Continual learning in cross-modal retrieval

Kai Wang[1], Luis Herranz[1], Joost van de Weijer[1]
[1] Computer Vision Center, Universitat Autònoma de Barcelona, Barcelona, Spain
{kwang,lherranz,joost}@cvc.uab.es

Background

- Cross-modal Retrieval
  - Metric:
    - Flickr30K: Recall@K **Bi-directional**
    - MSCOCO: Results on 1000 test images and their corresponding sentences
- Continual Learning
  - **Other names**: lifelong learning, sequential learning or incremental learning
  - **Key problem**: catastrophic forgetting (CF) of old concepts as new ones learnt
  - Learning representations for a new domain (called a **task**)

Introduction

- Continual learning + Cross-modal Retrieval ?
- Retrieval: Traning -> Indexing -> Query
  - Pay special attention to the role of “indexing” stage
- Contribution:
  - A continual cross-modal retrieval framework
  - Identify and study the different factors lead to forgettting in cross-modal embeddings and retrieval
    - Study modifications in the retrieval framework, network archi. and regularization

Continual Cross-modal Retrieval

- Cross-modal Deep Metric Learning
  - Two-branch network: image-specific & text-specific
  - Aligned with similarity matrix *S* (binary)
  - Constraints:
    - x: image
    - y: text

$$d(x_i, y_j) + m \le d(x_i, y_k) \quad \text{s.t. } s_{ij} = 1 \text{ and } s_{ik} = 0 \tag{1}$$

and (in the other direction)

$$d(y_{i'}, x_{j'}) + m \le d(y_{i'}, x_{k'}) \quad \text{s.t. } s_{i'j'} = 1 \text{ and } s_{i'k'} = 0 \tag{2}$$

where m is the predefined margin. The triplets are con-

  - Loss:

$$\begin{aligned} L_{\mathrm{T}}(\mathcal{X}, \mathcal{Y}) &= \lambda_1 \sum_{i,j,k} \left[d(x_i, y_j) + m - d(x_i, y_k)\right]_+ \\ &+ \lambda_2 \sum_{i',j',k'} \left[d(y_{i'}, x_{j'}) + m - d(y_{i'}, x_{k'})\right]_+ \end{aligned} \tag{3}$$

Continual Cross-modal Retrieval

- Training, indexing and query stages:
  - Training: Learning embedding networks
  - Indexing:
    - Construct a database expressed with embeddings
    - Training data are not necessarily same as indexing data
    - For simplicity, consider they are the same
  - Query: Compute similarity between a query sample and the **index** data
  - Deplyed system only conduct query

Continual Cross-modal Retrieval

- Continual Learning in Retrieval
  - Setting:
    - Data are presented as a sequence of tasks
    - Each task involves data from a different domain (animal, vehicle ...)
    - Embedding networks are updated
    - Evaluation with seperate data from each task
    - **Classify a negative pair as intra-task neg. pair (ITNP) & cross-task neg. pair (CTNP)**
      - CNTPs are not available during training
    - **Assume all positive pairs are intra-task**

Continual Cross-modal Retrieval

- Continual Retrieval:
  - Reindexing or not?

Reindexing

- Index both data of current and previous tasks
- Database and query are processed with the same network
- Time & Resourse consuming

No Reindexing

- Only index data of current task
- Efficient
- Asymmetry, query embeddings are extracted with new operators while database embedding are extracted with old ones

Catastrophic Forgetting

- Embedding networks:
  - Parameters drift from priviously optimal values
- Embedding misalignment:
  - Embeddings of different modalities may drift differently
- Task overlap:
  - CTNPs are the only force to discriminate samples of different tasks

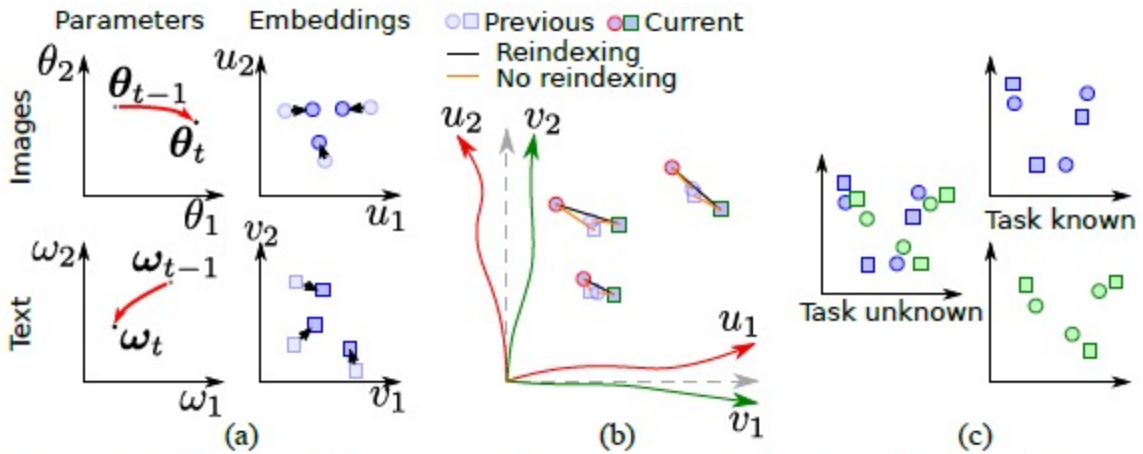

Figure 4. Causes of forgetting in cross-modal embeddings: (a) embedding networks become less discriminative due to drift in parameter space, and (b) unequal drift increases cross-modal misalignment, and (c) task overlap in embedded space (when task is unknown). Best viewed in color.

Preventing Forgetting

- Embeddding drift
  - Regularization term: To penalize weighted Euclidean distance

$$L_{\mathrm{R}} = \sum_{k} \Theta_{k}^{(t-1)} \left(\theta_{k}^{(t-1)} - \theta_{k}\right)^{2} + \sum_{k'} \Omega_{k'}^{(t-1)} \left(\omega_{k'}^{(t-1)} - \omega_{k'}\right)^{2}$$

  - Θ and Ω are iteratable weights (initialized as 0)
    - Methods to iterate are left out
  - Final loss:

$$L = L_{\mathrm{T}} + \lambda_{3} L_{\mathrm{R}}$$

Preventing Forgetting

- Unequal Drift
  - Tying the networks by sharing layers at the top
  - Bottom layers must remain modality-specific
- Decoupling Retrieval directions
  - In the case of no reindexing
  - Beneficial when image and text embeddings drift in different directions
- Cross-task overlap
  - Weight regularization and sharing layers could help

Experiments

- Settings:
  - Joint vs Continual
  - I2T & T2I
  - Known task & Unknown task
  - Reindexing
  - Weight regularization
  - Decoupled directions
  - Layer sharing

Experiments

- Sequential Visual Genome (SeViGe)
  - Divide Visual Genome into three domains: animals, vehicles and clothes

Domain	*im2txt*										*txt2im*									
	Joint		*Continual*								*Joint*		*Continual*							
	CTNP		*reindexing*			*no reindexing*					*CTNP*		*reindexing*			*no reindexing*				
	Yes	No	*ft*	*EWC*	*MAS*	*ft*	*EWC*	*EWC-im*	*MAS*	*MAS-im*	Yes	No	*ft*	*EWC*	*MAS*	*ft*	*EWC*	*EWC-txt*	*MAS*	*MAS-txt*
Architecture: *no sharing*																				
animals	29.1	26.0	16.1	16.8	16.9	24.5	24.6	24.2	24.7	24.3	27.8	25.9	15.4	15.2	15.4	20.8	20.8	20.9	19.8	20.7
vehicles	30.9	27.7	20.8	23.3	22.7	24.0	25.1	24.8	26.0	24.8	30.9	27.0	17.5	18.6	19.5	27.2	29.4	28.0	28.8	28.7
clothes	27.9	27.5	27.4	27.0	27.5	27.4	27.0	27.3	27.5	26.3	29.3	27.7	28.1	27.5	28.0	28.1	27.5	27.4	28.0	28.5
average	29.3	27.0	21.5	22.3	**22.4**	24.5	24.6	24.2	**24.7**	24.3	29.3	26.8	20.3	20.5	**21.0**	25.4	25.9	25.4	25.6	**26.0**
A+V+C	28.5	24.4	17.0	**18.4**	17.8	18.6	17.9	17.5	**19.0**	18.3	28.0	23.8	16.3	16.3	**16.9**	20.7	21.3	20.9	20.9	**21.4**
Architecture: *sharing*																				
animals	28.3	25.3	18.4	17.1	16.4	23.1	21.2	21.4	21.1	21.4	26.8	24.4	16.6	14.8	14.3	22.1	20.7	21.1	20.6	22.2
vehicles	30.2	28.6	22.6	24.7	23.5	23.0	24.9	25.0	23.8	26.0	31.2	27.9	16.9	17.8	16.3	27.3	29.4	29.5	28.4	28.7
clothes	26.7	27.4	27.7	26.9	27.1	27.7	26.9	27.3	27.1	26.7	27.5	26.8	27.2	27.0	26.0	27.2	27.0	27.5	26.0	28.0
average	28.4	27.1	22.9	**22.9**	22.3	24.6	24.3	24.6	24.0	**24.7**	28.5	26.4	**20.3**	19.9	18.9	25.6	25.7	26.0	25.0	**26.3**
A+V+C	27.8	24.5	18.2	**18.2**	17.6	**19.0**	17.9	18.2	17.9	18.8	27.2	23.7	**15.9**	15.5	14.9	21.8	21.5	22.2	21.0	**22.6**

Table 1. Results in SeViGe after learning all tasks (Recall@10 in %). *average* measures performance with *known* task, while *A+V+C* with *unknown* task. Best joint learning result in green, best continual learning result in **red**.

Experiments

- Sequential MS-COCO (SeCOCO)
  - Challenging to organize data into tasks

Domain	*im2txt*											*txt2im*										
	Joint		*Continual*									*Joint*		*Continual*								
	CTNP		*reindexing*			*no reindexing*						*CTNP*		*reindexing*			*no reindexing*					
	Yes	No	*ft*	*EWC*	*MAS*	*ft*	*EWC*	*EWC-im*	*MAS*	*MAS-im*	Yes	No	*ft*	*EWC*	*MAS*	*ft*	*EWC*	*EWC-txt*	*MAS*	*MAS-txt*
Architecture: *no sharing*																					
task1	65.7	63.8	33.6	32.0	33.0	49.8	48.1	47.2	50.5	47.1	69.7	68.2	40.1	38.0	38.2	59.8	59.2	58.3	60.0	59.7
task2	56.5	54.9	39.8	38.5	40.0	47.0	46.6	46.4	47.0	46.9	65.2	62.6	46.8	44.7	46.9	54.6	55.5	55.1	55.5	55.9
task3	38.2	39.9	39.7	40.1	40.2	39.7	40.1	39.9	40.5	39.7	44.6	45.7	46.7	46.7	46.0	46.7	46.7	46.7	46.0	46.2
average	53.5	52.9	**37.7**	36.9	**37.7**	45.5	44.9	44.5	**46.0**	44.6	59.8	58.9	**44.5**	43.1	43.7	53.7	53.8	53.4	53.8	**54.0**
total	52.4	49.8	**33.0**	32.1	**33.0**	37.1	36.2	35.6	**37.4**	36.0	58.5	56.3	**40.4**	38.7	39.7	48.3	48.0	47.3	48.2	**48.4**
Architecture: *sharing*																					
task1	65.3	63.9	32.9	31.9	34.1	48.4	47.7	47.7	47.8	45.1	70.2	67.7	38.2	37.4	39.8	58.6	56.3	58.4	57.1	57.5
task2	55.7	55.3	40.6	39.9	40.4	46.3	46.0	45.2	44.0	44.4	64.7	63.1	46.0	45.7	46.3	54.6	54.2	55.6	54.6	54.9
task3	37.6	40.1	39.6	39.7	39.3	39.6	39.7	39.9	40.0	39.7	44.8	46.5	46.2	45.8	45.7	46.2	45.8	45.7	46.7	46.1
average	52.9	53.1	37.7	37.2	**37.9**	**44.8**	44.5	44.3	43.9	43.1	59.9	59.1	43.5	43.0	**43.9**	53.1	52.1	**53.2**	52.8	52.8
total	51.8	50.1	33.2	32.5	**33.5**	**36.1**	35.9	35.4	35.5	35.3	58.7	56.4	39.3	38.9	**39.9**	47.7	46.8	**48.1**	47.1	47.5

Table 2. Results in SeCOCO after learning all tasks (Recall@10 in %). *average* measures performance with *known* task, while *total* with *unknown* task. Best joint learning result in green, best continual learning result in red.

Conclusion

- A piece of “digging hole” work
- Massive experiments
- Lack of dataset and “tasks”