Real-world Cross-modal Retrieval via Sequential Learning

Ge Song and Xiaoyang Tan

***Abstract*—Cross-modal retrieval is playing an increasingly important role in our daily life with the explosive growth of multimedia data. However, its learning paradigm under real-life environments is less studied, and most existing approaches are developed in the pre-desired settings (e.g., unchanging modalities and explicitly modal-aligned samples). Inspired by the recent achievement in the field of cognition mechanism on how the human brain acquires knowledge, we present a new sequential learning method for real-world cross-modal retrieval. In this method, a unified model is maintained to capture the common knowledge of various modalities but are learned 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 reformulate the objective of optimization-based meta-learning and propose a novel meta-learning method to overcome the catastrophic forgetting encountered in sequential learning. Extensive experiments are conducted on four popular image-text multimodal datasets and a five-modal dataset, showing that our method achieves state-of-the-art cross-modal retrieval performance without explicit modal-alignment.**

***Index Terms*—Cross-modal Retrieval, Sequential Learning, Deep Learning, Meta Learning.**

I. Introduction

Cross-modal retrieval, aiming to search relevant instances from one modality in response to a query of another modality, has been an active research area for the past few years due to its wide usage in real-world applications, e.g., sketch-based image retrieval in the criminal investigation and product search. The difficulty of the measurement of content similarity among data from different modalities, which is known as the heterogeneity gap [1], makes this task very challenging. Thus, bridging the heterogeneity gap between different modalities plays a key role in cross-modal retrieval.

A typical approach to bridge the heterogeneity gap is cross-modal representation learning. It tries to find functions to map the data samples from different modalities into a shared feature space, such that their similarity becomes computationally. Many methods [2], [3] have been developed to find the common space in different learning ways. Hashing-based methods [4], [5] also extends to cross-modality retrieval via embedding the data of interest into low-dimensional binary codes, bringing the lower storage costs and the higher computational efficiency. We observe that these methods are built in the same manner: developing individual sub-models for each modality and jointly learning them by explicit aligned multi-modal data (e.g., text-image pairs). 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 could not be applied in a real-world environment due to the following reasons. 1) The explicit aligned multi-modal data are expensive to obtain and thus limit the amount of training data available. 2) It is unlikely to be adapted without retraining the whole system under the new environment when the underlying distribution of different modalities is gradually changing. 3) The space and training costs of the system will increase with modalities become more.

To find a more practical learning paradigm, we investigate the biological mechanism of cross-modal learning. Recent work [6] 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

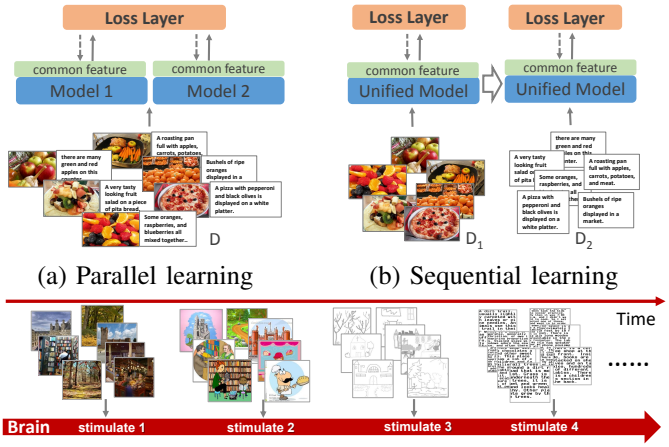


Fig. 1. Illustration of the difference between two cross-modal learning paradigms: parallel and sequential. In the parallel paradigm, the whole architecture involves multiple individual sub-models with each responsible for one modality, and well-aligned multi-modal data (e.g., image-text pairs) are needed 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. In the real-world learning paradigm, our brain receives the stimulation modality-by-modality and can interpret them into the common concepts implicitly. We observe that the sequential paradigm behaves more similar to the real-world learning paradigm than the parallel paradigm.

This work is partially supported by National Science Foundation of China (61976115,61672280, 61732006), AI+ Project of NUAA(56XZA18009), research project no. 6140312020413, Jiangsu Innovation Program for Graduate Education (KYCX18_0307), China Scholarship Council (201906830057). (Corresponding author: Xiaoyang Tan.)

G. Song and X. Tan are with the College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China, also with MIIT Key Laboratory of Pattern Analysis and Machine Intelligence, and the Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing 211106, China (e-mail: sunge@nuaa.edu.cn; x.tan@nuaa.edu.cn;).

Manuscript received Oct 5, 2019; revised Oct 5, 2019.

one unitary representation. In other words, our brain acquires knowledge or concepts implicitly across different modalities in a sequential learning manner (i.e., modality-by-modality), as Fig 1 (c) shows. In contrast with PCML, this sequential manner is more practical in real-life scenario: 1) it is easier for us to learn a stable conceptual representations from one modality (e.g., object recognition of images) 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; 4) sequential accommodating new modality makes it possible for our brain to handle more modalities. Inspired by the 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, it is not trivial to learn the representation of different modalities in one model sequentially. Training a model with new information could interfere with the previously acquired knowledge, which is often referred to as *catastrophic forgetting* [7]–[10]. 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 [7], [8] and the appropriate optimization for those parameters is of importance. But most of the existing methods tackle this problem by imposing different constraints on updating instead of seeking suitable changes. Please note that the suitable changes may not be the opposite direction of gradients but can be learned in an automatic way. In the meta-learning community, an optimization-based meta-learner [11] can be trained with limited samples on massive same single old tasks for effectively and fast optimizing a new model of a new task, or this meta-learner can learn to output effective changes for a new model. 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 past tasks. Motivated by this, we propose a novel LSTM-based meta-learner to address the catastrophic forgetting issue in sequential cross-modal learning. This meta-learner is first trained with limited samples of old and new modalities when new modality data is available, and then it is adopted to optimize the unified model with new modality data. The main contributions of this paper are summarized as follows:

- We present a novel sequential cross-modal learning method (SCML), which is consistent with the cognitive mechanism of human beings in acquiring knowledge across multi-modalities. In contrast with previous approaches, SCML is more adaptive to the evolving distributions of different modalities, and it does not need the laborious explicit 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 detail, we redefined the objective of the traditional LSTM-based meta-learner and revised its structure to absorb the feedback from limited samples of old and new modalities, making it able to effectively optimize the old model on the new task and maintain previous knowledge.
- Extensive experimental results demonstrate that the proposed SCML method is resistant to catastrophic forgetting. It can perform cross-modal learning well in a sequential manner and yield state-of-the-art retrieval performance on four image-text cross-modality datasets and a five-modalities dataset.

II. Related Work

Cross-modal retrieval. Cross-modal learning approaches [2], [4], [12]–[15] can roughly be divided into real-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. DeepCCA [16] learns two separate deep neural networks for two corresponding modalities so that they are maximally correlated in the common subspace. Wu et al. [17] propose a semantic structure-preserved embedding learning method based on the semantic structure and local geometric structure consistency. Some methods [2], [18], [19] also utilize attention mechanism to matching image regions and words for fine-level text-image retrieval. The hashing methods [4], [14], [15], [20] seek to encode high-dimensional features into compact binary codes, hence enabling fast similarity search with Hamming distances. Li et al. [15] propose a self-supervised adversarial hashing (SSAH) approach, which attempts to incorporate adversarial learning into the cross-modal hashing. Cao et al. [12] generate compact hash codes of images and sentences using stacked LSTMs and CNN. Whereas the sketch-image hashing by Shen et al. [21] consists of two convolutional neural networks to encode ketches and natural images. 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. Song et al. [22] propose a memory neural network model for representing unaligned cross-modal data. But it needs to retrain the whole system with all multi-modal data when new modality is available, which limits its usage.

Sequential learning. The sequential learning also can be called 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 computational models regarding lifelong learning is that they are prone to catastrophic forgetting. Massive methods [7], [23], [24] 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. [7] propose a model called elastic weight consolidation (EWC), where a quadratic penalty is used to slow down the learning for task-relevant

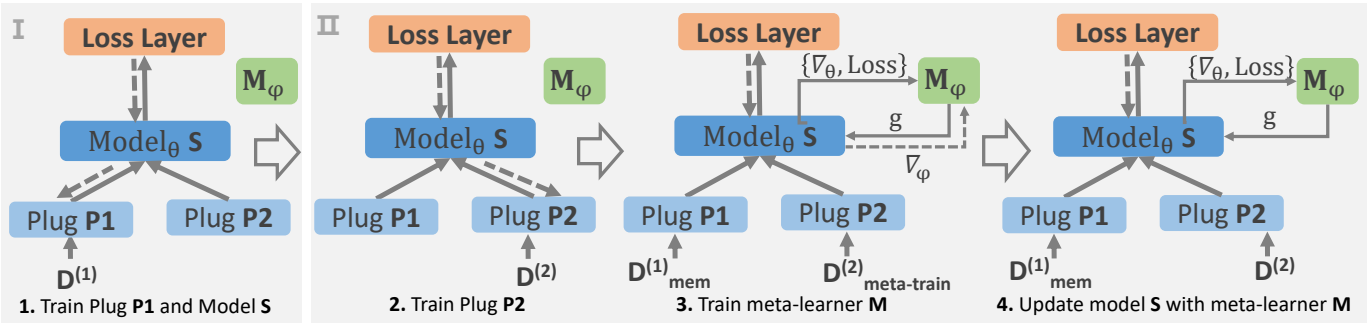


Fig. 2. Overview of sequential cross-modal learning on two modalities. The architecture consists of three components (plugs P1, P2, a unified model S, and a meta-learner M) and two stages (four steps): The model firstly takes one modality $\mathrm{D}^{(1)}$ to jointly learn P1 and S. When the new modality $\mathrm{D}^{(2)}$ 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 $\mathrm{D}^{(1)}_{\mathrm{mem}}$ 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 $\mathrm{D}^{(1)}_{\mathrm{mem}}$ and $\mathrm{D}^{(2)}$.

weights coding for previously learned knowledge, whereas Riemannian Walk (RWalk) [25] adopts an online manner to estimate the importance of weights. Zenke et al. [24] propose to alleviate catastrophic forgetting by allowing individual synapses to estimate their importance for solving a learned task. [9] develops a variational continual learning framework based on the argument that previous data have told us about the model parameters (the previous posterior), and the current data is telling us (the likelihood). The second is dynamic architectures [10], [26], 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. Rusu et al. [26] propose a progressive network that can expand the architecture by allocating novel sub-networks with a fixed capacity to be trained with the new information. The last is the memory-based method, [8] proposes a model called Gradient Episodic Memory (GEM) to alleviate forgetting, which uses a set of previous tasks data to constraint optimizing. GEM is upgraded to A-GEM [27] by slacking the constraints. These methods are designed for the single modality problem, and they have not been verified on large-scale cross-modal datasets.

III. The Proposed Method

In this section, we first give the problem definition of SCML and its model structure. Then, we detail the meta-learner component and show how to use it to perform SCML.

A. 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 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 $\mathrm{D}^{(2)} = \{\mathrm{x}_\mathrm{i}^{(2)}, \mathrm{y}_\mathrm{i}^{(2)}\}_{\mathrm{i}=1}^{\mathrm{N}_2}$, where $\mathrm{x}_\mathrm{i}^{(2)} \in \mathrm{R}^{\mathrm{d}_2}$ 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.

B. The structure of the model

According to the above problem definition, our SCML model consists of three components for sequential bi-modality learning (see Fig. 2): plugs P1, P2, a unified model S, and meta-learner M.

Plugs: Plugs are designed respectively for mapping original features (e.g., CNN or hand-crafted) of different modalities into the same dimension. Each modality has an independent plug. For flexible expansion, we implement them as 3-layers deep neural networks.

Unified Model: The unified model S is a general model that 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-connected neural network. To speed up retrieval, we quantize the output h of the last layer by simple quantization $b = sign(h)$ to obtain final binary common representation.

$$sign(x) = \begin{cases} 0, \, if x \leq 0 \\ 1, \, otherwise \end{cases} \tag{1}$$

Meta-Learner: The meta-learner is designed to learn to optimize the unified model when a new modality is available. Because the *catastrophic forgetting* is triggered by the inappropriate changes of weights, which are caused by the optimizer, we propose to learn a meta-learner for updating the unified model more appropriately. Please see Sec.III-C for details.

C. 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 learner f, the loss function $\mathcal{L}$ for meta-learner M can be defined as follows:

$$\mathcal{L} = \sum_{i=1}^{N_T} l\big(D^{(i)}, M(D^{(i)}, f^{(i)}; \varphi)\big) \tag{2}$$

The optimal φ_* can be obtained via minimizing Eq. (2). 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 $f_s^{(o)}$ (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:

$$\begin{aligned} \mathcal{L} &= l(D^{(n)}, M(D^{(n)}, f_s^{(o)}; \varphi)) \\ s.t.\ l(D^{(o)}, M(D^{(n)}, f_s^{(o)}; \varphi)) &\le l(D^{(o)}, f_s^{(o)}) \end{aligned} \tag{3}$$

We rephrase the constraint term as the task of better performance on $T^{(o)}$, then Eq.(3) is rewritten as follows:

$$\mathcal{L} = l(D^{(n)}, M(D^{(n)}, f_s^{(o)})) + l(D^{(o)}, M(D^{(n)}, f_s^{(o)})) \tag{4}$$

The first term encourages the M to update $f_s^{(o)}$ for better performance on the new task, while the second term imposes M to update $f_s^{(o)}$ for less forgetting old task knowledge. However, two facts make the optimization of Eq.(4) be impossible: 1) the whole datasets $D^{(o)}$ 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.(4) is rewritten as follows:

$$\begin{aligned} \mathcal{L} = &\, l\big(D_{meta-train}^{(n)}, M(D_{mem}^{(o)}, D_{meta-train}^{(n)}, f_s^{(o)}; \varphi)\big) \\ &+ l\big(D_{mem}^{(o)}, M(D_{mem}^{(o)}, D_{meta-train}^{(n)}, f_s^{(o)}; \varphi)\big) \end{aligned} \tag{5}$$

The optimal φ_* can be obtained via minimizing Eq. (5). 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:

$$f_s^{(n)} = M(D_{mem}^{(o)}, D^{(n)}, f_s^{(o)}; \varphi_*) \tag{6}$$

Implementation. We adopt the LSTM-based meta-learner (in [11], 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:

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

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.(5), 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 change little. Thus, the loss function of the modified M is written as follows:

$$\begin{aligned} \mathcal{L} &= \sum_{t=1}^{T} \big(l_t^{(n)}(D_t^{(n)}, \theta_t) + l_t^{(o)}(D_t^{(o)}, \theta_t) + \lambda |g_t|_1\big) \\ \theta_{t+1} &= \theta_t + g_t \\ [g_t, h_{t+1}] &= M([\nabla_{\theta_t}^{(o)}, \nabla_{\theta_t}^{(n)}, l_t^{(o)}, l_t^{(n)}], h_t; \varphi) \end{aligned} \tag{8}$$

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.(8) have very different magnitudes so that M cannot work stably. So we preprocess M's inputs by the following formula:

$$x \to \begin{cases} \big(\dfrac{log(|x|)}{p}, sgn(x)\big) & , if |x| \ge e^{-p} \\ (-1, e^p x) & , otherwise \end{cases} \tag{9}$$

where $p > 0$ is a parameter controlling small disregard values, and we set it to 10 in all experiments according to [11].

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, where i is the index of parameters.

Optimization. To minimize the Eq.(8) by Backpropagation Through Time (BPTT), we need to unroll the LSTM meta-learner in T steps and store g_t, θ_t, $\nabla_{\theta_t}^{(o,n)}$ at each time-step (equivalent to four times the number of the parameters of the learner f at least), and T depends on the number of training epochs of learner f, the number of training data D, and the batch size. In practice, both T and the number of θ are larger than thousands or even ten thousand, and we could not unroll the LSTM meta-learner in such large T steps. Therefore we minimize the Eq.(8) in a step-by-step way instead of optimizing the entire trajectory by 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:

$$\begin{aligned} \mathcal{L}_t &= l(D_t^{(n)}, \theta_{t+1}) + l(D_t^{(o)}, \theta_{t+1}) + \lambda |g_t|_1 \\ \theta_{t+1} &= \theta_t + g_t \\ [g_t, h_{t+1}] &= M([\nabla_{\theta_t}^{(o)}, \nabla_{\theta_t}^{(n)}, l_t^{(o)}, l_t^{(n)}], h_t; \varphi_t) \end{aligned} \tag{10}$$

where $l_t^{(o)} = l(D_t^{(o)}, \theta_t)$, $l_t^{(n)} = l(D_t^{(n)}, \theta_t)$, $\nabla_{\theta_t}^{(o)} = \frac{\partial l_t^{(o)}}{\partial \theta_t}$, $\nabla_{\partial \theta_t}^{(n)} = \frac{l_t^{(n)}}{\partial \theta_t}$. h_1 is initialized with zero vector. The update of φ can be roughly denoted as $\varphi_{t+1} = \varphi_t - \frac{\partial \mathcal{L}_t}{\partial \varphi_t}$.

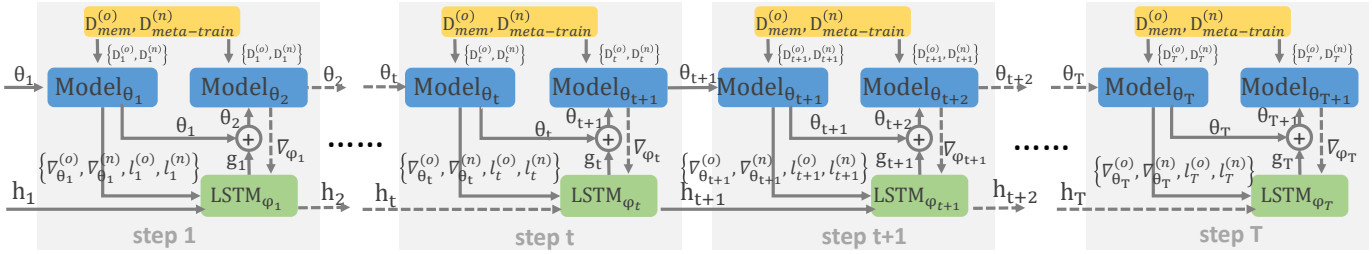


Fig. 3. The unrolled time-step of training LSTM meta-learner. We minimize the loss at each time-step.

D. Sequential cross-modal learning

Based on the above model, we perform sequential cross-modal learning in two stages, (stage I contains one step and stage II contains three steps), which is shown in Fig. 2.

Stage I: learn model for the first modality. For the first modality $D^{(1)}$, we train the plug P1 and the unified model S jointly. 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:

$$l(D^{(1)};\theta) = -\sum_{i=1}^{N_1}\sum_{j=1}^{C}\Big((w_p \cdot y_{ij}\log(\hat{y_{ij}}) + (1-y_{ij})\log(1-\hat{y_{ij}})\Big) \tag{11}$$

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 get the optimal parameter $\theta_*^{(1)}$ of S, and 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 model for the new modality. We perform new modality learning in three steps.

1) 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 a gradient descent algorithm (e.g., SGD) to optimize P2 and S jointly, which is observed and discussed in Sec. IV-C. 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 step by minimizing the loss $l(D^{(2)};\theta_{P2})$ using SGD with the same epochs of stage I.

2) Learn meta-learner M. In this step, we train meta-learner M to learn to update the unified model S. We use $D_{mem}^{(1)}$ as $D_{mem}^{(o)}$ and a random subset $D_{meta-train}^{(2)}$ from $D^{(2)}$ as $D_{meta-train}^{(n)}$ to minimize the Eq.(10), where $l(D,\theta)$ is defined as Eq.(11), θ is the parameter of unified model S, $\theta_1 = \theta_*^{(1)}$, h_1 is zero vector. After T (which is empirically set to the same epochs of stage I) steps, we can obtain the trained M with parameters φ_*.

3) Update the model S. As we get the learned M, we will use it to update the unified model S with $D_{mem}^{(1)}$ and $D^{(2)}$. The update rules are defined as follows.

$$\begin{aligned} \theta_{t+1} &= \theta_t + g_t,\ t = 1,\cdots,T-1 \\ [g_t, h_{t+1}] &= M([\nabla_{\theta_t}^{(o)}, \nabla_{\theta_t}^{(n)}, l_t^{(o)}, l_t^{(n)}], h_t; \varphi_*) \end{aligned} \tag{12}$$

the optimal parameter θ_* of S is θ_T, $\theta_1 = \theta_*^{(1)}$, the step number T is empirically set to the same steps of stage I.

E. Extension for more than two modalities

The proposed SCML framework is able to handle cases that contain more than two modalities. For example, when the third modality comes, we just need to repeat the stage II to accommodate new modality, i.e., 1) learn plug of new modality, 2) learn meta-learner M with $D_{mem}^{(o)} = \{D_{mem}^{(1)}, D_{mem}^{(2)}\}$ and a subset of $D^{(3)}$, 3) update the model S with $D^{(3)}$ and the learnt meta-learner M.

IV. Experiments

We conduct experiments of cross-modal retrieval tasks on four image-text datasets and a five-modalities dataset to verify the effectiveness of our SCML

A. Datasets

Wiki [28] 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 [29] 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 [30]. The 1386-dimensional bag-of-words vector gives the text description.

NUS-WIDE contains 260,648 web images, and some images associated with textual tags, belonging to 81 concepts. Following [5], [31], 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 [30]. The 1000- dimensional bag-of-words vector gives the text description. We sampled 5,000 pairs from the training set for training.

COCO is a large-scale object dataset, containing 82,783 training and 40,504 testing images. Each image is associated with five sentences (only the first sentence is used in our experiments), belonging to 80 most frequent categories. After pruning images with no category information, we obtained 82,081 image-sentence pairs as the training set. We represent each image as a 2,048-dimensional deep feature extracted from the ResNet [30] network pre-trained on the ImageNet. The 4800-dimensional Skip-thought vector [32] gives the sentence description. We sample 10,000 pairs from the training set for training and 4,956 pairs from the testing set as queries.

CMPlaces is a large-scale places dataset that consists of five modalities. It includes 2.4 million training and 20,500 testing natural images (NAT), 14,830 training and 2,050 testing line drawings (LDR), 9,752 training and 2050 testing textual descriptions (DSC), 11,372 training and 1,954 testing clip art (CLP), 456,300 training and 2,050 testing synthetic Spatial text images (SPT). Each sample associated with a unique label of 205 scene categories. The average-pooling the 4800-D Skip-thought vectors [32] of each sentence give the DSC description. For pixel-based modalities (e.g., NAT, LDR, etc.), we separately fine-tuned the AlexNet (pre-trained on the Place 205 dataset [33]) on the corresponding training data and extract the 4,096-D feature from the `fc7` layer as the representation. We sample 38,950 examples of NAT and 28,149 examples of SPT from corresponding training sets, and we then mix them with all training examples of LDR, CLP, DSC modalities for training and being retrieved. We sample 2,050 examples from the testing set (if the size of the testing set is larger than 2,050) of each modality as queries.

B. Experimental Settings

1) Evaluation protocol: We perform two cross-modal retrieval tasks. (1) **Image to Text (I→T)**: retrieve relevant data in the text training set using an image query. (2)**Text to Image (T→I)**: retrieve relevant data in the image training set using a text query. We adopt the commonly-used Mean Average Precision (MAP) and the Normalized Discounted Cumulative Gain (NDCG) [34] as performance metrics. We consider two points are similar if they belong to the same category or share one tag at least. Two metrics are defined as follows:

$$\begin{aligned} \mathrm{P@n} &= \frac{\#\left\{\text{relevant images in top N results}\right\}}{\mathrm{N}} \\ \mathrm{AP} &= \frac{\sum_{\mathrm{n}} \mathrm{P@n} \times \mathrm{I}\left\{\text{image n is relevant}\right\}}{\#\left\{\text{retrieved relevant image}\right\}} \\ \mathrm{mAP} &= \frac{1}{\mathrm{Q}} \sum_{\mathrm{i}} \mathrm{AP_i} \end{aligned} \tag{13}$$

where # is a count function, I is an indicator function, and Q represents the total number of queries. Notably, the mAP is computed from the retrieval list over the whole database set, while $\mathrm{mAP@n}$ is calculated from the top n retrieval results.

TABLE I
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
NUS-WIDE	2048-1024(d)-128	1000-1024(d)-128	128(d)-128-K
COCO	2048-2048(d)-128	4800-2048(d)-128	128(d)-128-K
	For All P		S
CMPlace	feature dim.-4096(d)-128		128(d)-128-K

$$NDCG@p = \frac{1}{Z} \sum_{i=1}^{p} \frac{2^{r_i} - 1}{\log(1+i)} \tag{14}$$

where Z is the ideal $DCG@p$ and calculated form the correct ranking list. $r(i) = |l_q \cap l_i|$ denotes the similarity between the i-th point and the query. l_q and l_i denote the label set of the query and i-th position point.

2) Baselines: We firstly validate the ability of SCML for overcoming catastrophic forgetting and compare it with four state-of-the-art continual learning methods **LwF** [35], **EWC** [7], **RWalk** [25], **A-GEM** [27], a cross-modal learning method **Deep-SM** [36] which needs no alignment information for training, and a incremental deep hashing method **DIHN** [37], which is originally proposed for open-set image retrieval problem. We modify DIHN by associating each task to a hash learning task of one modality. We also perform cross-modal learning in a parallel manner, comparing it with SCML to verify the adaptability of SCML for the variation of modalities' distribution. We implement all methods with the same structure as the SCML.

Then, we compare our SCML with fifteen cross-modal learning methods: eight real-value methods including unsupervised **corAE** [38], **CDPAE** [39] and supervised **TV-CCA** [40], **LCFS** [41], **JFSSL** [42], **VSE++** [43], **DAML** [44], **SSPE** [17], seven cross-modal hashing methods including unsupervised **CMFH** [45], **UCH** [46] and supervised **SCM** [47], **SePH** [5], **DCMH** [31], **TDH** [48], **SSAH** [15]. For a fair comparison, all methods take the deep off-the-shelf features as inputs, and our SCML takes the probabilistic approach [5] to exploit alignment information after training. For deep models, we carefully implement them and replace their CNN sub-structures with the same multiple fully-connected layers network of the SCML method for pre-extracted features.

Finally, we investigate the influence of modality sequence, parameters, and critical components and stages in the SCML method. We also perform comparison and validation experiments on the CMPlace dataset to test the ability of SCML for the long sequential cross-modal learning. We set the parameters of all baselines according to the original papers or experimental validations.

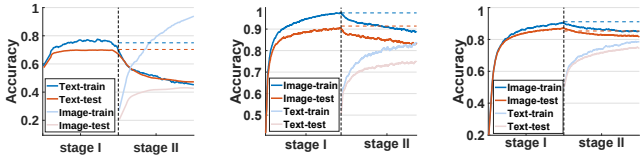
3) Implementation details: Our SCML method is implemented with Tensorflow. The detailed configurations of P1,P2 and S are illustrated in Table I. In all experiments, the batch size is set to 64, λ to 0.1. At the stage I, we use SGD (setting epochs=150, lr=0.01, dropout=0.5 for WIKI, 250, 0.01, 0.6

TABLE II
THE COMPARISON OF DIFFERENT CONTINUAL LEARNING METHODS ON WIKI, MIRFLICKR, NUS-WIED, AND COCO DATASETS.

Method	WIKI (MAP)			MIRFLICKR (NDCG@500)			NUS-WIDE (NDCG@500)			COCO (NDCG@500)		
	I→T	T→I	Avg	I→T	T→I	Avg	I→T	T→I	Avg	I→T	T→I	Avg
DeepSM [36]	0.3940	0.6963	0.5452	0.4983	0.4202	0.4593	0.6107	0.6003	0.6055	0.2591	0.1992	0.2291
DIHN [37]	0.3760	0.7001	0.5380	0.4073	0.3747	0.3910	0.5947	0.5134	0.5540	0.2012	0.1660	0.1836
LwF [35]	0.2350	0.3666	0.3008	0.3379	0.3260	0.3319	0.4340	0.5259	0.4799	0.0962	0.1213	0.1087
EWC [7]	0.2472	0.3726	0.3099	0.3916	0.4736	0.4326	0.4106	0.4710	0.4408	0.2498	0.2138	0.2318
RWalk [25]	0.3104	0.3078	0.3091	0.3789	**0.5054**	0.4421	0.5523	0.5482	0.5503	0.2509	0.2173	0.2341
A-GEM [27]	0.3356	0.3938	0.3647	0.4382	0.3837	0.4109	0.5706	0.5708	0.5707	0.2773	0.2191	0.2482
LwF+P	0.4165	0.7013	0.5589	0.5352	0.4201	0.4777	0.6254	0.6281	0.6267	0.2883	0.2217	0.2550
EWC+P	0.4174	0.6782	0.5478	0.5310	0.4195	0.4753	0.6264	0.6210	0.6237	0.2810	0.2170	0.2490
RWalk+P	0.4197	0.6958	0.5577	0.5342	0.4260	0.4801	0.6249	0.6244	0.6246	0.2885	0.2248	0.2566
A-GEM+P	0.4185	0.6975	0.5580	0.5361	0.4243	0.4802	**0.6272**	0.6241	0.6257	0.2803	0.2241	0.2522
SCML	**0.4232**	**0.7049**	**0.5640**	**0.5403**	0.4203	**0.4803**	0.6238	**0.6337**	**0.6288**	**0.2886**	**0.2258**	**0.2572**

TABLE III
THE MAP@500 COMPARISON OF CONTINUAL LEARNING METHODS ON NUS-WIED, AND COCO DATASETS.

Method	NUS-WIDE			COCO		
	I→T	T→I	Avg	I→T	T→I	Avg
DeepSM [36]	0.8359	0.8145	0.8252	0.6189	0.5797	0.5993
DIHN [37]	0.8050	0.7103	0.7577	0.6347	0.5546	0.5947
LwF [35]	0.6258	0.6986	0.6622	0.5021	0.5236	0.5129
EWC [7]	0.6245	0.6769	0.6507	0.6067	0.5815	0.5941
RWalk [25]	0.7693	0.7567	0.7630	0.6160	0.6044	0.6102
A-GEM [27]	0.7955	0.7835	0.7895	0.6485	0.5939	0.6212
LwF+P	0.8520	0.8129	0.8324	0.6501	0.6098	0.6300
EWC+P	0.8575	0.8203	0.8389	0.6459	0.6037	0.6248
RWalk+P	0.8584	0.8259	0.8421	0.6542	0.6145	0.6343
A-GEM+P	0.8605	0.8258	0.8432	0.6502	0.6113	0.6307
SCML	**0.8607**	**0.8298**	**0.8453**	**0.6589**	**0.6198**	**0.6394**

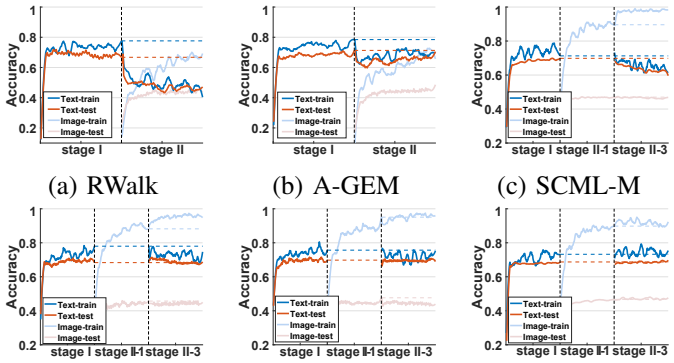


(a) WIKI (b) MIRFLICKR (c) NUSWIDE

Fig. 4. The training and testing accuracy of different modalities at different stages on three datasets when performing sequential cross-modal learning.

for NUS-WIDE, 250, 0.01, 0.6 for MIRFLICKR, 400, 0.1, 0.5 for COCO) for optimization. At the first step of stage II, we use SGD (setting lr=0.1, dropout=0.5 for WIKI, 0.1, 0.6 for NUS-WIDE, 0.1, 0.6 for MIRFLICKR, 0.1, 0.5 for COCO) for plug P2 training. At the second step of stage II, we use Adam (setting lr=0.0001, dropout=0.5 for WIKI, 0.0001, 0.6 for NUS-WIDE, 0.0001, 0.6 for MIRFLICKR, 0.0001, 0.6 for COCO) for the optimization of meta-learner. The w_p, $\mathrm{N_{mem}}$, and the size of $\mathrm{D}^{(2)}_{\mathrm{meta-train}}$ are set to {1, 200, 1,086} for WIKI, {20, 256, 2,500} for MIRFLICKR, NUS-WIDE, and {30, 200, 5,000} for COCO. 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.

For CMPlace dataset, we perform SCML in five stages,



(a) RWalk (b) A-GEM (c) SCML-M

(d) RWalk+P (e) A-GEM+P (f) SCML

Fig. 5. The training and testing accuracy of different continual learning methods at different stages on WIKI dataset.

we set the modality sequence to 'NAT-CLP-LDR-DSC-SPT' (other sequences are illustrated in Sec. IV-E), the batch size to 64, the $\mathrm{N_{mem}}$ to 200, λ to 0.1. At the stage I, we use SGD (with epochs=50, lr=0.01, drop.=0.5) to train P1 and S. At stages II to V, we first use SGD (with lr=0.1, drop.=0.5) to train plugs P2 to P5, then use Adam (with lr=0.0001, drop.=0.5, $\mathrm{N_{meta-train}} = \mathrm{N}/2$) to train meta-learner, where N is the number of training samples of new modality.

C. Experimental Results

1) Overcoming catastrophic forgetting: We first conducted experiments (in the SCML, we first trained P1 and S with one modality jointly and then trained P2 and S with another together) to confirm the *catastrophic forgetting* in sequential cross-modal learning. Fig. 4 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 verified the effectiveness of meta-learner for overcoming catastrophic forgetting in our SCML. **SCML-M** denotes that we use SGD instead of the learned optimizer M to update the unified model S at the second stage. Fig. 5 (c) and (f) report the accuracy of two modalities on the WIKI dataset at different stages (we do not report the stage II-2 since S has no changes) respectively. We see that the accuracy of text

TABLE IV
THE COMPARISON OF SEQUENTIAL (DIFFERENT SEQUENCE ORDER) AND PARALLEL MANNER ON WIKI, MIRFLICKR, NUS-WIED, COCO DATASETS.

Method	WIKI (MAP)			MIRFLICKR (NDCG@500)			NUS-WIDE (NDCG@500)			COCO (NDCG@500)		
	I→T	T→I	Avg	I→T	T→I	Avg	I→T	T→I	Avg	I→T	T→I	Avg
PCML	0.3830	0.7087	0.5458	0.4305	**0.4222**	0.4264	0.6148	0.5634	0.5891	0.2656	0.2249	0.2453
$SCML_{T\rightarrow I}$	**0.4232**	0.7049	**0.5640**	0.4671	0.4200	0.4436	0.5844	0.5793	0.5818	0.2746	**0.2324**	0.2535
$SCML_{I\rightarrow T}$	0.3855	**0.7126**	0.5403	**0.5491**	0.4203	**0.4803**	**0.6238**	**0.6337**	**0.6288**	**0.2886**	0.2258	**0.2572**

TABLE V
COMPARISON OF DIFFERENT REAL-VALUED CROSS-MODAL LEARNING METHODS ON WIKI, MIRFLICKR, NUS-WIED, AND COCO DATASETS.

Method	WIKI (MAP)			MIRFLICKR (NDCG@500)			NUS-WIDE (NDCG@500)			COCO (NDCG@500)		
	I→T	T→I	Avg	I→T	T→I	Avg	I→T	T→I	Avg	I→T	T→I	Avg
TV-CCA [40]	0.2890	0.4966	0.3928	0.3033	0.3034	0.3034	0.5129	0.5050	0.5090	0.2221	0.1955	0.2088
corAE [38]	0.3792	0.2215	0.3004	0.4591	0.3268	0.3930	0.5148	0.5234	0.5191	0.2900	0.2267	0.2583
CDPAE [39]	0.2711	0.3050	0.2881	0.3616	0.3677	0.3647	0.5535	0.5359	0.5447	0.2326	0.2251	0.2289
LCFS [41]	0.3578	0.5624	0.4601	0.3576	0.3243	0.3409	0.5725	0.5800	0.5762	0.2965	0.2073	0.2519
JFSSL [42]	0.4253	0.6654	0.5454	0.3479	0.2971	0.3225	0.5726	0.5355	0.5540	0.2878	0.1914	0.2396
VSE++ [43]	0.2860	0.3629	0.3245	0.3825	0.3525	0.3675	0.6113	0.5617	0.5865	0.1766	0.1908	0.1837
DAML [44]	0.3790	0.4898	0.4344	0.3462	0.3281	0.3372	0.5616	0.5102	0.5359	0.2034	0.2009	0.2022
SSPE [17]	0.3912	**0.7169**	0.5540	0.4629	0.3968	0.4298	0.5832	0.5331	0.5582	0.2989	**0.2962**	0.2975
SCML	**0.4907**	0.6885	**0.5896**	**0.5519**	**0.3994**	**0.4756**	**0.6854**	**0.6190**	**0.6522**	**0.3669**	0.2362	**0.3015**

TABLE VI
THE MAP@500 COMPARISON OF REAL-VALUED CROSS-MODAL LEARNING METHODS ON NUS-WIED, AND COCO DATASETS.

Method	NUS-WIDE			COCO		
	I→T	T→I	Avg	I→T	T→I	Avg
TV-CCA [40]	0.6393	0.6052	0.6222	0.5838	0.5166	0.5502
corAE [38]	0.7151	0.7143	0.7147	0.6502	0.5348	0.5925
CDPAE [39]	0.7604	0.7202	0.7403	0.5918	0.5692	0.5805
LCFS [41]	0.7667	0.7698	0.7683	0.6824	0.5427	0.6125
JFSSL [42]	0.7790	0.7417	0.7604	0.7195	0.4404	0.5799
VSE++ [43]	0.8041	0.7462	0.7752	0.5295	0.5547	0.5421
DAML [44]	0.7576	0.7223	0.7400	0.6457	0.6166	0.6312
SSPE [17]	0.7704	0.7246	0.7475	0.6395	**0.6682**	0.6539
SCML	**0.8725**	**0.8061**	**0.8393**	**0.7716**	0.6186	**0.6951**

TABLE VIII
THE MAP COMPARISON OF DIFFERENT CROSS-MODAL HASHING METHODS ON NUS-WIED, COCO DATASETS WITH DIFFERENT LENGTH.

	Image to Text				Text to Image			
Method	16 bits	32 bits	64 bits	128 bits	16 bits	32 bits	64 bits	128 bits
	NUS-WIDE dataset (MAP@500)							
CMFH [45]	0.6868	0.7023	0.7328	0.7477	0.6553	0.6667	0.6951	0.7056
UCH [46]	0.7144	0.7148	0.7231	0.7222	0.6802	0.7011	0.6957	0.7178
SCM [47]	0.6944	0.7068	0.7219	0.7270	0.6825	0.6972	0.7067	0.7095
SePH [5]	0.7802	0.7898	0.7988	0.8122	0.7310	0.7300	0.7302	0.7372
DCMH [31]	0.7933	0.8253	0.8396	0.8414	0.7384	0.7470	0.7441	0.7416
TDH [48]	0.8332	0.8301	0.8375	0.8409	0.7384	0.7443	0.7440	0.7381
SSAH [15]	0.8182	0.8313	0.8279	0.8036	0.7336	0.7453	0.7597	0.7367
SCML	**0.8438**	**0.8606**	**0.8725**	**0.8728**	**0.7728**	**0.7892**	**0.8061**	**0.8132**
	COCO dataset (MAP@500)							
CMFH [45]	0.5605	0.6088	0.6530	0.6809	0.5216	0.5319	0.5585	0.5775
UCH [46]	0.5854	0.6088	0.6244	0.6838	0.5099	0.5626	0.6054	0.5954
SCM [47]	0.4615	0.4858	0.4738	0.4162	0.3945	0.3988	0.3989	0.3821
SePH [5]	0.5706	0.6381	0.6624	0.6700	0.5982	0.6344	0.6462	0.6568
DCMH [31]	0.5943	0.6508	0.6847	0.7129	0.5818	0.6417	0.6631	0.6856
TDH [48]	0.6210	0.6441	0.6784	0.6819	**0.6278**	0.6375	0.6389	0.6623
SSAH [15]	0.6124	0.7252	0.7570	0.8112	0.6048	**0.6580**	**0.6867**	**0.7005**
SCML	**0.6651**	**0.7598**	**0.7716**	**0.8113**	0.6129	0.6173	0.6186	0.6275

modality decreases at stage II-3 of SCML-M while that at stage II-3 of SCML dost not, and the score of image modality of SCML obtains a slight increase. This result demonstrates the ability of meta-learner.

Finally, we compared SCML with continual learning methods, EWC, LWF, RWalk, and A-GEM. For a fair comparison, we also modified them by splitting the joint training of P2 and S into two stages, and these methods are called '*+P.' Tables II and III report the results. we see that: 1) The DeepSM beats most original continual learning methods on WIKI, MIRFLICKR, and NUS-WIDE datasets except the COCO dataset. It mainly because the DeepSM can learn simple semantics of each modality of the former three datasets easily by the modality-specific networks, but continual learning methods can maintain complex semantics of the COCO dataset well via transferring knowledge between modalities within a unified model; 2) The SCML could gain advantages over most baselines, e.g., compared to the A-GEM method, the SCML method obtains the relative increase of 20.0%, 6.9%, 5.81%, and 0.9% NDCG scores on average on WIKI, MIRFLICKR, NUS-WIDE, and COCO datasets, respectively. The results show the superiority of our meta-learner for continual learning; 3) As the break of joint learning of P2 and S can prevent the P2's limited update, the '*+P' continual learning methods achieve a significant boost compared to their original version and perform comparably with SCML.

To further investigate the differences of SCML, RWalk, A-GEM, RWalk+P, and A-GEM+P, we report the classification accuracy of two modalities on the WIKI dataset at different stages in Fig. 5. We see that the accuracy scores of text modality of all methods reach a similar value since the same network structure and learning policy at stage I. At stage II, RWalk and A-GEM methods have a decrease in the accuracy score of text modality and have the limited growth of image modality's accuracy score. Whereas RWalk+P, A-GEM+P, and SCML methods avoid this and obtain better performance on image modality at stage II-1. Even so, RWalk+P and A-GEM+P methods still have a subtle performance decrease of text modality at stage II-2, while SCML method keeps its performance. Therefore, the performance gap between different methods mainly comes from the limited learning of

TABLE VII
COMPARISON OF DIFFERENT CROSS-MODAL HASHING METHODS ON WIKI, MIRFLICKR, NUS-WIED, COCO DATASETS WITH DIFFERENT LENGTH.

	Image to Text				Text to 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)											
CMFH [45]	0.2053	0.2397	0.2395	0.2291	0.3299	0.3886	0.3738	0.3181	0.2676	0.3141	0.3066	0.2736
UCH [46]	0.1736	0.2132	0.2015	0.2060	0.3905	0.4808	0.5178	0.5303	0.2821	0.3470	0.3597	0.3681
SCM [47]	0.1807	0.1712	0.1698	0.1707	0.6695	0.6911	0.6833	0.7002	0.4251	0.4312	0.4265	0.4355
SePH [5]	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 [31]	0.3724	0.4366	0.4369	0.3521	0.6169	0.6610	0.6011	0.5635	0.4947	0.5488	0.5190	0.4578
TDH [48]	0.3571	0.3927	0.4005	0.3892	0.6301	**0.7015**	0.7048	0.7027	0.4936	0.5471	0.5526	0.5459
SSAH [15]	0.2844	0.3810	0.4212	0.4118	0.4013	0.6851	**0.7349**	**0.7249**	0.3428	0.5331	0.5780	0.5684
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)											
CMFH [45]	0.2908	0.3059	0.3099	0.3162	0.2830	0.3012	0.3054	0.3054	0.2869	0.3035	0.3076	0.3108
UCH [46]	0.3017	0.3114	0.3197	0.3362	0.3027	0.3000	0.3157	0.3051	0.3022	0.3057	0.3177	0.3206
SCM [47]	0.3229	0.3449	0.3573	0.3628	0.2959	0.3105	0.3222	0.3256	0.3094	0.3277	0.3397	0.3442
SePH [5]	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 [31]	0.3857	0.4013	0.4162	0.4046	0.3719	0.3819	0.3877	0.3894	0.3788	0.3916	0.4020	0.3970
TDH [48]	0.4379	0.4491	0.4491	0.4458	0.3789	0.3978	0.3924	0.3937	0.4084	0.4234	0.4208	0.4197
SSAH [15]	0.4203	0.4392	0.4626	0.4681	0.3648	0.3815	0.3710	0.3923	0.3926	0.4103	0.4168	0.4302
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)											
CMFH [45]	0.4875	0.5012	0.5270	0.5394	0.4642	0.4775	0.4998	0.5091	0.4758	0.4893	0.5134	0.5242
UCH [46]	0.5202	0.5175	0.5227	0.5175	0.4546	0.5103	0.5008	0.5232	0.4874	0.5139	0.5118	0.5203
SCM [47]	0.5075	0.5149	0.5299	0.5308	0.4941	0.5010	0.5141	0.5143	0.5008	0.5080	0.5220	0.5226
SePH [5]	0.5726	0.5809	0.5867	0.6043	0.5264	0.5285	0.5245	0.5337	0.5495	0.5547	0.5556	0.5690
DCMH [31]	0.5757	0.6239	0.6353	0.6391	0.5375	0.5421	0.5435	0.5438	0.5566	0.5830	0.5894	0.5915
TDH [48]	0.6196	0.6242	0.6305	0.6324	0.5329	0.5340	0.5368	0.5302	0.5763	0.5791	0.5837	0.5813
SSAH [15]	0.6026	0.6331	0.6263	0.6140	0.5702	0.5781	0.5875	0.5387	0.5864	0.6056	0.6069	0.5763
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**
	COCO dataset (NDCG@500)											
CMFH [45]	0.2224	0.2571	0.2899	0.3098	0.1982	0.2182	0.2398	0.2539	0.2103	0.2377	0.2649	0.2819
UCH [46]	0.2234	0.2588	0.2757	0.3143	0.1895	0.2356	0.2557	0.2687	0.2064	0.2472	0.2652	0.2915
SCM [47]	0.1494	0.1735	0.1610	0.1270	0.1194	0.1288	0.1260	0.1126	0.1344	0.1512	0.1435	0.1198
SePH [5]	0.2052	0.2623	0.2889	0.3152	0.1968	0.2315	0.2475	0.2655	0.2010	0.2469	0.2682	0.2903
DCMH [31]	0.1638	0.2046	0.2591	0.3428	0.1751	0.2044	**0.2660**	0.2860	0.1537	0.1748	0.2626	0.3144
TDH [48]	0.2447	0.2692	0.2946	0.3138	**0.2249**	**0.2420**	0.2521	0.2712	0.2348	0.2556	0.2733	0.2925
SSAH [15]	0.1955	0.2797	0.3157	0.3917	0.1870	0.2382	0.2548	**0.3114**	0.1913	0.2590	0.2853	**0.3516**
SCML	**0.2828**	**0.3168**	**0.3669**	**0.3938**	0.2002	0.2157	0.2362	0.2438	**0.2415**	**0.2662**	**0.3015**	0.3188

new modality and the decrease of old's performance.

2) 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 performed parallel cross-modal learning as Fig. 1 (a) shows, which was called **PCML** (jointly learn P1, P2, and S with two modalities) and compared it with SCML. Table IV 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 the sequential learning manner for modalities' distribution.

3) Comparisons with Cross-modal Methods: Tables V, VI, VII, and VIII report the results. From Tables V and VI, we observe that: 1) The SSPE can beat most baselines, since it takes advantage of original feature similarity, semantic similarity, and semantic discriminative information simultaneously to guide learning while others only use one or two of them to learn cross-modal representation, e.g., VSE++ uses semantic similarity information; 2) The SCML method outperforms other compared methods on most datasets and tasks. Specifically, compared to the best real-value methods, the SCML achieves boosts of 3.5%, 4.6%, 6.5%, and 0.4% NDCG scores on average on WIKI, MIRFLICKR, NUS-WIDE, and COCO datasets, respectively. We could explain this by the difference between the manner to narrow the discriminative ability gap between different modalities features. The parallel manner of compared methods prefers to sacrifice the high discriminative of one modality to compensate another, but the SCML seeks to keep the previous's discriminative and gradually boosting the later's. The SCML achieves consistent superiority in terms of MAP scores as in NDCG scores.

From Tables VII, and VIII, we see that: 1) The deep cross-modal hashing methods perform better than shallow hashing methods, and the supervised ones outperform unsupervised ones. This result indicates the significance of supervised information and non-linear transformation for cross-modal learning. 2) The SCML method obtains the relative increase of 1.1%∼22.7%, 3.2%∼6.9%, and 3.4%∼8.1% NDCG scores compared to the state-of-the-art hashing methods TDH/SSAH on average with different bits on WIKI, MIRFLICKR, and NUS-WIDE datasets, respectively. The results on Table VIII is similar to that on Table VII, but a small NDCG performance gap may reflect a big MAP performance gap due to their different definitions. 3) However, we notice that the performance of the SCML method is inferior to that of the TDH/SSAH methods on the 'Text to Image' task of the COCO dataset. The main reason is that SCML can not exploit classification loss to fully capture similarity structure among samples of the COCO dataset containing complex and abundant semantic information (i.e., 80 semantic concepts), whereas the TDH/SSAH methods

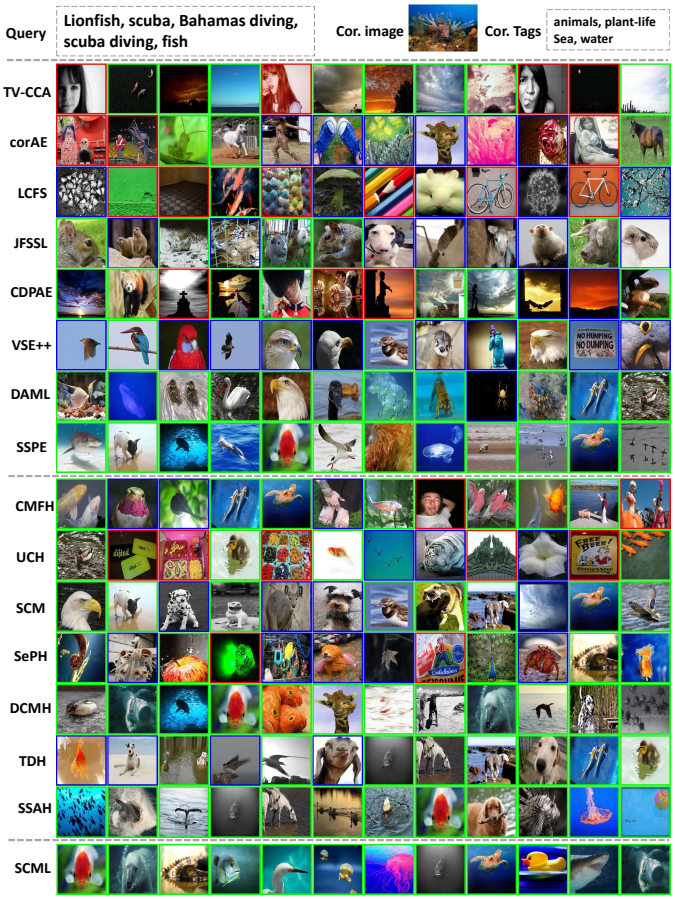
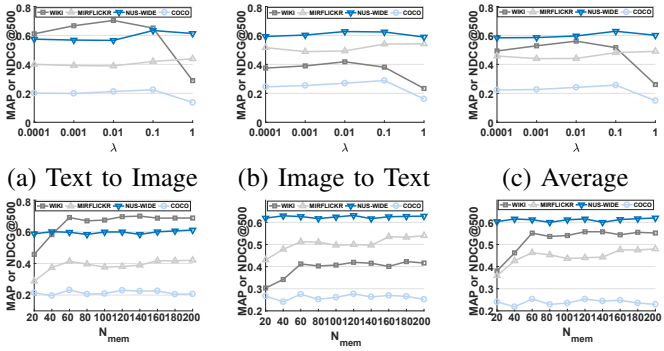


Fig. 6. Retrieval examples of 'Text to Image' on MIRFLICKR dataset. Red border denotes irrelevant; blue denotes sharing one tag with the query, green denotes sharing two tags with the query at least.

Fig. 7. Retrieval examples of 'Image to Text' on COCO dataset. Red dot denotes irrelevant; blue denotes sharing one tag with the query, green denotes sharing two tags with the query at least.



(a) Text to Image (b) Image to Text (c) Average

(d) Text to Image (e) Image to Text (f) Average

Fig. 8. The compact of parameters λ and N_{mem} on the WIKI, MIRFKICKR, NUS-WIDE, and COCO datasets.

can do this better by pairwise or triplet loss.

Fig. 6 shows 'Text to Image' search example of compared methods on MIRFLICKR dataset. As can be seen, the SCML method tends to retrieve more relevant images than others for the query containing certain concepts, e.g., sea and animals. Fig. 7 shows 'Image to Text' search example of compared methods on COCO dataset. We observe that the SCML method can roughly capture the similarity across images and text descriptions through the alignment of tags. Compared to other methods, the top 5 retrieved textual results of SCML are more meaningful and relevant.

D. Empirical Analysis

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

2) Sensitivity to Parameters: We analyzed the effect of balance weight λ and the memory size $\mathrm{N}_{\mathrm{mem}}$ of $\mathrm{D}^{(1)}_{\mathrm{memory}}$. We initially set $\{\lambda, \mathrm{N}_{\mathrm{mem}}\}$ to $\{0.01, 200\}$ for WIKI, $\{0.1, 256\}$ for MIRFLICR, NUS-WIDE and COCO. Then, we separately tune them with other parameters fixing and report the cross-modal retrieval performance in Fig. 8

From Fig. 8(a)-(c), we see that the SCML method achieves the best performance at a particular value, since a smaller λ may lead to dramatic changes of the model and cause knowledge forgetting, while a larger λ may encourage the

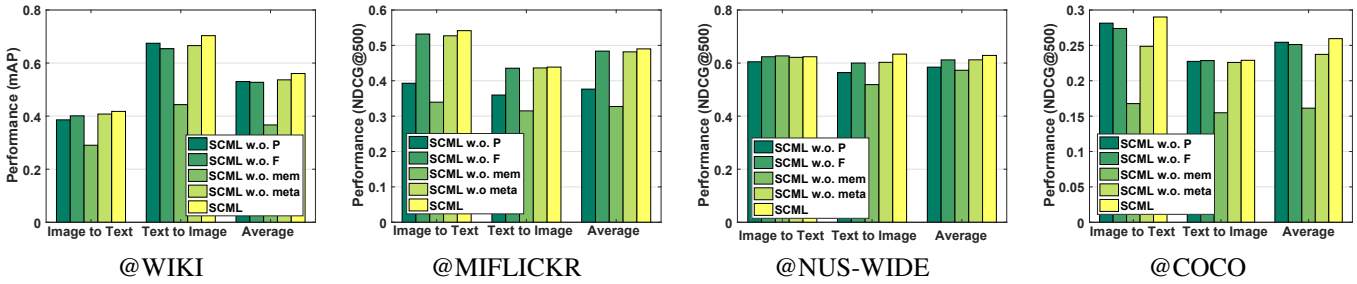


Fig. 9. Evaluations (mAP and NDCG) of the proposed SCML with ablating different components.

TABLE IX
MAP(%) COMPARISON OF DIFFERENT METHODS ON CMPLACES DATASET. EACH COLUMN SHOWS A DIFFERENT QUERY-TARGET PAIR. ON THE FAR RIGHT, WE AVERAGE OVER ALL PAIRS.

Query	NAT				CLP				SPT				LDR				DSC				mean
Target	CLP	SPT	LDR	DSC	NAT	SPT	LDR	DSC	NAT	CLP	LDR	DSC	NAT	CLP	SPT	DSC	NAT	CLP	SPT	LDR	mAP
PCML	26.1	15.1	21.2	**22.1**	34.0	17.7	24.9	**28.3**	21.0	21.5	18.5	18.0	12.8	13.1	9.10	**13.5**	8.21	8.14	5.14	7.15	17.2
DeepSM	22.1	**19.8**	24.5	22.1	30.8	**24.4**	26.3	27.9	24.9	22.1	**22.5**	**23.3**	9.00	9.10	8.86	10.1	4.08	4.29	3.92	4.07	17.2
LWF [35]	14.9	9.20	10.5	7.83	22.3	11.3	16.6	9.61	11.5	11.7	10.3	9.54	8.51	8.85	5.90	5.85	4.02	4.49	4.03	4.24	9.56
EWC [7]	20.3	12.8	15.6	13.8	29.2	16.5	21.2	17.9	21.2	19.6	19.5	17.8	11.1	10.7	9.10	9.97	5.90	5.87	**5.37**	5.77	14.4
A-GEM [27]	27.0	10.2	23.4	16.1	34.3	9.9	24.2	16.3	21.9	16.6	15.4	10.2	14.7	12.6	4.93	7.79	6.70	5.95	2.82	5.70	14.3
A-GEM$_{s_1}$	24.9	8.88	20.9	13.0	35.2	8.97	23.1	14.7	20.7	15.6	14.6	8.96	13.9	12.1	5.03	7.28	6.70	5.05	2.74	5.21	13.3
A-GEM$_{s_2}$	22.4	8.01	23.5	11.2	31.7	8.04	26.5	10.8	16.2	12.3	12.7	7.16	17.5	14.9	5.29	7.31	6.47	5.35	2.61	5.57	12.7
A-GEM$_{s_3}$	22.0	9.55	20.8	11.2	30.8	8.89	19.6	12.2	20.8	14.4	12.9	7.03	15.2	11.5	4.92	5.64	8.45	7.03	3.37	5.72	12.6
RWalk [25]	**29.7**	13.0	27.6	21.3	36.5	13.2	28.6	22.1	25.5	20.9	18.6	14.5	18.8	15.3	6.29	9.69	7.78	6.87	3.17	5.99	17.2
RWalk$_{s_1}$	28.9	10.3	23.1	16.2	**39.1**	13.8	30.5	23.8	23.2	21.0	15.5	11.0	15.3	14.5	5.06	7.66	7.29	6.86	2.81	5.06	16.0
RWalk$_{s_2}$	26.1	11.1	28.5	16.8	34.4	11.3	30.8	18.1	23.6	18.7	20.6	11.3	**20.5**	17.2	7.49	10.9	7.28	6.31	3.15	6.61	16.6
RWalk$_{s_3}$	25.8	10.3	23.2	19.3	32.7	10.2	23.6	19.2	22.6	17.6	15.4	12.9	16.3	13.2	5.39	8.37	10.0	7.92	3.98	6.61	15.2
SCML	27.0	17.3	22.5	18.1	34.5	19.1	25.6	20.2	**28.4**	**23.9**	21.3	17.4	14.1	14.0	**9.19**	9.82	6.31	6.50	4.66	5.24	**17.3**
SCML$_{s_1}$	28.8	9.50	21.5	14.6	38.3	13.8	**31.0**	24.0	21.5	21.2	15.2	10.9	14.8	15.5	5.15	7.35	6.34	6.48	2.81	4.84	15.7
SCML$_{s_2}$	25.5	11.1	**28.6**	16.3	32.9	11.8	30.9	17.7	22.4	18.3	20.9	11.7	19.6	**17.5**	7.54	10.9	7.48	6.36	3.50	7.16	16.4
SCML$_{s_3}$	23.9	9.37	20.4	17.0	31.5	9.67	22.6	19.2	20.5	17.8	15.0	12.7	14.3	12.7	4.95	7.96	**10.2**	**8.20**	3.97	**7.23**	14.5

less change of model and depress the learning of new modality data. From Fig. 8(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.

3) Ablation Study: To analyze the effectiveness of different stages and components in the proposed SCML method, We separately removed: 'the stage II-1', 'the stage II-2,3', 'the memory $D^{(1)}_{memory}$' with others remained to evaluate their influence on the final performance. These three models were called **SCML w.o. P**, **SCML w.o.F**, and **SCML w.o. mem**. We also replaced M with SGD for fine-tuning the unified model S at stage II in SCML to investigate the M's advantage, which were denoted as **SCML w.o. meta**. Fig.9 shows the results of two cross-modal retrieval tasks on four datasets.

We can see that separately removing the training stages and memory damage the retrieval performance of the 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.

E. Long Sequential Cross-modal Learning

In this section, we verified the long sequential cross-modal learning ability of our SCML method on CMPlace dataset. We first compared SCML with parallel cross-modal learning method **PCML**, continual learning methods **EWC** [7], **LWF** [35], **RWalk** [25], **A-GEM** [27], and **Deep-SM** [36]. Then, we performed RWalk, A-GEM, and SCML with different modality sequences to investigate their influences. We defined three sequences which start from more abstract modality than 'NAT'. s_1 is '**CLP-NAT-LDR-DSC-SPT**', s_2 is '**LDR-CLP-NAT-DSC-SPT**', s_3 is '**DSC-CLP-LDR-NAT-SPT**'. '$*_{s_k}$' denotes the $*$ method is trained on the s_k sequence. Finally, we studied the ability of SCML for overcoming forgetting and the effectiveness of different components.

Table IX reports the mAP results of compared methods, where each "Query-Target" pair means using the testing data of the "Query" modality to retrieve relevant data of "Target" modality. As this dataset is extremely noisy and diverse, the absolute mAP for all methods is low. From Table IX, we can see that: 1) The PCML method beats most compared continual learning methods, and the DeepSM method achieves comparable results with the PCML. This result indicates that training a unified model by all modalities data at the same time is practical when modality data are complex; 2) The SCML method performs comparably with the PCML method and the best continual learning method RWalk, which demonstrates the long sequential learning ability of SCML and the effectiveness of the sequential manner for cross-modal learning.

After further comparing SCML and PCML methods, we see that the SCML method outperforms the PCML method on most cross-modalities retrieval tasks (e.g., on the "SPT-NAT" task, from 21.0% obtained by PCML to 28.2%). However, for

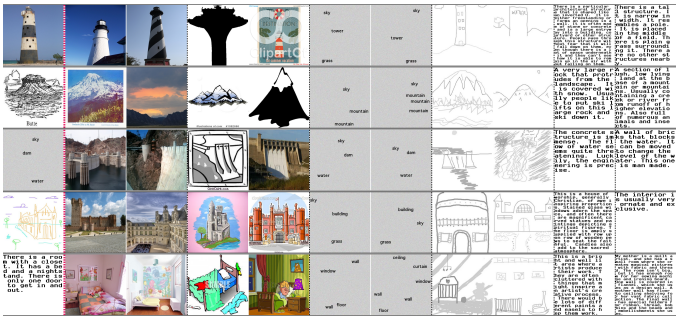


Fig. 10. Quality retrieval examples on the CMPlaces dataset. The first column represents the query, and the top 2 results for each modality are shown.

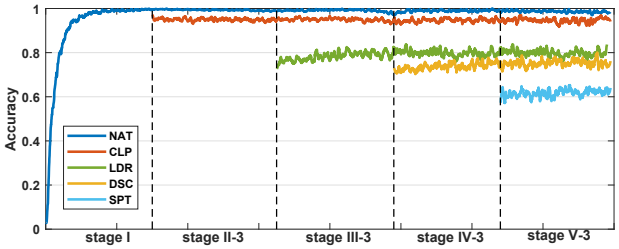


Fig. 11. The training accuracy of different modalities at different stages on the CMPlace dataset when performing SCML method

several tasks, the results of the SCML method do not show the consistent superior, e.g., on the "NAT-DSC" task. We notice that most of these tasks are related to the DSC modality, which is also observed on other continual learning methods, e.g., the performances of the RWalk method related to DSC modality are all lower than those of the PCML. To explain this, we analyzed the learned testing SCML feature of each modality and found that the discriminative power of DSC was weaker than that of other modalities. This observation explains the low performance of the SCML method on tasks related to DSC and indicates that SCML would not sacrifice the acquired knowledge to perform the new modality.

Besides, comparing SCML and SCML$_{s_*}$ methods, we can see that the SCML method performs better than others since the discriminative of 'NAT' modality fed at the first stage on SCML is more powerful than others. This result shows that feeding the more discriminative modality data to train SCML in the early stage is more useful for the learning of later stages. Notably, the results related to some modality are better if this modality is trained more early. For example, the 'DSC-*' retrieval results of SCML$_{s_3}$ are better than that of SCML when we train SCML on 'DSC' modality first. We also observe similar results in RWalk and A-GEM methods.

Fig. 10 shows the quality retrieval examples of our SCML method with different modalities queries. As can be seen, our SCML method can return relevant results for queries containing the specific semantic concepts, e.g., "tower," "dam," and "building." It confirms that our SCML method can effectively capture the similarities of unaligned cross-modal data even though the cross-modal sequence is long.

To investigate the ability of SCML for overcome-forgetting learning and the effectiveness of different components, we firstly report the classification accuracy of each modality at

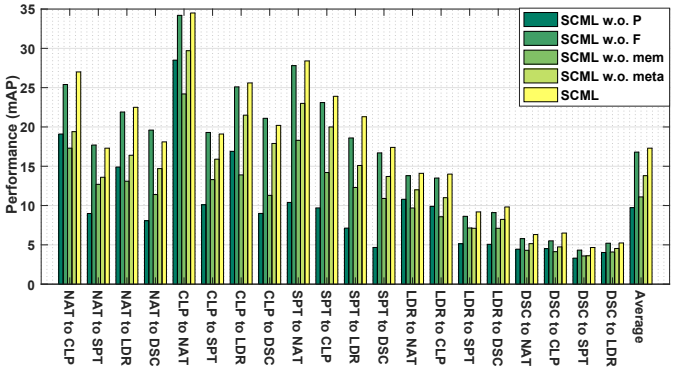


Fig. 12. Evaluations (mAP) of the proposed SCML with ablating different components on the CMPlace dataset.

stage I to V in Fig. 11 (for stage II-V, we only report the first and third steps where the unified model actually changes). We observe that the performance of each modality does not appear catastrophic decrease, which validates the overcoming-forgetting ability of SCML in long sequential cross-modal learning. We then do the same **ablation study (see Sec. IV-D)** on CMPlace dataset and report the results in Fig. 12. We can see that separately removing the training stages, and memory will damage the retrieval performance of SCML method to varying degrees, which re-confirms the contributions of different components.

V. Conclusion

In this paper, we have presented a novel cross-modal representation learning method, name SCML, for the 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 four popular datasets and a five-modality dataset.

References

[1] T. Baltrusaitis, C. Ahuja, and L. Morency, "Multimodal machine learning: A survey and taxonomy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 2, pp. 423–443, 2019.

[2] J. Gu, J. Cai, S. R. Joty, L. Niu, and G. Wang, "Look, imagine and match: Improving textual-visual cross-modal retrieval with generative models," in *CVPR*, 2018, pp. 7181–7189.

[3] N. C. Mithun, R. Panda, E. E. Papalexakis, and A. K. Roy-Chowdhury, "Webly supervised joint embedding for cross-modal image-text retrieval," in *ACM MM*, 2018, pp. 1856–1864.

[4] H. Liu, M. Lin, S. Zhang, Y. Wu, F. Huang, and R. Ji, "Dense auto-encoder hashing for robust cross-modality retrieval," in *ACM MM*, 2018, pp. 1589–1597.

[5] Z. Lin, G. Ding, M. Hu, and J. Wang, "Semantics-preserving hashing for cross-view retrieval," in *CVPR*, 2015, pp. 3864–3872.

[6] F. Kemény and B. Meier, "Multimodal sequence learning," *Acta psychologica*, vol. 164, pp. 27–33, 2016.

[7] J. Kirkpatrick, R. Pascanu, N. C. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," *CoRR*, vol. abs/1612.00796, 2016.
[8] D. Lopez-Paz and M. Ranzato, "Gradient episodic memory for continual learning," in *NeurIPS*, 2017, pp. 6470–6479.
[9] C. V. Nguyen, Y. Li, T. D. Bui, and R. E. Turner, "Variational continual learning," 2018.
[10] J. Yoon, E. Yang, J. Lee, and S. J. Hwang, "Lifelong learning with dynamically expandable networks," in *ICLR*, 2018.
[11] M. Andrychowicz, M. Denil, S. G. Colmenarejo, M. W. Hoffman, D. Pfau, T. Schaul, and N. de Freitas, "Learning to learn by gradient descent by gradient descent," in *NeurIPS*, 2016, pp. 3981–3989.
[12] Y. Cao, M. Long, J. Wang, Q. Yang, and P. S. Yu, "Deep visual-semantic hashing for cross-modal retrieval," in *SIGKDD*, 2016, pp. 1445–1454.
[13] L. Wang, Y. Li, and S. Lazebnik, "Learning deep structure-preserving image-text embeddings," in *CVPR*, 2016, pp. 5005–5013.
[14] Z. Ye and Y. Peng, "Multi-scale correlation for sequential cross-modal hashing learning," in *ACM MM*, 2018, pp. 852–860.
[15] C. Li, C. Deng, N. Li, W. Liu, X. Gao, and D. Tao, "Self-supervised adversarial hashing networks for cross-modal retrieval," in *CVPR*, 2018, pp. 4242–4251.
[16] G. Andrew, R. Arora, J. A. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *ICML*, 2013, pp. 1247–1255.
[17] Y. Wu, S. Wang, and Q. Huang, "Learning semantic structure-preserved embeddings for cross-modal retrieval," in *ACM MM*, 2018, pp. 825–833.
[18] Y. Huang, Q. Wu, C. Song, and L. Wang, "Learning semantic concepts and order for image and sentence matching," in *AAAI*, 2018, pp. 6163–6171.
[19] K. Lee, X. Chen, G. Hua, H. Hu, and X. He, "Stacked cross attention for image-text matching," in *ECCV*, 2018, pp. 212–228.
[20] E. Yang, C. Deng, W. Liu, X. Liu, D. Tao, and X. Gao, "Pairwise relationship guided deep hashing for cross-modal retrieval," in *AAAI*, 2017, pp. 1618–1625.
[21] Y. Shen, L. Liu, F. Shen, and L. Shao, "Zero-shot sketch-image hashing," in *CVPR*, 2018, pp. 3598–3607.
[22] G. Song, D. Wang, and X. Tan, "Deep memory network for cross-modal retrieval," *IEEE Trans. Multimedia*, vol. 21, no. 5, pp. 1261–1275, 2019.
[23] J. Serrà, D. Suris, M. Miron, and A. Karatzoglou, "Overcoming catastrophic forgetting with hard attention to the task," in *ICML*, 2018, pp. 4555–4564.
[24] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, and T. Tuytelaars, "Memory aware synapses: Learning what (not) to forget," in *ECCV*, 2018, pp. 144–161.
[25] A. Chaudhry, P. K. Dokania, T. Ajanthan, and P. H. S. Torr, "Riemannian walk for incremental learning: Understanding forgetting and intransigence," in *ECCV*, 2018, pp. 556–572.
[26] A. A. Rusu, N. C. Rabinowitz, G. Desjardins, H. Soyer, J. Kirkpatrick, K. Kavukcuoglu, R. Pascanu, and R. Hadsell, "Progressive neural networks," *CoRR*, vol. abs/1606.04671, 2016. [Online]. Available: http://arxiv.org/abs/1606.04671
[27] A. Chaudhry, M. Ranzato, M. Rohrbach, and M. Elhoseiny, "Efficient lifelong learning with A-GEM," in *ICLR*, 2019.
[28] N. Rasiwasia, J. C. Pereira, E. Coviello, G. Doyle, G. R. G. Lanckriet, R. Levy, and N. Vasconcelos, "A new approach to cross-modal multimedia retrieval," in *ACM MM*, 2010, pp. 251–260.
[29] M. J. Huiskes and M. S. Lew, "The MIR flickr retrieval evaluation," in *Proceedings of the 1st ACM SIGMM International Conference on Multimedia Information Retrieval, MIR 2008, Vancouver, British Columbia, Canada, October 30-31, 2008*, 2008, pp. 39–43.
[30] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.
[31] Q. Jiang and W. Li, "Deep cross-modal hashing," in *CVPR*, 2017, pp. 3270–3278.
[32] R. Kiros, Y. Zhu, R. Salakhutdinov, R. S. Zemel, R. Urtasun, A. Torralba, and S. Fidler, "Skip-thought vectors," in *NeurIPS*, 2015, pp. 3294–3302.
[33] B. Zhou, À. Lapedriza, J. Xiao, A. Torralba, and A. Oliva, "Learning deep features for scene recognition using places database," in *NeurIPS*, 2014, pp. 487–495.
[34] K. Järvelin and J. Kekäläinen, "Ir evaluation methods for retrieving highly relevant documents," in *SIGIR*, 2000, pp. 41–48.
[35] Z. Li and D. Hoiem, "Learning without forgetting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 12, pp. 2935–2947, 2018.
[36] Y. Wei, Y. Zhao, C. Lu, S. Wei, L. Liu, Z. Zhu, and S. Yan, "Cross-modal retrieval with CNN visual features: A new baseline," *IEEE Trans. Cybernetics*, vol. 47, no. 2, pp. 449–460, 2017.
[37] D. Wu, Q. Dai, J. Liu, B. Li, and W. Wang, "Deep incremental hashing network for efficient image retrieval," in *CVPR*, 2019, pp. 9069–9077.
[38] F. Feng, X. Wang, and R. Li, "Cross-modal retrieval with correspondence autoencoder," in *ACM MM*, 2014, pp. 7–16.
[39] Y. Zhan, J. Yu, Z. Yu, R. Zhang, D. Tao, and Q. Tian, "Comprehensive distance-preserving autoencoders for cross-modal retrieval," in *ACM MM*, 2018, pp. 1137–1145.
[40] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik, "A multi-view embedding space for modeling internet images, tags, and their semantics," *IJCV*, vol. 106, no. 2, pp. 210–233, 2014.
[41] K. Wang, R. He, W. Wang, L. Wang, and T. Tan, "Learning coupled feature spaces for cross-modal matching," in *ICCV*, 2013, pp. 2088–2095.
[42] K. Wang, R. He, L. Wang, W. Wang, and T. Tan, "Joint feature selection and subspace learning for cross-modal retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2010–2023, 2016.
[43] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler, "VSE++: improving visual-semantic embeddings with hard negatives," in *BMVC*, 2018, p. 12.
[44] X. Xu, L. He, H. Lu, L. Gao, and Y. Ji, "Deep adversarial metric learning for cross-modal retrieval," *World Wide Web*, vol. 22, no. 2, pp. 657–672, 2019.
[45] G. Ding, Y. Guo, and J. Zhou, "Collective matrix factorization hashing for multimodal data," in *CVPR*, 2014, pp. 2083–2090.
[46] C. Li, C. Deng, L. Wang, D. Xie, and X. Liu, "Coupled cyclegan: Unsupervised hashing network for cross-modal retrieval," in *AAAI*, 2019, pp. 176–183.
[47] D. Zhang and W. Li, "Large-scale supervised multimodal hashing with semantic correlation maximization," in *AAAI*, 2014, pp. 2177–2183.
[48] C. Deng, Z. Chen, X. Liu, X. Gao, and D. Tao, "Triplet-based deep hashing network for cross-modal retrieval," *IEEE Trans. Image Processing*, vol. 27, no. 8, pp. 3893–3903, 2018.

Ge Song received the B.Sc. degree in computer science and technology from Zhengzhou University, Zhengzhou, China, in 2014. He is currently working toward the Ph.D. degree at the Nanjing University of Aeronautics and Astronautics, Nanjing, China. His research interests are in image retrieval, machine learning, pattern recognition, and computer vision.

Xiaoyang Tan received his BSc and MSc degrees in computer applications from Nanjing University of Aeronautics and Astronautics (NUAA) in 1993 and 1996, respectively. Then he worked at NUAA in June 1996 as an assistant lecturer. He received a PhD degree from Department of Computer Science and Technology of Nanjing University, China, in 2005. From September 2006 to October 2007, he worked as a postdoctoral researcher in the LEAR (Learning and Recognition in Vision) team at INRIA Rhone-Alpes in Grenoble, France. His research interests are in face recognition, machine learning, pattern recognition, and computer vision. In these fields, he has authored or coauthored over 40 scientific papers.