Sequential Learning for Cross-modal Retrieval

Ge Song[1,2,3] and Xiaoyang Tan[1,2,3*]
[1]College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics
[2]MIIT Key Laboratory of Pattern Analysis and Machine Intelligence
[3]Collaborative Innovation Center of Novel Software Technology and Industrialization
{sunge,x.tan}@nuaa.edu.cn

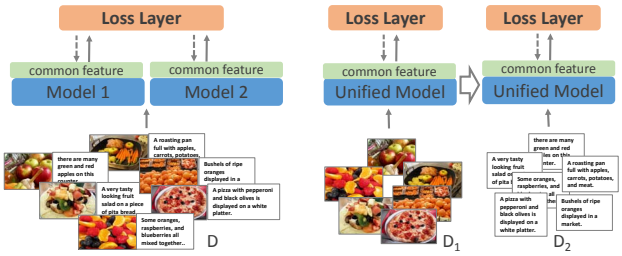
Abstract

Cross-modal retrieval has attracted increasing attention with the rapid growth of multimodal data, but its learning paradigm under changing environment is less studied. Inspired by the recent achievement in the field of cognition mechanism on how the human brain acquires knowledge, we propose a new sequential learning method for cross-modal retrieval. In this method, a unified model is maintained to capture the common knowledge of various modalities but are learnt in a sequential manner such that it behaves adaptively according to the evolving distribution of different modalities, and needs no laborious alignment operations among multimodal data before learning. Furthermore, we propose a novel meta-learning based method to overcome the catastrophic forgetting encountered in sequential learning. Extensive experiments are conducted on three popular multimodal datasets, showing that our method achieves state-of-the-art cross-modal retrieval performance without any modal-alignment.

1. Introduction

Cross-modal retrieval, aiming to search instances in one modality that display similar content as the query from another modality, has gained increasing attention from both industrial and academic communities due to its wide usage, e.g., sketch-based image retrieval in the criminal investigation. The difficulty of the measurement of content similarity among data from different modalities, which is known as the heterogeneity gap [4], makes this task very challenging. Thus, bridging the heterogeneity gap between different modalities plays a key role in cross-modal retrieval.

Many methods [9, 21] have been developed to learn mapping different modalities into a shared feature space, such that the data of different modalities become computationally comparable. Due to the low storage costs and the high computational efficiency of binary codes, hashing-based methods [19, 18] also have been extended to cross-modality retrieval by embedding the data of interest into a low-dimensional Hamming space. We observe that these methods are built in the same manner: developing individual sub-models for each modality and jointly learning them by aligned multi-modal data. We call this manner as *parallel cross-modal learning* (PCML), which is shown in Fig.1 (a). Despite the effectiveness of this parallel learning paradigm, it is unlikely to be adapted without retraining the whole system under a real-world environment when the underlying distribution of different modalities are gradually changing.



(a) Parallel learning (b) Sequential learning

Figure 1. Illustration of the difference between two cross-modal learning paradigms: parallel and sequential. In the parallel paradigm, the whole architecture involves multiple individual sub-models with each responsible for one modality, and well-aligned multi-modal data (e.g., image-text pairs) are needed to jointly train them, while in the sequential paradigms, a single unified model is used to map all modalities into a common feature space, and the model is trained on different modalities sequentially.

Recent work in cognitive science reveals that when a sequence of multimodal signals stimulates our brain, it is able to automatically integrate the elements from different modalities into one unitary representation [14]. In other words, our brain acquires knowledge or concepts across different modalities in a sequential learning manner (i.e., modality-by-modality). In contrast with PCML, this sequential manner is more practical in real-life scenario: 1)

*Corresponding author

it is easier for us to learn a stable conceptual representations from one modality than that from multi-modalities simultaneously; 2)we can adaptively adjust the learned distribution when a new modality is available, which is more robust to the concept drift across modalities; 3) by performing cross-modal learning on one modality first and then on another, we avoid the needs that the training data are aligned among various modality at a fine-grained level, e.g., in the form of image-text pairs. Inspired by above observations, we propose to perform cross-modal learning in a sequential manner, which is called as *sequential cross-modal learning* (SCML) and is illustrated in Fig 1 (b). In SCML, a unified model is maintained to capture the common representation of different modalities and is trained sequentially.

However, training a model with new information could interfere with the previously learned knowledge, which is often referred to as *catastrophic forgetting* [15, 20, 22, 32]. This phenomenon typically leads to an abrupt performance decrease or, in the worst case, to the completely overwritten of old knowledge by the new one. Some evidence has suggested that inappropriate changes of specific parameters for old tasks tend to cause catastrophic forgetting [15, 20] and appropriate optimization for those parameters is of importance. We notice that the optimization-based meta-learner [3] can be trained on massive same single old tasks for effectively and fast optimizing new model on the new task with limited samples. Similarly, we can design a new meta-learner which is trained on multi-tasks (contain new and old tasks) to learn to optimize the old model for performing well on the new task and keeping the performance of old tasks. Motivated by this, we propose a novel LSTM-based meta-learner to address the catastrophic forgetting issue. The main contributions of this paper are summarized as follows:

- A novel sequential cross-modal learning method (SCML)is proposed which is consistent with the cognitive mechanism of human beings in acquiring knowledge across multi-modalities. In contrast with previous methods, SCML is more adaptive to the evolving distributions of different modalities, and it does not need the laborious alignment operations for multimodal data before learning, enabling the learning to be more flexible in practice.

- A new meta-learning method is proposed to handle the catastrophic forgetting problem in sequential learning. In details, a special LSTM-based meta-learner is designed to learn to effectively optimize the old model for the new task and maintain previous knowledge.

- Extensive experimental results demonstrate that the SCML method can perform cross-modal learning well in sequential manner and yield state-of-the-art retrieval performance on three cross-modality datasets.

2. Related Work

Cross-modal retrieval. Cross-modal learning approaches [2, 9, 5, 19, 31, 16] can roughly be divided into continual-value learning method and hashing method. The key idea of the former is to map heterogeneous data into a continual-value shared space to account for the diversity of different modalities. Wu et al. [30] propose a semantic structure-preserved embedding learning method based on the semantic structure and local geometric structure consistency. The hashing method [19, 31, 16, 26] seeks to encode high-dimensional features into compact binary codes, hence enabling fast similarity search with Hamming distances. Li et al. [16] propose a self-supervised adversarial hashing (SSAH) approach, which attempts to incorporate adversarial learning into cross-modal hashing. Despite their effectiveness, most of them assume the availability of a large number of matched aligned cross-modal pairs which are unfortunately not always available.

Sequential learning. The sequential learning also can be called as continual or lifelong learning which refers to the ability to continually learn over time by accommodating new knowledge while retaining previously learned experiences. The main issue of sequential learning model is catastrophic forgetting. Massive methods [15, 25, 1, 22] have attempted to mitigate catastrophic forgetting, and they can mainly be classified into three types. The first is regularization approaches, which impose different constraints on the update of the neural weights to alleviate catastrophic forgetting. Kirkpatrick et al. [15] proposed a model called elastic weight consolidation (EWC) where a quadratic penalty is used to slow down the learning for task-relevant weights coding for previously learned knowledge. The second is dynamic architectures [32, 24], which change architectural properties in response to new information by dynamically accommodating novel neural resources, e.g., re-training with an increased number of neurons or network layers. The last is the memory-based method [20], which uses a set of previous tasks data to constraint optimizing. Most of previous methods are designed for single modality and have not been verified on large-scale cross-modal datasets.

3. The proposed method

In this section, we give the problem definition of SCML and detail the model structure and the learning steps.

3.1. The problem definition

Without loss of generality, we focus on sequential cross-modal learning for bi-modality (i.e., image and text). Our goal is to learn a unified model in a sequential manner that maps different modalities into a common feature space. Suppose that we are firstly given a training set of N_1 images $\mathrm{D}^{(1)} = \{\mathrm{x}_\mathrm{i}^{(1)}, \mathrm{y}_\mathrm{i}^{(1)}\}_{\mathrm{i}=1}^{\mathrm{N}_1}$, $\mathrm{x}_\mathrm{i}^{(1)} \in \mathrm{R}^{\mathrm{d}_1}$, $\mathrm{y}_\mathrm{i}^{(1)} \in \{0,1\}^{\mathrm{C}}$, where

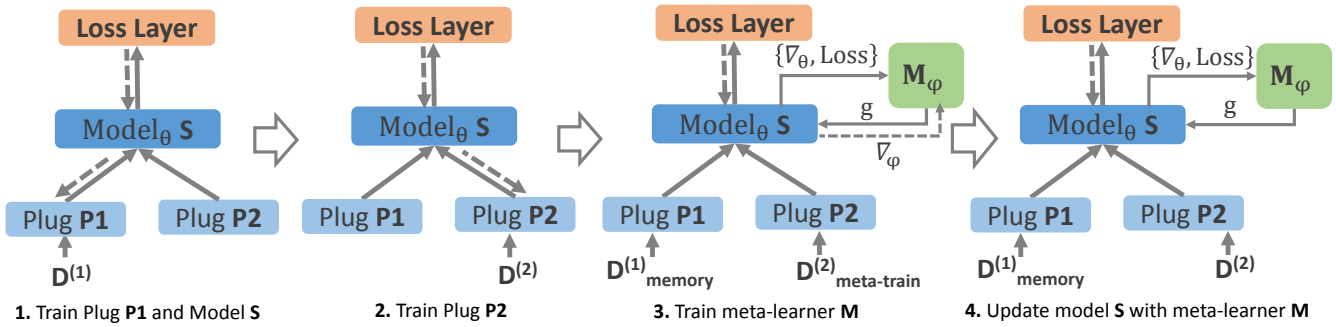


Figure 2. Overview of sequential cross-modal learning on two modalities. The architecture consists of four components (two plugs P1, P2, a unified model S, and a meta-learner M) and four steps: The model firstly takes one modality $\mathrm{D}^{(1)}$ to jointly learn P1 and S. When the new modality $\mathrm{D}^{(2)}$ is available, the P2 is trained with S fixed to avoid the forgetting of S for $\mathrm{D}^{(1)}$. After that, M is learned to update S with a pre-preserved set $\mathrm{D}^{(1)}_{\mathrm{memory}}$ of $\mathrm{D}^{(1)}$ and a set $\mathrm{D}^{(2)}_{\mathrm{meta-train}}$ of $\mathrm{D}^{(2)}$. Finally, the S is updated by the learnt M with $\mathrm{D}^{(1)}_{\mathrm{memory}}$ and $\mathrm{D}^{(2)}$.

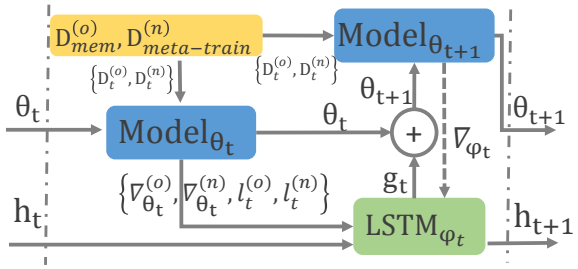


Figure 3. One step of training meta-learner.

C is the number of class. The first goal is jointly to learn two nonlinear functions: $f_1 : x^{(1)} \mapsto z^{(1)} \in R^d$ from image feature space $\mathrm{R}^{\mathrm{d}_1}$ to input space R^{d} of the unified model, $f : x \mapsto h \in R^K$ from input space R^{d} to common feature space R^{K} with semantic-preserving. Then the $\mathrm{D}^{(1)}$ is discarded and we are given a new training set of N_2 text $\mathrm{D}^{(2)} = \{\mathrm{x}^{(2)}_{\mathrm{i}}, \mathrm{y}^{(2)}_{\mathrm{i}}\}^{\mathrm{N}_2}_{\mathrm{i}=1}$, where $\mathrm{x}^{(2)}_{\mathrm{i}} \in \mathrm{R}^{\mathrm{d}_2}$ is associated with the same categories as images. The second goal is to learn a nonlinear function $f_2 : x^{(2)} \mapsto z^{(2)} \in R^d$ from text feature space $\mathrm{R}^{\mathrm{d}_2}$ to input space R^{d} of the unified model and to update f for mapping $z^{(2)}$ while the semantics of both image and text are preserved.

3.2. The structure of the model

Our SCML model consists of four components: plugs P1, P2, a unified model S, meta-learner M (see Sec.3.3).

Plugs P1 and P2: Plugs are designed respectively for mapping original features (e.g. CNN or hand-crafted) of different modalities into the same dimension. For flexible expansion, we implement them as two deep neural networks.

Unified Model S: The unified model S is a general model which mappings the outputs of different plugs into a common space with semantic-preserving, so the capacity of S should be large enough to store knowledge from multi-modal sources. For this, we build S as a 3-layers fully-connect neural network. To speed up retrieval, we quantize the output h of last layer by simple quantization $b = sign(h)$ to obtain binary common representation.

3.3. Meta-learner for catastrophic forgetting

In regular meta-learning scheme, a meta-learner M is trained on a meta-train task set (e.g., classification) $\mathcal{T}_{train} = \{T^{(i)}\}_{i=1}^{N_T}$ to learn to optimize corresponding learners (e.g., classifier) $\{f^{(i)}\}_{i=1}^{N_T}$, and it is used to optimize the learner $f^{(test)}$ of meta-test task $\mathcal{T}_{test} = T^{(test)}$, where each task $T^{(i)}$ associates with a dataset $D^{(i)}$, the optimized learner $f^{(i)}_*$ for task $T^{(i)}$ is $f^{(i)}_* = M(D^{(i)}, f^{(i)}; \varphi)$, φ is the parameter of M. Let $l(D, f)$ denotes the loss function of f, the loss function of M can be defined as follows:

$$\mathcal{L} = \sum_{i=1}^{N_T} l\big(D^{(i)}, M(D^{(i)}, f^{(i)}; \varphi)\big) \tag{1}$$

After training, the optimized learner $f^{(test)}_*$ of task $T^{(test)}$ can be obtained by $f^{(test)}_* = M(D^{(test)}, f^{(test)}; \varphi)$.

In our meta-learning scheme, the meta-learner M is learned to update the unified learner $f^{(o)}_s$ (well trained on $D^{(o)}$) on the new dataset $D^{(n)}$ for performing task $T^{(n)}$ well without catastrophic forgetting $T^{(o)}$'s performance. Thus, the objective of M can be formulated as follows:

$$\begin{aligned} &\mathcal{L} = l(D^{(n)}, M(D^{(n)}, f^{(o)}_s; \varphi)) \\ &s.t.\ l(D^{(o)}, M(D^{(n)}, f^{(o)}_s; \varphi)) \le l(D^{(o)}, f^{(o)}_s) \end{aligned} \tag{2}$$

We rephrase the constraint term as the task of better performance on $T^{(o)}$, then Eq.(2) is rewritten as follows:

$$\mathcal{L} = l(D^{(n)}, M(D^{(n)}, f^{(o)}_s)) + l(D^{(o)}, M(D^{(n)}, f^{(o)}_s)) \tag{3}$$

The first term encourages the M to update $f^{(o)}_s$ for better performance of new task, while the second term impose M

to update $f_s^{(o)}$ for less forgetting old task knowledge. However, two facts makes the optimization of Eq.(3) be impossible: 1) the whole dataset $D^{(o)}$ and $D^{(n)}$ are not available simultaneously in practice; 2) the regular M cannot effectively minimize the second term since the lack of $T^{(o)}$ information. To handle these problems, 1) we remain $\mathrm{N_{mem}}$ samples of $D^{(o)}$ as the episodic memory $D_{mem}^{(o)}$ to keep the $T^{(o)}$'s information and randomly select $N_{meta-train}^{(n)}$ samples of $D^{(n)}$ as $D_{meta-train}^{(n)}$ for meta-training; 2) we modify the regular M to make it be able to take in old and new tasks information from $D_{mem}^{(o)}$ and $D^{(n)}$. Then, the Eq.(3) is rewritten as follows:

$$\begin{aligned}\mathcal{L} = & \, l\big(D_{meta-train}^{(n)}, M(D_{mem}^{(o)}, D_{meta-train}^{(n)}, f_s^{(o)}; \varphi)\big) \\ & + l\big(D_{mem}^{(o)}, M(D_{mem}^{(o)}, D_{meta-train}^{(n)}, f_s^{(o)}; \varphi)\big)\end{aligned} \tag{4}$$

In the meta-testing phase, the update of $f_s^{(o)}$ is performed by the well trained meta-learner M. The new learner $f_s^{(n)}$ can be obtained as follows:

$$f_s^{(n)} = M(D_{mem}^{(o)}, D^{(n)}, f_s^{(o)}; \varphi) \tag{5}$$

In implementation, we adopt the LSTM-based meta-learner (in [3], a two-layer LSTMs with 20 hidden units in each layer) as the original M, which learns to output good update for the learner f at each optimization step of f. If we take the update g_t as the output of this original M at t step, the objective of M on the entire optimization trajectory of f will be clear. For one meta-training task, we have:

$$\begin{aligned}\mathcal{L} &= \sum_{t=1}^{T} l_t(D_t, \theta_t) \\ \theta_{t+1} &= \theta_t + g_t, [g_t, h_{t+1}] = M([\nabla_{\theta_t}, l_t], h_t; \varphi)\end{aligned} \tag{6}$$

where T denotes the number of training step, D_t denotes the batch data at t step, θ_t is the parameter of learner f at t step, h_t is the hidden state of M at t step, $\nabla_{\theta_t} = \partial l_t / \partial \theta_t$.

According to Eq.(4), we modify above meta-learner via expanding its inputs to accommodate both old and new tasks information, i.e., $[\nabla_\theta, l]$ to $[\nabla_\theta^{(o)}, \nabla_\theta^{(n)}, l^{(o)}, l^{(n)}]$, where $l^{(*)}$ and $\nabla_\theta^{(*)} = \partial l^{(*)} / \partial \theta$ denote the loss and gradients of dataset $D^{(*)}$ respectively. Meanwhile, we hope that the updates g are sparse to make the θ changes little. Thus, the loss function of modified M is written as follows:

$$\begin{aligned}\mathcal{L} = & \sum_{t=1}^{T} \big(l_t^{(n)}(D_t^{(n)}, \theta_t) + l_t^{(o)}(D_t^{(o)}, \theta_t) + \lambda |g_t|_1\big) \\ & \theta_{t+1} = \theta_t + g_t \\ & [g_t, h_{t+1}] = M([\nabla_{\theta_t}^{(o)}, \nabla_{\theta_t}^{(n)}, l_t^{(o)}, l_t^{(n)}], h_t; \varphi)\end{aligned} \tag{7}$$

where λ balances the sparse term, $D_t^{(n)}$ and $D_t^{(o)}$ are batches of $D_{meta-train}^{(n)}$ and $D_{mem}^{(o)}$ at t step respectively. Notably, the ∇_θ and l in Eq.(7) have very different magnitudes so that M cannot work robustly. So we preprocess M's inputs by the following formula:

$$x \to \begin{cases} \big(\dfrac{log(|x|)}{p}, sgn(x)\big), if |x| \geq e^{-p} \\ (-1, e^p x), otherwise \end{cases} \tag{8}$$

where $p > 0$ is a parameter controlling small disregard values, and we set it to 10 in all experiments according to [3].

Another practical problem is that there are tens of thousands of parameters θ in f need to be updated, but it is impossible to register new LSTMs for each parameter. To avoid this difficulty, we only learn a small LSTMs as M to operate coordinatewise on θ, i.e., all θ_i shares one M.

Optimization. Due to the limited computational resources, we minimize the Eq.(7) in a step-by-step way instead of optimizing the entire trajectory (tens of thousands of parameters need to be stored at each time-step) by Back-propagation Through Time (BPTT). For this, at each optimizing time-step (the computational graph is detailed in Fig.3), we will minimize the following objective with gradient descent method:

$$\begin{aligned}\mathcal{L}_t &= l_t^{(n)}(D_t^{(n)}, \theta_t) + l_t^{(o)}(D_t^{(o)}, \theta_t) + \lambda |g_t|_1 \\ \theta_{t+1} &= \theta_t + g_t \\ [g_t, h_{t+1}] &= M([\nabla_{\theta_t}^{(o)}, \nabla_{\theta_t}^{(n)}, l_t^{(o)}, l_t^{(n)}], h_t; \varphi_t)\end{aligned} \tag{9}$$

the update of φ can be roughly denoted as $\varphi_{t+1} = \varphi_t - \frac{\partial \mathcal{L}_t}{\partial \varphi_t}$.

3.4. Sequential cross-modal learning

Based on the above model, we perform sequential cross-modal learning in four steps, which is shown in Fig. 2.

Stage I: learn plug P1 and unified model S. When the first modality $D^{(1)}$ comes, we train the plug P1 and the unified model S initially. Since the aligned multi-modal data is missing, we only consider to learn the semantic-preserved representation of data by minimizing cross-entropy loss with stochastic gradient descent (SGD), that is the loss layer of S is a softmax layer (for multi-class data) or a sigmoid layer (for multi-label data). The loss is defined as follows:

$$\begin{aligned}l(D^{(1)}; \theta) = - \sum_{i=1}^{N_1} \sum_{j=1}^{C} \Big(& (w_p \cdot y_{ij} \log(\hat{y_{ij}}) \\ & + (1 - y_{ij}) \log(1 - \hat{y_{ij}})\Big)\end{aligned} \tag{10}$$

where θ is parameters of P1 and S, $\hat{y}$ is the predict label, w_p is the weight of positive points. If y is a multi-class vector, then w_p is set to 1, if y is a multi-label vector, then $w_p > 1$.

After training, we randomly remain N_{mem} training samples from $D^{(1)}$ as the episodic memory $D_{mem}^{(1)}$ to keep current knowledge. This memory will be replayed later to guide no-forgetting meta-learning.

Stage II: learn plug P2. When the second modality $D^{(2)}$ is available, we train the plug P2 for transforming $D^{(2)}$ into the unified dimension and fine-tune S for representing $D^{(2)}$. However, the *catastrophic forgetting* will occur when directly adopting gradient descent algorithm (e.g., SGD) to optimize P2 and S jointly, which is observed and discussed in Sec. 4.3. Intuitively, since the S contains certain knowledge to distinguish $D^{(1)}$ samples in some high-level space (or distribution), we can first update P2 to map $D^{(2)}$ approximately into the same space so that it can be roughly distinguished and then carefully adjust S for better performance on $D^{(2)}$. Therefore, we only train P2 with S fixed at this stage by minimizing the loss $l(D^{(2)}; \theta_{P2})$ using SGD with the same epochs of stage I.

Stage III: learn meta-learner M. To learn M, we use $D^{(1)}_{mem}$ as $D^{(o)}_{mem}$ and a random subset $D^{(2)}_{meta-train}$ from $D^{(2)}$ as $D^{(n)}_{meta-train}$ to minimize the Eq.(7), where l is defined as Eq.(10), θ is the parameter of unified model S. After T (which is empirically set to the same epochs of stage I) steps, we can obtain the trained M with parameters φ_T.

Stage IV: update the model S. As we get the learnt M, we will use it to update the unified model S with $D^{(1)}_{mem}$ and $D^{(2)}$. The update rules is defined in Eq.(7), where T is empirically set to the same epochs of stage I.

4. Experiments

We conduct experiments of cross-modal retrieval task on image-text datasets to verify the effectiveness of our SCML.

4.1. Datasets

Wiki [23] consists of 2,173 training and 693 testing image-text pairs. Each image is represented by the 4096-dimensional CNN descriptor vector from pre-trained AlexNet, and the 10-dimensional vector derived from a latent Dirichlet allocation (LDA) model gives the text description. Each pair is associated with one of 10 semantic labels.

MIRFLICKR [11] contains 25,000 image-text pairs. Each point associates with some of 24 labels. We remove pairs without textual tags or labels and subsequently get 18,006 pairs as the training set and 2,000 pairs as the testing set. We represent each image as a 2,048-dimensional feature extracted from the pre-trained ResNet [10]. The 1386-dimensional bag-of-words vector gives the text description.

NUS-WIDE contains 260,648 web images, and some images associate with textual tags, belonging to 81 concepts. Following [18, 13], only the top 10 most frequent labels and the corresponding 186,577 text-image pairs are kept. In our experiments, 80,000 pairs and 2,000 pairs are sampled as the training and testing sets respectively. We represent each image as a 2,048-dimensional deep feature extracted from the pre-trained ResNet [10]. The 1000- dimensional bag-of-words vector gives the text description. We sampled 5,000 pairs of the training set for training.

Table 1. The configurations of SCML. All layers are activated by tanh, 'd' denotes 'dropout', 'K' is the length of the feature.

Model	P1	P2	S
Wiki	4096-4096(d)-128	10-4096(d)-128	128(d)-128-K
Mirflickr	2048-1024(d)-128	1386-1024(d)-128	128(d)-128-K
Nuswide	2048-1024(d)-128	1000-1024(d)-128	128(d)-128-K

4.2. Evaluation protocol and Baselines

Evaluation protocol. We perform cross-modal retrieval with two tasks. (1) **Image vs. Text (I vs. T)**: retrieve relevant data in text training set using an image query. (2) **Text vs. Image (T vs. I)**: retrieve relevant data in image training set using a text query. For multi-class dataset, we consider two points are similar if they belong to the same category. We adopt the commonly-used Mean Average Precision (mAP) as the performance metric. While for the multi-label dataset, we adopt Normalized Discounted Cumulative Gain (NDCG) [12] as the performance metric.

Baselines. We firstly compare our SCML with eight cross-modal learning methods: real-value methods **TV-CCA** [8], **LCFS** [28], **JFSSL** [27], **corAE** [7], and hashing methods **CMFH** [6], **SCM** [33], **SePH** [18], **DCMH** [13]. For a fair comparison, all methods take the off-the-shelf deep features as inputs, and our SCML takes the probabilistic approach [18] to exploit alignment information after training. For deep models, we carefully implement them and replace their CNN sub-structures with the same multiple fully-connect layers network of the SCML method for pre-extracted features. Then, we validate the ability of SCML for overcoming catastrophic forgetting and compare it with two state-of-the-art continual learning methods **LwF** [17], **EWC** [15] and a cross-modal learning method **Deep-SM** [29] which needs no alignment information for training. We implement them with the same structure of SCML. Finally, we perform cross-modal learning in a parallel manner and compare it with SCML to verify the adaptability of SCML for the variation of modalities' distribution. The parameters for all baselines are set according to the original papers or experimental validations.

Implementation details. Our SCML method is implemented with Tensorflow. The detailed configurations of P1,P2 and S are illustrated in Table 1. In all experiments, the batch size is set to 64, λ to 0.1. In the first three stages, we separately use SGD (with epochs=150, lr=0.01, dropout=0.5 for WIKI, 250, 0.01, 0.6 for NUS-WIDE, 250, 0.01, 0.6 for MIRFLICKR), SGD (with lr=0.1, dropout=0.5 for WIKI, 0.1, 0.6 for NUS-WIDE, 0.1, 0.6 for MIRFLICKRr) and Adam (with lr=0.0001, dropout=0.5 for WIKI, 0.0001, 0.6 for NUS-WIDE, 0.0001, 0.6 for MIRFLICKR) for optimizing. The w_p, $\mathrm{N_{mem}}$, and the size of

Table 2. Comparison of different real-valued cross-modal learning methods on three datasets.

Method	Wiki (MAP)			MIRFLICKR (NDCG@500)			NUS-WIDE (NDCG@500)		
	I *vs.* T	T *vs.* I	Avg	I *vs.* T	T *vs.* I	Avg	I *vs.* T	T *vs.* I	Avg
corAE [7]	0.3792	0.2215	0.3004	0.4591	0.3268	0.3930	0.5148	0.5234	0.5191
TV-CCA [8]	0.2890	0.4966	0.3928	0.3033	0.3034	0.3034	0.5129	0.5050	0.5090
LCFS [28]	0.3578	0.5624	0.4601	0.3576	0.3243	0.3409	0.5725	0.5800	0.5762
JFSSL [27]	0.4253	0.6654	0.5454	0.3479	0.2971	0.3225	0.5726	0.5355	0.5540
SCML	**0.4907**	**0.6885**	**0.5896**	**0.5519**	**0.3994**	**0.4756**	**0.6854**	**0.6190**	**0.6522**

Table 3. Comparison of different cross-modal hashing methods on three datasets with different code length.

	Image *vs.* Text				Text *vs.* Image				Average			
Method	16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits
	WIKI dataset (MAP)											
SCM [33]	0.1807	0.1712	0.1698	0.1707	0.6695	**0.6911**	0.6833	0.7002	0.4251	0.4312	0.4265	0.4355
CMFH [6]	0.2053	0.2397	0.2395	0.2291	0.3299	0.3886	0.3738	0.3181	0.2676	0.3141	0.3066	0.2736
SePH [18]	0.4220	0.4507	0.4544	0.4561	0.6254	0.6384	0.6413	0.6485	0.5237	0.5445	0.5478	0.5523
DCMH [13]	0.3724	0.4366	0.4369	0.3521	0.6169	0.6610	0.6011	0.5635	0.4947	0.5488	0.5190	0.4578
SCML	**0.4705**	**0.4654**	**0.4907**	**0.4905**	**0.6702**	0.6787	**0.6885**	**0.7050**	**0.5704**	**0.5720**	**0.5896**	**0.5978**
	MIFLICKR dataset (NDCG@500)											
SCM [33]	0.3229	0.3449	0.3573	0.3628	0.2959	0.3105	0.3222	0.3256	0.3094	0.3277	0.3397	0.3442
CMFH [6]	0.2908	0.3059	0.3099	0.3162	0.2830	0.3012	0.3054	0.3054	0.2869	0.3035	0.3076	0.3108
SePH [18]	0.4216	0.4416	0.4506	0.4749	0.3089	0.3260	0.3136	0.3563	0.3652	0.3838	0.3821	0.4156
DCMH [13]	0.4064	0.4305	0.4553	0.4623	0.3132	0.3348	0.3392	0.3367	0.3598	0.3826	0.3972	0.3995
SCML	**0.4923**	**0.5178**	**0.5519**	**0.5538**	**0.3896**	**0.3979**	**0.3994**	**0.4001**	**0.4410**	**0.4578**	**0.4756**	**0.4769**
	NUS-WIDE dataset (NDCG@500)											
SCM [33]	0.5075	0.5149	0.5299	0.5308	0.4941	0.5010	0.5141	0.5143	0.5008	0.5080	0.5220	0.5226
CMFH [6]	0.4875	0.5012	0.5270	0.5394	0.4642	0.4775	0.4998	0.5091	0.4758	0.4893	0.5134	0.5242
SePH [18]	0.6157	0.6251	0.6335	0.6493	0.5275	0.5320	0.5251	0.5353	0.5716	0.5786	0.5793	0.5923
DCMH [13]	0.5757	0.6159	0.6079	0.6237	0.5756	0.5858	0.5901	0.6007	0.5756	0.6008	0.5990	0.6122
SCML	**0.6534**	**0.6741**	**0.6854**	**0.6906**	**0.5878**	**0.6078**	**0.6190**	**0.6239**	**0.6206**	**0.6410**	**0.6522**	**0.6573**

$\mathrm{D}^{(2)}_{\mathrm{meta-train}}$ are set to {20, 256, 200} for MIRFLICKR, NUS-WIDE and {1, 200, 200} for WIKI. Specially, since the gradients from multi-label loss are imbalance and the meta-learner M cannot handled it effectively, we disentangle the ∇_{θ_t} and l_t into $\{\nabla^+_{\theta_t}, \nabla^-_{\theta_t}\}$ and $\{l^+_t, l^-_t\}$ according positive and negative sample. Then these gradients and losses are processed and fed into meta-learner M for training.

4.3. Experimental Results

Comparisons with cross-modal methods. From Table 2 and Table 3, we can observe that the SCML method substantially outperforms other compared methods on all used datasets. Specifically, compared to the best shallow method JFSSL, SCML achieves boosts of 4.5%, 15.3%, and 9.8% on average on WIKI, MIRFLICKR, and NUS-WIDE datasets, respectively. Compared to the state-of-the-art hashing methods SePH/DCMH, SCML obtains the relative increase of 2.7%∼14%, 6.1%∼9.3%, and 4.9%∼7.2% on average for different bits on the three datasets, respectively. Because the original features from different modalities have a certain gap in discriminative ability, parallel learning methods (e.g., JFSSL and DCMH) will sacrifice the high discriminative of one modality and compensate another to narrow this gap. But SCML can avoid this problem by sequential learning, i.e., keeping the former's discriminative and gradually boosting the later's.

Fig. 4 shows Text vs. Image search example of compared methods. As can be seen, our SCML method tends to retrieve more relevant images than others for the query containing certain concepts, e.g., sea and animals.

Overcoming catastrophic forgetting. We first conduct experiments (in the SCML, we first train P1 and S with one modality jointly and then train P2 and S with another jointly) to confirm the *catastrophic forgetting* in sequential cross-modal learning. Fig. 5 reports the accuracy of two modalities at two stages. We observe that the performance of the first modality decreases at the second stage on all datasets, which verifies the catastrophic forgetting problem.

Next, we verify the effectiveness of meta-learner for overcoming catastrophic forgetting in our SCML. We use SGD instead of the learned optimizer M to update the unified model S at the fourth stage, which is denoted as **SCML-M**. Fig. 6 (c) and (d) report the accuracy of two modalities on WIKI dataset at different stages of SCML-M and

Table 4. The comparison of different continual learning methods on three datasets.

Method	Wiki (MAP)			MIRFLICKR (NDCG@500)			NUS-WIDE (NDCG@500)		
	I *vs.* T	T *vs.* I	Avg	I *vs.* T	T *vs.* I	Avg	I *vs.* T	T *vs.* I	Avg
Deep-SM [29]	0.3940	0.6963	0.5452	0.4983	0.4202	0.4593	0.6107	0.6003	0.6055
LwF [17]	0.2350	0.3666	0.3008	0.3379	0.3260	0.3319	0.4340	0.5259	0.4799
EWC [15]	0.2472	0.3726	0.3099	0.3916	**0.4736**	0.4326	0.4106	0.4710	0.4408
LwF+P	0.4165	0.7013	0.5589	0.5352	0.4201	0.4777	0.6254	0.6281	0.6267
EWC+P	0.4174	0.6782	0.5478	0.5310	0.4195	0.4753	0.6264	0.6210	0.6237
SCML	**0.4232**	**0.7049**	**0.5640**	**0.5403**	0.4203	**0.4803**	**0.6238**	**0.6337**	**0.6288**

Table 5. The comparison of sequential (different sequence order) and parallel manner on the three datasets.

Method	Wiki (MAP)			MIRFLICKR (NDCG@500)			NUS-WIDE (NDCG@500)		
	I *vs.* T	T *vs.* I	Avg	I *vs.* T	T *vs.* I	Avg	I *vs.* T	T *vs.* I	Avg
PCML	0.3830	0.7087	0.5458	0.4305	0.4222	0.4264	0.6148	0.5634	0.5891
$\text{SCML}_{\text{T}\rightarrow\text{I}}$	**0.4232**	0.7049	**0.5640**	0.4671	0.4200	0.4436	0.5844	0.5793	0.5818
$\text{SCML}_{\text{I}\rightarrow\text{T}}$	0.3855	**0.7126**	0.5403	**0.5491**	**0.4203**	**0.4803**	**0.6238**	**0.6337**	**0.6288**

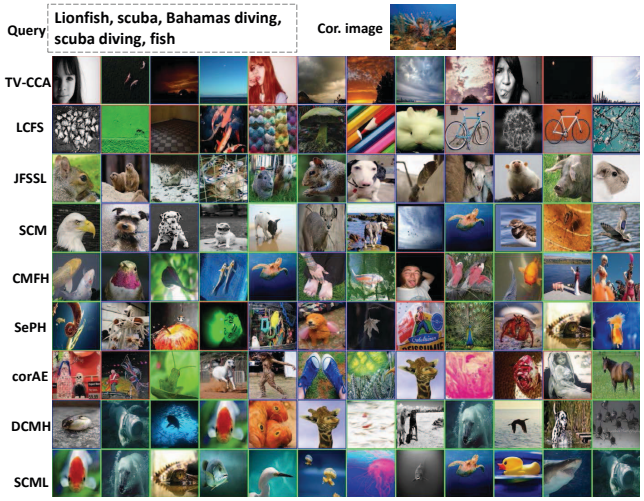
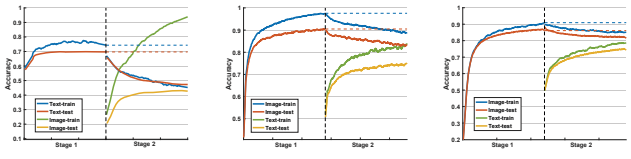


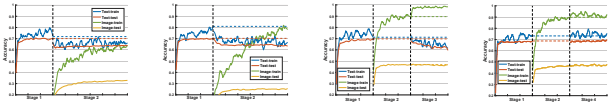
Figure 4. Retrieval examples of 'Text vs. Image' on MIRFLICKR dataset. Red border denotes irrelevant; blue denotes sharing one tag with the query, green denotes sharing two tags at least.



(a) WIKI (b) MIRFLICKR (c) NUSWIDE

Figure 5. Training accuracy of cross-modal representation learning in sequence on three datasets.

SCML (the stage III of SCML is not reported since S has no changes) respectively. We see that the accuracy of text modality decreases at the third stage of SCML-M while that at the fourth stage of SCML dost not, and the score of image modality of SCML obtains a slight increase. This result demonstrates the ability of meta-learner.

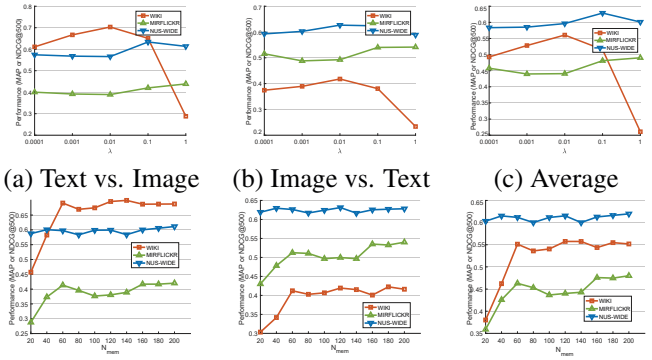


(a) EWC (b) LWF (c) SCML-M (d) SCML

Figure 6. Training accuracy of different continual learning methods on WIKI dataset.

Finally, we compare SCML with continual learning methods EWC and LWF. Table 4 reports the results. For a fair comparison, we modified EWC and LWF methods by splitting the joint training of P2 and S into two stages, and these methods are called EWC+P and LWF+P. We see that the SCML gains all-around advantages over all baselines. To further investigate the different of SCML, EWC, and LWF, we report the accuracy of two modalities on WIKI dataset at different stages in Fig. 6. At stage I, the accuracy scores of text modality of all methods reach a similar value, since the same network structure and learning policy. At stage II, the image modality's accuracy score of all methods gains rapid growth. But both EWC and LWF cannot achieve the same boost as SCML, they also have a decrease in the accuracy score of text modality, whereas SCML's keeps its performance. Therefore, the performance gap between EWC/LWF and SCML mainly comes from the limited representative of new modality and the decrease of old's performance. Besides, because EWC+P/LWF+P can prevent the limited P2's update by breaking the joint learning of P2 and S, they achieve a comparable result with SCML.

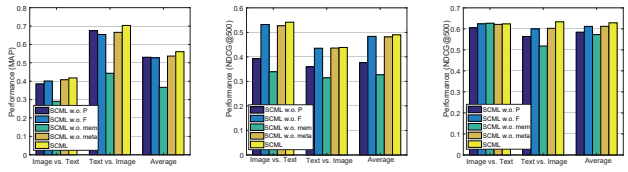
Adaptability for the variation of modalities' distribution. To validate that the sequential manner is more adaptive for the variation of modalities' distribution than the parallel manner, we perform cross-modal learning in a parallel manner as Fig. 1 (a) shows, which is called **PCML** (jointly learn P1, P2, and S with two modalities data) and compare



(a) Text vs. Image (b) Image vs. Text (c) Average

(d) Text vs. Image (e) Image vs. Text (f) Average

Figure 7. The compact of parameters λ and N_{mem} on the WIKI, MIRFKICKR, and NUS-WIDE datasets.



@WIKI @MIFLICKR @NUS-WIDE

Figure 8. Evaluations (mAP and NDCG) of the proposed SCML with ablating different components.

it with SCML. Table 5 reports the result. We observe that SCML outperforms PCML on all used datasets since PCML is prone to suffer from the inconsistent of modalities' distribution (e.g., the discriminative of one modality cannot be improved). This result demonstrates the adaptability of sequential learning manner for modalities' distribution.

4.4. Empirical Analysis

Impact of Modality Sequence To investigate the influence of different sequences of modality, we train the SCML method on two modality sequences: 'Text-Image' (training SCML firstly on text modality and then on image modality) and vice versa, these two models are called $\textbf{SCML}_{\mathrm{T}\to\mathrm{I}}$ and $\textbf{SCML}_{\mathrm{I}\to\mathrm{T}}$. Table 5 shows the result. We find that their average performances on three datasets are different. Specifically, $\mathrm{SCML}_{\mathrm{T}\to\mathrm{I}}$ outperforms $\mathrm{SCML}_{\mathrm{I}\to\mathrm{T}}$ on WIKI dataset, whereas $\mathrm{SCML}_{\mathrm{I}\to\mathrm{T}}$ performs better than $\mathrm{SCML}_{\mathrm{T}\to\mathrm{I}}$ on MIRFLICKR and NUS-WIDE datasets. Indeed, the discriminative of text feature is more powerful than that of image feature on WIKI dataset, which is exactly reversed on MIRFLICKR and NUS-WIDE datasets. This result implies that feeding the more discriminative modality data to training SCML in the early stage is more useful for the learning of later stages.

Sensitivity to Parameters We analyze the effect of balance weight λ and the memory size $\mathrm{N_{mem}}$ of $\mathrm{D^{(1)}_{memory}}$. We initially set $\{\lambda, \mathrm{N_{mem}}\}$ to $\{0.01, 200\}$ for WIKI, $\{0.1, 256\}$ for MIRFLICR and NUS-WIDE. Then, we separately tune them with other parameters fixing and report the cross-modal retrieval performance in Fig. 7

From Fig. 7(a)-(c), we see that the SCML method achieves the best performance at a certain value, since a smaller λ may lead to dramatic changes of model and cause knowledge forgetting, while a larger λ may encourage the less change of model and depress the learning of new modality data. From Fig. 7(d)-(f) we find that the retrieval performance increases firstly and then fluctuates within a certain range with an increase of N_{mem}. This result indicates that an appropriate size of memories can help to learn cross-modal representation better, while a larger N_{mem} may be useless.

Ablation Study To analyze the effectiveness of different stages and components in the proposed SCML method, We separately remove: 'the second stage', 'the third and fourth stages', 'the memory $D^{(1)}_{memory}$' with others remained to evaluate their influence on the final performance. These three models are called **SCML w.o. P**, **SCML w.o.F**, and **SCML w.o. mem**. We also replace M with SGD for fine-tuning the unified model S at stage 4 in SCML to investigate the M's advantage, which is denoted as **SCML w.o. meta**. Fig.8 shows the result of two tasks.

We can see that separately removing the training stages, and memory will damage the retrieval performance of SCML method to varying degrees, e.g., the performance of SCML w.o. meta is inferior to SCML, which re-confirms that the ability of meta-learner in SCML for dealing with the catastrophic forgetting problem. The result also indicates that each stage and components of SCML are essential for sequential cross-modal representation learning and have separated contributions to the final performance.

5. Conclusion

In this paper, we have presented a novel cross-modal representation learning method, name SCML, for retrieval task. Unlike previous methods that design multiple sub-models for each modality and joint learn them with aligned multi-modality data, our method conforms to the human's cognitive mechanism, and it only includes one unified model to be sequentially trained on different modalities to map them into the common feature space. Particularly, to overcome the catastrophic forgetting in sequential learning, we propose to learn an optimizer to guide the update of the unified model. Our experimental results demonstrate that the proposed method can sequentially perform cross-modal learning and achieves state-of-the-art retrieval performance on three popular datasets.

Acknowledgements This work is partially supported by National Science Foundation of China (61976115,61672280, 61732006), AI+ Project of NUAA(56XZA18009), Jiangsu Innovation Program for Graduate Education (KYCX18_0307).

References

[1] Rahaf Aljundi, Francesca Babiloni, Mohamed Elhoseiny, Marcus Rohrbach, and Tinne Tuytelaars. Memory aware synapses: Learning what (not) to forget. In *ECCV*, pages 144–161, 2018.

[2] Galen Andrew, Raman Arora, Jeff A. Bilmes, and Karen Livescu. Deep canonical correlation analysis. In *ICML*, pages 1247–1255, 2013.

[3] Marcin Andrychowicz, Misha Denil, Sergio Gomez Colmenarejo, Matthew W. Hoffman, David Pfau, Tom Schaul, and Nando de Freitas. Learning to learn by gradient descent by gradient descent. In *NeurIPS*, pages 3981–3989, 2016.

[4] Tadas Baltrusaitis, Chaitanya Ahuja, and Louis-Philippe Morency. Multimodal machine learning: A survey and taxonomy. *IEEE Trans. Pattern Anal. Mach. Intell.*, 41(2):423–443, 2019.

[5] Yue Cao, Mingsheng Long, Jianmin Wang, Qiang Yang, and Philip S. Yu. Deep visual-semantic hashing for cross-modal retrieval. In *SIGKDD*, pages 1445–1454, 2016.

[6] Guiguang Ding, Yuchen Guo, and Jile Zhou. Collective matrix factorization hashing for multimodal data. In *CVPR*, pages 2083–2090, 2014.

[7] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. Cross-modal retrieval with correspondence autoencoder. In *ACM MM*, pages 7–16, 2014.

[8] Yunchao Gong, Qifa Ke, Michael Isard, and Svetlana Lazebnik. A multi-view embedding space for modeling internet images, tags, and their semantics. *IJCV*, 106(2):210–233, 2014.

[9] Jiuxiang Gu, Jianfei Cai, Shafiq R. Joty, Li Niu, and Gang Wang. Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models. In *CVPR*, pages 7181–7189, 2018.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[11] Mark J. Huiskes and Michael S. Lew. The MIR flickr retrieval evaluation. In *ACM SIGMM MIR*, pages 39–43, 2008.

[12] Kalervo Järvelin and Jaana Kekäläinen. Ir evaluation methods for retrieving highly relevant documents. In *International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 41–48, 2000.

[13] Qing-Yuan Jiang and Wu-Jun Li. Deep cross-modal hashing. In *CVPR*, pages 3270–3278, 2017.

[14] Ferenc Kemény and Beat Meier. Multimodal sequence learning. *Acta psychologica*, 164:27–33, 2016.

[15] James Kirkpatrick, Razvan Pascanu, Neil C. Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *CoRR*, abs/1612.00796, 2016.

[16] Chao Li, Cheng Deng, Ning Li, Wei Liu, Xinbo Gao, and Dacheng Tao. Self-supervised adversarial hashing networks for cross-modal retrieval. In *CVPR*, pages 4242–4251, 2018.

[17] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 40(12):2935–2947, 2018.

[18] Zijia Lin, Guiguang Ding, Mingqing Hu, and Jianmin Wang. Semantics-preserving hashing for cross-view retrieval. In *CVPR*, pages 3864–3872, 2015.

[19] Hong Liu, Mingbao Lin, Shengchuan Zhang, Yongjian Wu, Feiyue Huang, and Rongrong Ji. Dense auto-encoder hashing for robust cross-modality retrieval. In *ACM MM*, pages 1589–1597, 2018.

[20] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *NeurIPS*, pages 6470–6479, 2017.

[21] Niluthpol Chowdhury Mithun, Rameswar Panda, Evangelos E. Papalexakis, and Amit K. Roy-Chowdhury. Webly supervised joint embedding for cross-modal image-text retrieval. In *ACM MM*, pages 1856–1864, 2018.

[22] Cuong V. Nguyen, Yingzhen Li, Thang D. Bui, and Richard E. Turner. Variational continual learning. 2018.

[23] Nikhil Rasiwasia, Jose Costa Pereira, Emanuele Coviello, Gabriel Doyle, Gert R. G. Lanckriet, Roger Levy, and Nuno Vasconcelos. A new approach to cross-modal multimedia retrieval. In *ACM MM*, pages 251–260, 2010.

[24] Andrei A. Rusu, Neil C. Rabinowitz, Guillaume Desjardins, Hubert Soyer, James Kirkpatrick, Koray Kavukcuoglu, Razvan Pascanu, and Raia Hadsell. Progressive neural networks. *CoRR*, abs/1606.04671, 2016.

[25] Joan Serrà, Didac Suris, Marius Miron, and Alexandros Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. In *ICML*, pages 4555–4564, 2018.

[26] Yuming Shen, Li Liu, Fumin Shen, and Ling Shao. Zero-shot sketch-image hashing. In *CVPR*, pages 3598–3607, 2018.

[27] Kaiye Wang, Ran He, Liang Wang, Wei Wang, and Tieniu Tan. Joint feature selection and subspace learning for cross-modal retrieval. *IEEE Trans. Pattern Anal. Mach. Intell.*, 38(10):2010–2023, 2016.

[28] Kaiye Wang, Ran He, Wei Wang, Liang Wang, and Tieniu Tan. Learning coupled feature spaces for cross-modal matching. In *ICCV*, pages 2088–2095, 2013.

[29] Yunchao Wei, Yao Zhao, Canyi Lu, Shikui Wei, Luoqi Liu, Zhenfeng Zhu, and Shuicheng Yan. Cross-modal retrieval with CNN visual features: A new baseline. *IEEE Trans. Cybernetics*, 47(2):449–460, 2017.

[30] Yiling Wu, Shuhui Wang, and Qingming Huang. Learning semantic structure-preserved embeddings for cross-modal retrieval. In *ACM MM*, pages 825–833, 2018.

[31] Zhaoda Ye and Yuxin Peng. Multi-scale correlation for sequential cross-modal hashing learning. In *ACM MM*, pages 852–860, 2018.

[32] Jaehong Yoon, Eunho Yang, Jeongtae Lee, and Sung Ju Hwang. Lifelong learning with dynamically expandable networks. In *ICLR*, 2018.

[33] Dongqing Zhang and Wu-Jun Li. Large-scale supervised multimodal hashing with semantic correlation maximization. In *AAAI*, pages 2177–2183, 2014.