R.CCA	**33.7**±0.683	**67.8**±0.795	**80.7**±0.811
Flickr30k	I → T	R.CCA	40.6±1.075	73.0±0.933	83.1±0.833
		DS-R.CCA	**42.4**±1.146	**75.0**±0.595	**84.4**±0.662
	T → I	R.CCA	40.6±0.804	71.6±1.260	82.5±0.735
		DS-R.CCA	**42.0**±0.898	**73.9**±0.627	**84.3**±0.664
IAPR	I → T	R.CCA	48.1±0.947	82.3±0.616	90.3±0.502
		DS-R.CCA	**49.1**±0.958	**82.7**±0.630	**90.8**±0.572
	T → I	R.CCA	49.2±0.791	81.1±0.572	89.7±0.685
		DS-R.CCA	**50.0**±0.794	**81.9**±0.642	**90.5**±0.504

Table VI: Recall rates statistics: mean and standard deviation (I=image, T=text)

Dataset	Task	Model	MAX			MIN		
			R_1	R_5	R_{10}	R_1	R_5	R_{10}
Flickr8k	I → T	R.CCA	35.3	67.9	79.7	31.8	64.5	75.7
		DS-R.CCA	**37.0**	**70.9**	**83.2**	**33.1**	**67.6**	**80.0**
	T → I	R.CCA	34.0	65.7	78.9	30.2	63.0	75.8
		DS-R.CCA	**34.9**	**69.3**	**82.3**	**32.3**	**66.9**	**79.6**
Flickr30k	I → T	R.CCA	42.5	74.2	84.5	38.6	70.9	81.8
		DS-R.CCA	**43.9**	**75.9**	**85.5**	**39.9**	**74.0**	**83.3**
	T → I	R.CCA	42.5	73.5	84.2	39.2	70.0	81.0
		DS-R.CCA	**44.2**	**75.0**	**85.6**	**40.3**	**73.0**	**83.4**
IAPR	I → T	R.CCA	49.6	83.4	91.0	46.7	81.4	89.1
		DS-R.CCA	**50.8**	**84.1**	**91.7**	**47.5**	**81.9**	**90.0**
	T → I	R.CCA	50.6	81.9	90.6	47.8	80.1	88.5
		DS-R.CCA	**50.8**	**83.0**	**91.3**	**48.3**	**80.6**	**89.8**

Table VII: Recall rates statistics: maximum and minimum values (I=image, T=text)

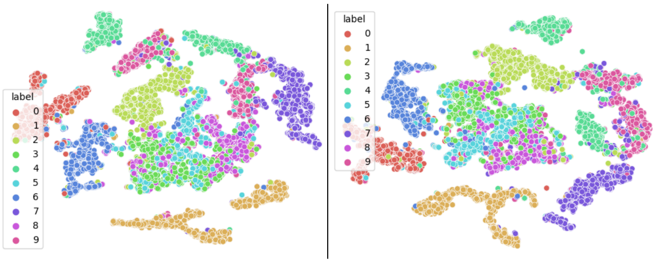

Fig. 1: t-SNE plots of the scaling network outputs for the left and right halves of MNIST. The scales display a per-class behavior despite training without labels.

4 Best Hyper-parameters

For the total canonical correlation and cross-modality retrieval experiments, which are presented in the paper, the hyper-parameters of each model were chosen out of the ranges mentioned in the paper. The best hyper-parameters on each experiment were chosen by the best performing ones on the validation set in terms of the total canonical correlation and each retrieval recall rate, respectively. The neural network based models were initialized by the same initial seed (217) using the default weight initialization of Pytorch.

Following are the best hyper-parameters, which we found for each model and each experiment.

4.1 Total Canonical Correlation

MNIST

- DCCA: regularization coefficients: $r_1 = 1e^{-2}$, $r_2 = 1e^{-4}$
- DCCAE: regularization coefficients: $r_1 = 1e^{-2}$, $r_2 = 1e^{-2}$; reconstruction coefficient=0.01
- NCCA: kernel widths: $\sigma_{x_1} = 50; \sigma_{x_2} = 50$
- Soft-CCA: SDL regularization coefficient=15; Running average $\alpha = 0.97$
- ℓ_0-CCA: regularization coefficients: $r_1 = 1e^{-2}$; $r_2 = 1e^{-4}$; stochastic gate regularization: 0.5; standard deviation of the stochastic gate's noise=0.75
- DS-DCCA: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-2}$; scaling network layers=(256,128,c)

XRMB

- DCCA: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$
- DCCAE: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; reconstruction coefficient=0.01
- NCCA: kernel widths: $\sigma_{x_1} = 0.4; \sigma_{x_2} = 0.4$
- Soft-CCA: SDL regularization coefficient=10; Running average $\alpha = 0.95$

- ℓ_0-CCA: regularization coefficients: $r_1 = 1e^{-4}$; $r_2 = 1e^{-4}$; stochastic gate regularization: 0.01;standard deviation of the stochastic gate's noise=0.1
- DS-DCCA: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; scaling network layers=(256,c)

Flickr8k

- DCCA: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$
- DCCAE: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; reconstruction coefficient=0.1
- NCCA: kernel widths: $\sigma_{x_1} = 0.7; \sigma_{x_2} = 0.4$
- Soft-CCA: SDL regularization coefficient=5; running average coefficient $\alpha = 0.95$
- ℓ_0-CCA: regularization coefficients: $r_1 = 1e^{-4}$; $r_2 = 1e^{-4}$; stochastic gate regularization: 0.5; standard deviation of the stochastic gate's noise=0.75
- DS-DCCA: regularization coefficients: $r_1 = 1e^{-2}$, $r_2 = 1e^{-6}$; scaling network layers=(256,c)

4.2 Cross-Modality Retrieval

Flickr8k

- Ranking CCA:
  - IMG → TXT:
    - R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.8$;running average coefficient $= 0.9$
    - R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.8$;running average coefficient $= 0.97$
    - R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.8$;running average coefficient $= 0.95$
  - TXT → IMG:
    - R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.8$;running average coefficient $= 0.9$
    - R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.7$;running average coefficient $= 0.9$
    - R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.7$;running average coefficient $= 0.9$
- DS-Ranking CCA ($\mathbf{z}$ & $\mathbf{x}$):
  - IMG → TXT:
    - R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.7$; running average coefficient $= 0.97$; scaling network layers=(128,c)
    - R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.7$; running average coefficient $= 0.97$; scaling network layers=(256,c)

    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.7$; running average coefficient = 0.97; scaling network layers=(256,c)
  - TXT → IMG:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.7$; running average coefficient = 0.97; scaling network layers=(256,c)
    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.7$; running average coefficient = 0.97; scaling network layers=(256,c)
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.7$; running average coefficient = 0.97; scaling network layers=(256,c)

Flickr30k

- – Ranking CCA:
  - IMG → TXT:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.5$;running average coefficient = 0.9
    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$;running average coefficient = 0.85
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$;running average coefficient = 0.85
  - TXT → IMG:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$;running average coefficient = 0.95
    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$;running average coefficient = 0.97
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.5$;running average coefficient = 0.9
- – DS-Ranking CCA (**z** & **x**):
  - IMG → TXT:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient = 0.85; scaling network layers=(256,c)
    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient = 0.85; scaling network layers=(256,128,c)
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient = 0.85; scaling network layers=(256,128,c)
  - TXT → IMG:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient = 0.85; scaling network layers=(256,c)

    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient $= 0.85$; scaling network layers=(256,c)
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient $= 0.85$; scaling network layers=(128,c)

IAPR TC-12

- Ranking CCA:
  - IMG → TXT:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-6}$; ranking margin $m = 0.6$;running average coefficient $= 0.97$
    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-6}$; ranking margin $m = 0.6$;running average coefficient $= 0.85$
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-6}$; ranking margin $m = 0.6$;running average coefficient $= 0.97$
  - TXT → IMG:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-6}$; ranking margin $m = 0.7$;running average coefficient $= 0.9$
    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-6}$; ranking margin $m = 0.6$;running average coefficient $= 0.9$
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-6}$; ranking margin $m = 0.6$;running average coefficient $= 0.9$
- DS-Ranking CCA ($\mathbf{z}$ & $\mathbf{x}$):
  - IMG → TXT:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient $= 0.9$; scaling network layers=(256,c)
    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient $= 0.95$; scaling network layers=(256,c)
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient $= 0.95$; scaling network layers=(256,c)
  - TXT → IMG:
    - ∗ R_1: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient $= 0.85$; scaling network layers=(256,c)
    - ∗ R_5: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient $= 0.85$; scaling network layers=(256,c)
    - ∗ R_{10}: regularization coefficients: $r_1 = 1e^{-4}$, $r_2 = 1e^{-4}$; ranking margin $m = 0.6$; running average coefficient $= 0.95$; scaling network layers=(128,c)

5 Computing Infrastructure

All neural network-based models were implemented in Pytorch 1.4 and were trained on a single GPU (NVIDIA GEFORCE RTX 2080 Ti). FKCCA and NKCCA were trained using their official implementation on a CPU (Intel(R) Xeon(R) Silver 4114). NCCA was trained using its official MATLAB implementation on a CPU (Intel(R) Core-i7).

NCCA was trained on a PC computer, whose operating system is Windows 10 and it has 8GB of RAM. All other models were trained on a computer, whose operating system is Ubuntu 18.04.6 LTS and it has a total of 252GB of RAM.

References

1. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning. pp. 448–456. PMLR (2015)
2. Lopez-Paz, D., Sra, S., Smola, A., Ghahramani, Z., Schölkopf, B.: Randomized nonlinear component analysis. In: International conference on machine learning. pp. 1359–1367. PMLR (2014)
3. Van der Maaten, L., Hinton, G.: Visualizing data using t-sne. Journal of machine learning research **9**(11) (2008)
4. Michaeli, T., Wang, W., Livescu, K.: Nonparametric canonical correlation analysis. In: International conference on machine learning. pp. 1967–1976. PMLR (2016)
5. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Icml (2010)
6. Tieleman, T., Hinton, G.: Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning (2012)
7. Wang, W., Arora, R., Livescu, K., Srebro, N.: Stochastic optimization for deep cca via nonlinear orthogonal iterations. In: 2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton). pp. 688–695. IEEE (2015)