Joint-teaching: Learning to Refine Knowledge for Resource-constrained Unsupervised Cross-modal Retrieval

Peng-Fei Zhang, Jiasheng Duan, Zi Huang*, Hongzhi Yin
University of Queensland
mima.zpf@gmail.com,j.duan@uqconnect.edu.au,huang@itee.uq.edu.au,h.yin1@uq.edu.au

ABSTRACT

Cross-modal retrieval has received considerable attention owing to its applicability to enable users to search desired information with diversified forms. Existing retrieval methods retain good performance mainly relying on complex deep neural networks and high-quality supervision signals, which deters them from real-world resource-constrained development and deployment. In this paper, we propose an effective unsupervised learning framework named JOint-teachinG (JOG) to pursue a high-performance yet light-weight cross-modal retrieval model. The key idea is to utilize the knowledge of a pre-trained model (a.k.a. the "teacher") to endow the to-be-learned model (a.k.a. the "student") with strong feature learning ability and predictive power. Considering that a teacher model serving the same task as the student is not always available, we resort to a cross-task teacher to leverage transferrable knowledge to guide student learning. To eliminate the inevitable noises in the distilled knowledge resulting from the task discrepancy, an online knowledge-refinement strategy is designed to progressively improve the quality of the cross-task knowledge in a joint-teaching manner, where a peer student is engaged. In addition, the proposed JOG learns to represent the original high-dimensional data with compact binary codes to accelerate the query processing, further facilitating resource-limited retrieval. Through extensive experiments, we demonstrate that in various network structures, the proposed method can yield promising learning results on widely-used benchmarks. The proposed research is a pioneering work for resource-constrained cross-modal retrieval, which has strong potential to be applied to on-device deployment and is hoped to pave the way for further study.

CCS CONCEPTS

• **Information systems → Multimedia and multimodal retrieval**.

KEYWORDS

Knowledge Distillation; Noise Refinery; Cross-modal Retrieval; Unsupervised Learning

ACM Reference Format:
Peng-Fei Zhang, Jiasheng Duan, Zi Huang*, Hongzhi Yin. 2021. Joint-teaching: Learning to Refine Knowledge for Resource-constrained Unsupervised Cross-modal Retrieval. In *Proceedings of the 29th ACM International Conference on Multimedia (MM '21), October 20–24, 2021, Virtual Event, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3474085.3475286

1 INTRODUCTION

The multimedia content, such as image, text and video, has been experiencing exponential growth in recent decades. The emerging demand for searching across these heterogeneous data has boosted cross-modal retrieval research. Recently, a rich line of research has been undertaken to investigate effective and efficient cross-modal representation learning [23, 27, 32], where deep neural networks play a critical role. Despite state-of-the-art performance achieved, there are two bottlenecks in applying deep techniques. First, existing methods usually pursue better performance with deeper and larger architectures, resulting in higher-complexity models. It consequently puts significant pressure on computation and memory during inference. Second, high-quality labelled data is usually required to ensure accurate results, which however is not always available. These limitations hinder the practicability of existing deep methods in real-world resource-constrained applications, which intuitively raises a question: *Can we design a learning framework to obtain a high-performance retrieval model that can be developed and deployed under a resource-constrained regime?*

Advances in Knowledge Distillation (KD) [12, 13, 48] provide inspiration for answers - it is possible to develop a "teacher-student" learning scheme to train a high-performance yet light-weight model (a.k.a. the "student") by leveraging the knowledge from a pre-trained model (a.k.a. the "teacher"). The rationale behind is that the predictions of the teacher contain discriminative information about data, which can be specific as a kind of knowledge to teach the student. For example, in the task of classification where KD originates from, a classifier is capable of estimating the probability that an object belongs to a certain category. The predicted probability is considered as the knowledge owned by the teacher, which is leveraged as a soft target for students to mimic. It is commonly believed that the soft target can favor the generalization of the student.

The ideal scenario assumed by the vanilla KD is that a good teacher exists who serves the same task as the student does, so that the knowledge can be easily transferred from the teacher to the student by directly aligning their outputs. Unfortunately, such a teacher is not always available in real-world applications. To solve this, a natural way is to hire a teacher on other tasks, provided that many pre-trained models exist on the Internet. The obstacle of applying cross-task knowledge distillation is the output difference

[*] Corresponding author.

MM '21, October 20–24, 2021, Virtual Event, China
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8651-7/21/10…$15.00
https://doi.org/10.1145/3474085.3475286

between the teacher and the student, which makes the direct learning experience transition inapplicable. For example, we pursue a *retrieval model* (i.e., the "student") on the dataset MIRFlickr-25K [15], while only having access to a *classifier* (i.e., the "teacher") on another dataset ImageNet [7]. In this situation, it is challenging to utilize the probabilistic information over classes from the teacher to train a model for the retrieval purposes. In this paper, we exploit the solution to bridge the task gap, accomplishing the knowledge transfer between models serving different tasks.

The key insight to deal with the challenge is to construct a transferable term. It is noted that, on different tasks, feature representation learning is always an essential component. The feature embedding learned from a model usually contains rich information, based on which we can capture the correlation between data samples. The correlation is inherent in data and independent of tasks, with which we can link the teacher and student models serving different tasks. For example, the embedding learned by a classifier that can describe the specialties of different concepts (e.g., "cat" and "dog") is also feasible for similarity search, where the discrepancy across categories can be naturally discriminated. It is worth noting that the above claim is in the premise that two models engage in the same or similar datasets in terms of the modality and feature space, which ensures a possible knowledge transition. In light of this, it is feasible to take an embedding-based knowledge distillation scheme to enable the knowledge transition across tasks. Nevertheless, a potential problem in this setting is that there exist inevitable noises in distilled knowledge due to the imperfection of the teacher and the limited transferability of the feature, which would hamper the training of the student model. The refinery of the noisy knowledge has crucial impacts on the final performance, which however is neglected by existing KD methods.

In this paper, we propose the JOint-teachinG (JOG) framework to train a high-performance yet light-weight cross-modal retrieval model under the reliable supervision ensured by offline cross-task knowledge distillation and online noise refinery. In detail, the proposed JOG exploits the discriminative ability of the feature learned from the teacher and takes it as the "dark knowledge" for the student to mimic in both the embedding level and prediction level. Furthermore, to alleviate the inevitable noises in the distilled knowledge, an online noise refinement strategy is designed to progressively improve the quality of the cross-task knowledge in a joint-teaching manner. Inspired by the mutual teaching strategy [9, 36, 48], we additionally construct a peer model for the student and collaboratively train them to provide extra knowledge for the refinery. Different from the conventional strategy that takes the knowledge of one model to train the other directly, the knowledge of two models are effectively incorporated to achieve a better understanding of the relationship among data samples. To attain a reliable refinery, we maintain their running average models, i.e., Mean-Teacher models [36], to produce an ensemble estimation. By training the network with the progressively optimized knowledge, the proposed JOG endows the student model with strong embedding ability and predictive power. Besides, the proposed JOG learns to represent data with compact binary codes via the hashing techniques [6, 16, 28] to accelerate the query speed, further satisfying the requirement of retrieval in resource-constrained environments.

Our contributions are summarized as follows:

- We introduce JOG, a novel KD-based learning framework for resource-limited unsupervised cross-modal retrieval, which is capable of building a high-performing yet light-weight retrieval model.
- To promote effective learning, the proposed JOG involves both offline cross-task knowledge distillation and online noise refinery to reliably optimize the to-be-learned model.
- The promising results obtained from extensive experiments on benchmark datasets demonstrate the superiority of the proposed JOG method.

2 RELATED WORK

In this section, we briefly review the related work, including cross-modal retrieval and knowledge distillation.

2.1 Cross-modal Retrieval

Cross-modal retrieval aims at enabling people to search across different modality data (e.g., texts, images, etc.) to find out what they prefer. To achieve this, a wide range of cross-model retrieval methods are proposed, which aim to find a common space for data from different modalities so that similarity between them would be directly measured. According to the representation type, existing cross-modal approaches proposed to cope with this issue can be roughly divided into real-valued representation learning [1, 40, 49] and binary-valued representation learning [20, 26, 32].

2.1.1 Real-valued Methods. As the name implies, the real-valued methods learn to describe data with real-valued descriptions. In the literature, conventional methods are based on hand-crafted features, with representative work including Joint Representation Learning (JRL) [45], Joint Graph Regularized Heterogeneous Metric Learning (JGRHML) [44], Multi-view Discriminant Analysis (MvDA) [17]. Despite progress, these traditional methods are restricted to the time-consuming hand-crafted feature extraction.

To tackle this, many efforts have been made to incorporate advanced deep learning techniques into cross-modal learning in order to learn high-quality non-linear features. Depending on if supervised information is used, current methodologies can be roughly categorized into supervised methods and unsupervised methods. In supervised settings, original data is pre-annotated, with which one can precisely capture the semantic relationships between heterogeneous data, thus effectively bridging the modality gap and achieving promising results. For example, [39] proposes to take advantage of the adversarial learning strategy to correlate data from different modalities. [49] correlates data instances in both label space and common space in order to learn discriminative and invariant representations. In spite of the great success achieved, the acquisition of large-scale data with high-quality manual annotations is really hard labor, deterring the practicability of supervised methods. To deal with it, unsupervised methodologies are proposed, which learn descriptions by exploiting the neighborhood relationships underlying original data. For example, [1, 40] learn predictions for multi-modal data based on co-occurrence information. [46] utilizes pair-wise relations and leverages denoising autoencoders to relieve the negative influence of redundant noises of different modalities, to learn similarity preserving representations

2.1.2 Binary-valued Methods. Compared to real-valued methods, learning to represent data with compact binary codes, i.e., hashing learning, is more general in cross-modal retrieval fields due to fast query speed and comparable retrieval accuracy. Prior methods are non-deep, which typically learn linear projection functions to transfer data into a common Hamming space, with representative work including Cross View Hashing (CVH) [20], Inter-Media Hashing (IMH) [34], Collective Matrix Factorization Hashing (CMFH) [8], Semi-Relaxation Supervised Hashing (SRSH) [47].

Due to promising performance, deep hashing learning has been the mainstream in the field and a wide range of work have been presented. On the one hand, in supervised settings, Deep Visual-Semantic Hashing (DVSH) [3] learns joint embeddings for visual and textual data in a visual-semantic fusion framework, where the spatial dependency of images and temporal dynamics of text are exploited. Deep Cross-Modal Hashing (DCMH) [16] designs an end-to-end learning network, which learns hash functions and binary codes synchronously without relaxation. On the other hand, in unsupervised scenarios, Deep Joint-Semantics Reconstructing Hashing (DJSRH) [35] integrates the original semantic affinities from different modalities to construct a high-order similarity matrix to guide the learning. Joint-modal Distribution-based Similarity Hashing (JDSH) [24] considers that there may exist redundant information in the aforementioned matrix and proposes a sampling and weighting scheme to further improve the reliability of the matrix. Unsupervised Knowledge Distillation UKD [14] trains a student model by leveraging knowledge from a teacher model on the same task. These deep methods consistently rely on deep and wide networks, leading to computation- and memory-consuming retrieval models. As a result, they are inapplicable to practical utilization, such as mobile retrieval. In this paper, we consider solving such a problem by training a light model with high performance.

2.2 Knowledge Distillation

Knowledge Distillation (KD) is proposed to facilitate the learning of a model (i.e., the "student") by taking advantages of the learning experience from other models (i.e., the "teacher"). The concept is first introduced by [12], which achieves significant performance improvements of a single model by distilling the knowledge of an ensemble of deep neural nets.

Conventional KD methods typically follow the "teacher-student" rule, where the teacher is assumed to already exist and the distillation is conducted by directly matching predictions between the student and the teacher, such as logits [2] and output categorical probability [12]. To excavate more knowledge from the teacher, many methods have been proposed to exploit intermediate hidden layer activations [43], and mimic parameter flows [42]. These schemes are mainly designed for close-set learning tasks, where the teacher and the student are assumed to share the same label space. To cope with more complex scenarios, cross-task knowledge distillation has attracted much attention, which aims to get rid of the task constraint and bridge the gap between models on different tasks [13, 41]. Instead of the direct alignment between the label predictions of different models, cross-task KD realizes the knowledge transfer by exploiting data relationships, such as pair-wise distance [37], distribution [21] and affinity graph [25].

The aforementioned schemes require a pre-trained teacher in advance, which is inflexible and limited in practical uses. To tackle this issue, many solutions are given, including online KD [5, 48] and ensemble [22, 36]. In online KD, all networks are trained collaboratively with knowledge gained from each other, while ensemble approaches take the ensemble of predictions or models as the target for the student to align. For example, Deep Mutual Learning (DML) [48] trains a pool of students by restricting them to mimic the output of each other via the KL Divergence. Temporal ensembling [22] and mean-teacher model [36] respectively leverage exponential moving average predictions of samples and model weights at different training iterations as the teacher. These distillation methods focus on how to perfectly borrow the learning experience from the teacher, however ignoring the inevitable noises in the distilled knowledge. In this paper, we propose to eliminate the noises to provide confident guidance for learning.

3 METHODOLOGY

3.1 Overview

To enable the resource-constrained retrieval, we design a novel JOint-teachinG (JOG) paradigm. The core of the JOG is to learn a high-performing yet light-weight retrieval model by reusing the knowledge from a cross-task teacher. To bridge the task gap in the knowledge transfer, we build transferable knowledge items based on the feature embedding from the teacher to guide student learning. To eliminate the inevitable noises in the distilled knowledge, an online progressive knowledge refinery method is proposed to further improve the reliability of the cross-task knowledge via a joint teaching manner.

As shown in Figure 1, the learning consists of two phases, i.e., the initialization and the refinery. In detail, in the initial phase of the training, offline knowledge distillation is performed, where the knowledge from a cross-task teacher is utilized to initialize the student and a peer model (i.e., "Model1" and "Model2") separately. In the next phase, online knowledge refinery is conducted by collaboratively training the student and its peer, during which we maintain their exponential moving average (EMA) [36] models to progressively refinery the knowledge from the cross-task teacher.

3.2 Preliminaries

3.2.1 Notation. Assume the learning problem is defined on a multi-modal dataset with n image-text pairs, indicated as $O = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{n}$, where $\mathbf{x}_i$ and $\mathbf{y}_i$ denote the i-th image and text sample, respectively. The supervised information such as labels and similarity matrix is not available.

3.2.2 Cross-modal Hashing Learning Revisit. Cross-modal hashing aims to project heterogeneous data from the original spaces into a common Hamming space with the original similarity preserved. To this end, a common way is to construct a similarity matrix $\mathrm{S} \in [-1, +1]^{n \times n}$ to indicate the relationships between paired data instances $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^{n}$ to guide learning, where larger S_{ij} means that the i-th sample and j-th sample are more semantically similar. Denote the hashing networks for the image and text are $H_{\theta_{\mathrm{x}}}(\cdot) = h_{\mathrm{x}}(f_{\mathrm{x}}(\cdot))$ and $H_{\theta_{\mathrm{y}}}(\cdot) = h_{\mathrm{y}}(f_{\mathrm{y}}(\cdot))$, where θ_{x} and θ_{y} are network parameters. f_{x} and f_{y} are the feature encoder backbone to learn

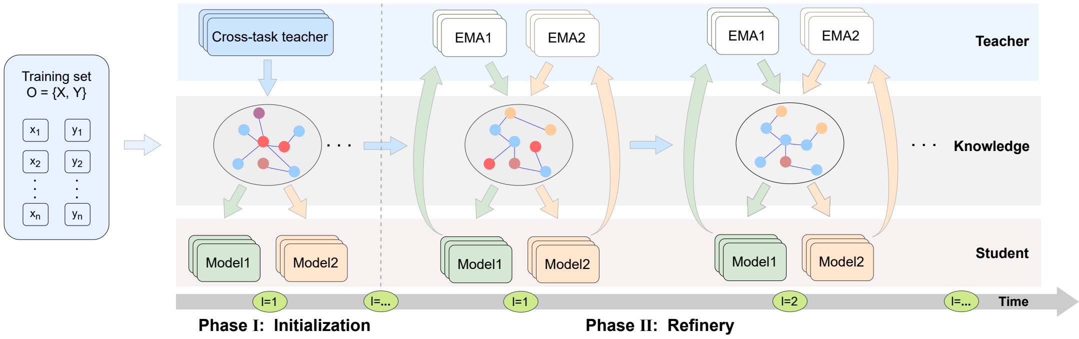

Figure 1: Illustration of the proposed JOG framework. The knowledge is distilled from a cross-task teacher and progressively refined to guide the learning of the student.

features, and $h_{\mathbf{x}}$ and $h_{\mathbf{y}}$ are the hashing layer to generate hash codes. For $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, their hash codes are obtained by applying the corresponding hashing networks, i.e., $\mathbf{B}_{\mathbf{x}_i} = sign(H_{\theta_{\mathbf{x}}}(\mathbf{x}_i))$ and $\mathbf{B}_{\mathbf{y}_i} = sign(H_{\theta_{\mathbf{y}}}(\mathbf{y}_i)) \in \{-1, +1\}^{n \times c}$, where c is the code length, and $sign(\cdot)$ is the sign function. A commonly-adopted objective for training the hashing networks is to restrict that the learned binary codes preserve the similarity defined in S by minimizing the following loss:

$$\mathcal{L}_{cmh} = \sum_{i,j=1}^{n} \|\mathrm{S}_{ij} - \mathcal{M}(\mathbf{B}_{\mathbf{x}_i}, \mathbf{B}_{\mathbf{y}_j})\|_F^2, \tag{1}$$

Where $\mathcal{M}$ measures the similarity between $\mathbf{B}_{\mathbf{x}_i}$ and $\mathbf{B}_{\mathbf{y}_i}$, e.g., cosine similarity. In the supervised scenarios, the similarity matrix S has already existed or can be easily constructed using labels. However, in the unsupervised scenes, it is not available. To tackle this, we plan to take advantage of the knowledge from a pre-trained model to exploit the underlying similarity structure to guide learning.

3.2.3 Knowledge Distillation Revisit. Distillation methods typically adopt a teacher-student learning pattern by reusing the knowledge from the teacher to facilitate the training of the student. Current KD methods are mainly designed for classification or recognition based tasks. Formally, denote the teacher model as $F_{\theta^t}(\cdot) = h^t(f^t(\cdot))$, where θ^t is the parameter, f^t and h^t are the feature encoder and the prediction layer, respectively. Similarly, the student is denoted as $F_{\theta^s}(\cdot) = h^s(f^s(\cdot))$. In a general pipeline, given a dataset $\mathbf{X} = \{\mathbf{x}_i\}_{i=1}^n$, the knowledge transfer from the teacher to the student can be realized by aligning their outputs:

$$\mathcal{L}_{kd} = \sum_{i=1}^{n} L_D\left(\mathcal{P}^t(\mathbf{x}_i), \mathcal{P}^s(\mathbf{x}_i)\right), \tag{2}$$

where L_D is a distillation loss, such as the Kullback-Leibler (KL) divergence [12]. $\mathcal{P}^t(\mathbf{x}_i) = softmax(F_{\theta^t}(\mathbf{x}_i)/\tau)$ and $\mathcal{P}^s(\mathbf{x}_i) = softmax(F_{\theta^s}(\mathbf{x}_i)/\tau)$ are the soften probability distribution for the teacher and student respectively, where τ is the temperature.

Note that in Eq. (2), the teacher and the student are required to share the same label space so that the knowledge can be directly transferred by matching the predictions between them. However, in the real world, the teacher serving the same task as the student is not always available, which means that the direct knowledge transition is no longer applicable. To tackle this, we propose the JOG to borrow the learning experience from a cross-task teacher to facilitate student learning.

3.3 Cross-task Knowledge Distillation

To bridge the task gap, we propose to build transferable knowledge items from the teacher to optimize the student from two perspectives, i.e., prediction-level distillation and embedding-level distillation. In prediction-level distillation, a similarity matrix is constructed to carry the knowledge from the teacher to realize the binary learning of the student. At the same time, an item that reflects the overall comparison information between data instances is learned to optimize the feature encoder of the student. Through the two-level optimization, the proposed method is expected to endow the student with both strong feature representing ability and predictive power, thereby benefiting the final results.

3.3.1 Prediction-level Distillation. As stated in Section 3.2.2, to learn binary codes, one key is to construct a reliable S. In light of this, the embeddings extracted from pre-trained models (a.k.a. the "teacher") are leveraged to measure the similarity as a transitive term. Formally, denote the teacher networks as $F_{\theta_{\mathbf{x}}^t}(\cdot) = h_{\mathbf{x}}^t(f_{\mathbf{x}}^t(\cdot))$ and $F_{\theta_{\mathbf{y}}^t}(\cdot) = h_{\mathbf{y}}^t(f_{\mathbf{y}}^t(\cdot))$ for image and text modality, respectively. Given training data $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, we extract their features from the corresponding teacher networks, denoted as $\{(f_{\mathbf{x}}^t(\mathbf{x}_i), f_{\mathbf{y}}^t(\mathbf{y}_i))\}_{i=1}^n$. As features from different networks are usually different in size, it is infeasible to directly measure the similarity between data across different modalities. To tackle this, we first measure the similarity between samples within each modality by the cosine distance:

$$[\mathrm{S}_*]_{ij} = cos\left(f_*^t(*_i), f_*^t(*_j)\right), \tag{3}$$

where $* \in \{\mathbf{x}, \mathbf{y}\}$ and $cos(\cdot)$ is the cosine distance. When $f_*^t(*)$ are non-negative, $\mathrm{S}_* \in [0, +1]^{n \times n}$ would be further processed as $\mathrm{S}_* = 2 \cdot \mathrm{S}_* - 1 \in [-1, +1]^{n \times n}$. In other cases, they are kept unchanged.

The similarity matrices from different modalities are further incorporated into produce a unified similarity matrix, so that the knowledge from different modalities would be well leveraged. Then, we have:

$$\mathrm{S} = \sum_{* \in \{\mathbf{x},\mathbf{y}\}} \mu_* \cdot \mathrm{S}_*, \quad s.t. \sum_{* \in \{\mathbf{x},\mathbf{y}\}} \mu_* = 1, \tag{4}$$

where $\mu_* \in [0, 1]$ is the balance parameter.

Denote the student networks for the image and text modality as $F_{\theta_{\mathbf{x}}^{s_1}}(\cdot) = h_{\mathbf{x}}^{s_1}(f_{\mathbf{x}}^{s_1}(\cdot))$ and $F_{\theta_{\mathbf{y}}^{s_1}}(\cdot) = h_{\mathbf{y}}^{s_1}(f_{\mathbf{y}}^{s_1}(\cdot))$, respectively. Both networks are light-weight. With the above knowledge, the student can be optimized by minimizing the following loss:

$$\mathcal{L}_p(\mathbf{S}, F_{\theta_{\mathbf{x}}^{s_1}}, F_{\theta_{\mathbf{y}}^{s_1}}) = \sum_{i,j=1}^{n} \underbrace{\|\mathbf{S}_{ij} - cos(\mathbf{B}_{\mathbf{x}_i}^{s_1}, \mathbf{B}_{\mathbf{y}_j}^{s_1})\|_F^2}_{\text{inter-modality}}$$
$$+ \beta(\underbrace{\sum_{i,j=1}^{n} \|\mathbf{S}_{ij} - cos(\mathbf{B}_{\mathbf{x}_i}^{s_1}, \mathbf{B}_{\mathbf{x}_j}^{s_1})\|_F^2 + \sum_{i,j=1}^{n} \|\mathbf{S}_{ij} - cos(\mathbf{B}_{\mathbf{y}_i}^{s_1}, \mathbf{B}_{\mathbf{y}_j}^{s_1})\|_F^2}_{\text{intra-modality}}), \tag{5}$$

where $\beta > 0$ is the balance parameter, $\mathbf{B}_{\mathbf{x}_i}^{s_1} = sign(F_{\theta_{\mathbf{x}}^{s_1}}(\mathbf{x}_i))$ and $\mathbf{B}_{\mathbf{y}_i}^{s_1} = sign(F_{\theta_{\mathbf{y}}^{s_1}}(\mathbf{y}_i)) \in \{-1, +1\}^c$. The above loss function is imposed to constrict the student to preserve both the inter- and intra-modal similarity between data, so that the similarity-preserving hashing codes and functions could be obtained.

3.3.2 Embedding-level Distillation. As indicated by [5, 9, 41], the feature encoding ability of a model has an important impact on its performance in the "downstream" task. In light of this, we further optimize the feature encoders of the student by aligning their output embeddings with those from teachers to improve the discriminative ability of the features. Considering the difference between the embeddings from the teacher and student in terms of the scale and size, we turn to measure the relative comparison information between data distances, which shows how the teacher differentiates the similar neighbors from dissimilar neighbors for each instance. Inspired by [38, 41], we introduce the overall triplet probability to exploit the overall relative relationships between an anchor and its similar (dissimilar) neighbors compared to its dissimilar (similar) neighbors, which is defined as follows:

$$\mathcal{P}_i^*(f) = \frac{\sum_{j=1}^{n} S_{ij}^* \cdot \exp\left(-\, dist\left(f(\mathbf{x}_i), f(\mathbf{x}_j)\right)/\tau\right)}{\sum_{j=1}^{n} \exp\left(-\, dist\left(f(\mathbf{x}_i), f(\mathbf{x}_j)\right)/\tau\right)}, \tag{6}$$

where $* \in \{+, -\}$, f is a feature encoder, $\mathbf{x}_i$ is a sample, $\tau > 0$ is the temperature for controlling the smoothness of the output. $S^+(S^-)$ indicates if two samples are similar (dissimilar), where in our work, $S_{ij}^+ = \begin{cases} 1 & S_{ij} > 0 \\ 0 & S_{ij} \le 0 \end{cases}, S_{ij}^- = \begin{cases} 1 & S_{ij} \le 0 \\ 0 & S_{ij} > 0 \end{cases}.$

Based on the triple relations, we define the Bernoulli distribution $\mathcal{P}_i(f) = [\mathcal{P}_i^+(f), \mathcal{P}_i^-(f)]$ as a transferable item. Then, the objective to optimize the embeddings of the student are formulated as follows:

$$\mathcal{L}_e(\mathcal{P}(f_{\mathbf{x}}^t, f_{\mathbf{y}}^t), \mathcal{P}(f_{\mathbf{x}}^{s_1}), \mathcal{P}(f_{\mathbf{y}}^{s_1})) =$$
$$\sum_{*\in\{\mathbf{x},\mathbf{y}\}} \sum_{i=1}^{n} KL(\mathcal{P}_i(f_{\mathbf{x}}^t, f_{\mathbf{y}}^t) \| \mathcal{P}_i(f_*^{s_1})), \tag{7}$$
$$s.t.\ \mathcal{P}_i(f_{\mathbf{x}}^t, f_{\mathbf{y}}^t) = \sum_{*\in\{\mathbf{x},\mathbf{y}\}} \mu_* \cdot \mathcal{P}_i(f_*^t),$$

where KL is the KL-divergence, μ_* are the same as ones used in Eq. (4). Through the above knowledge transfer, the student is expected to develop the competitive discriminative ability as the teacher. Besides, unlike [38, 41] which only select a similar neighbor and a dissimilar impostor for an anchor in the similarity comparison, we measure the overall relative relationships that takes all data into consideration, further facilitating the transfer processing.

Incorporating Eq. (5) and (7), we can realize the cross-task knowledge transfer by minimizing the following loss function:

$$\mathcal{L}_{ini}^{s_1} = \mathcal{L}_p(\mathbf{S}, F_{\theta_{\mathbf{y}}^{s_1}}, F_{\theta_{\mathbf{y}}^{s_1}}) + \lambda \cdot \mathcal{L}_e(\mathcal{P}(f_{\mathbf{x}}^t, f_{\mathbf{y}}^t), \mathcal{P}(f_{\mathbf{x}}^{s_1}), \mathcal{P}(f_{\mathbf{y}}^{s_1})), \tag{8}$$

where $\lambda > 0$ is the balance parameter.

These "dark knowledge" distilled from the teacher can roughly reflect the underlying data correlations, which are independent of the task and thus can be utilized as the transferrable knowledge items to advise the student to build discriminative feature encoding ability and learn similarity preserving binary codes. Nevertheless, as features from the teacher inevitably contain noises owing to their limited transferability and the imperfection of the teacher, the above introduced offline knowledge distillation cannot offer perfect guidance for learning.

3.4 Progressive Refinery

To alleviate the noises in the offline distilled knowledge, we propose to generate online knowledge in the training process to progressively improve the quality of the cross-task knowledge. To this end, we leverage the mutual teaching strategy [9, 36, 48] to obtain online knowledge by introducing a peer model and training it with the student collaboratively. The rationale behind is that during training, models could roughly capture the underlying data distribution even with noisy supervision, by predictions of which we can excavate extra and diverse knowledge. In a general pipeline of mutual teaching, the predictions of one model would be directly utilized to guide the learning of the other, which however may harm the performance of each other due to the output difference between them [10]. Instead, we propose to incorporate diverse knowledge from both models to regularize the knowledge from the cross-task teacher. As the knowledge from different models, i.e., the cross-task teacher, the student and the peer, are all leveraged to supervise the student, we call the proposed scheme as the "Joint-teaching". To gain high-quality knowledge, we maintain the exponential moving average (EMA) models [36] of the student and the peer to obtain stable predictions.

Formally, denote the peer networks as $F_{\theta_{\mathbf{x}}^{s_2}}(\cdot) = h_{\mathbf{x}}^{s_2}(f_{\mathbf{x}}^{s_2}(\cdot))$ and $F_{\theta_{\mathbf{y}}^{s_2}}(\cdot) = h_{\mathbf{y}}^{s_2}(f_{\mathbf{y}}^{s_2}(\cdot))$ for image and text modality, respectively. Before the refinery processing, the peer student would be first initialized as the same way with the student. Then, the student and the peer would be simultaneously trained by taking the same data batch but with different noises added as input. The parameters of the EMA models of the student and the peer at the iteration l are calculated by the following rule:

$$\begin{aligned} [\theta_{\mathbf{x}}^{s_*}]_l' &= \alpha[\theta_{\mathbf{x}}^{s_*}]_{l-1}' + (1-\alpha)[\theta_{\mathbf{x}}^{s_*}]_l, \\ [\theta_{\mathbf{y}}^{s_*}]_l' &= \alpha[\theta_{\mathbf{y}}^{s_*}]_{l-1}' + (1-\alpha)[\theta_{\mathbf{y}}^{s_*}]_l, \end{aligned} \tag{9}$$

where $* \in \{1, 2\}$, $\alpha \in [0, 1]$ is the ensembling momentum, $[\theta_{\mathbf{x}}^{s_*}]_l'$ and $[\theta_{\mathbf{y}}^{s_*}]_l'$ are exponential moving average (EMA) weights of the models at the iteration l and $[\theta_{\mathbf{x}}^{s_*}]_l$ and $[\theta_{\mathbf{y}}^{s_*}]_l$ are parameters of models at the iteration l.

Based on the average models, the dynamically generated knowledge would be distilled to refine the offline cross-task knowledge.

To be specific, for the prediction-level knowledge refinery, an online similarity matrix is constructed by directly calculating the similarity between data across different modalities:

$$\begin{aligned} [\mathbf{S}^{s_*}_{ij}]_l &= (cos(F_{[\theta^{s_*}_{\mathbf{x}}]'_l}(\mathbf{x}_i), F_{[\theta^{s_*}_{\mathbf{y}}]'_l}(\mathbf{y}_j)) \\ &\quad + cos(F_{[\theta^{s_*}_{\mathbf{y}}]'_l}(\mathbf{y}_i), F_{[\theta^{s_*}_{\mathbf{x}}]'_l}(\mathbf{x}_j)))/2, \\ s.t.\ &* \in \{1, 2\}. \end{aligned} \tag{10}$$

Then, the exploited similarity knowledge is incorporated together to refine the offline one:

$$\begin{aligned} [\mathbf{S}]_l &= \rho \cdot \mathbf{S} + (1-\rho) \cdot [\mathbf{S}^e]_l, \\ s.t.\ [\mathbf{S}^e]_l &= \eta \cdot [\mathbf{S}^{s_1}]_l + (1-\eta) \cdot [\mathbf{S}^{s_2}]_l, \end{aligned} \tag{11}$$

where ρ and $\eta \in [0, 1]$ are the balance parameters.

As for mitigating the noises in the embedding-level knowledge, we utilize the embeddings extracted from the EMA models. As the embeddings from the image and text networks may also differ from each other in size, we separately compute the Bernoulli distribution within each modality using approaches defined in Eq. (6) and (7) and then incorporate them together to optimize the original one:

$$\begin{aligned} [\mathcal{P}(f^t_{\mathbf{x}}, f^t_{\mathbf{y}})]_l &= \rho \cdot \mathcal{P}(f^t_{\mathbf{x}}, f^t_{\mathbf{y}}) + (1-\rho) \cdot [\mathcal{P}^e]_l, \\ s.t.\ [\mathcal{P}^e]_l &= \eta \cdot \mathcal{P}([f^{s_1}_{\mathbf{x}}]'_l, [f^{s_1}_{\mathbf{y}}]'_l) + (1-\eta) \cdot \mathcal{P}([f^{s_2}_{\mathbf{x}}]'_l, [f^{s_2}_{\mathbf{y}}]'_l), \end{aligned} \tag{12}$$

where $[f^{s_1}_{\mathbf{x}}]'_l, [f^{s_1}_{\mathbf{y}}]'_l, [f^{s_2}_{\mathbf{x}}]'_l, [f^{s_2}_{\mathbf{y}}]'_l$ are running average feature encoders. In both Eq. (11) and Eq. (12), ρ increases during the learning procedure, so that such a refinement scheme is named as the "progressive refinery". The reason for applying a progressive scheme is that during training, models gradually achieve better understanding of data and provides better insight into the data correlations, so that we gradually increase the weights of the online knowledge to achieve a stable and confident refinement.

With the progressively refined knowledge, the loss function for optimizing the student and its peer can be formulated as:

$$\begin{aligned} \mathcal{L}^{s_*}_{ref} &= \mathcal{L}_p([\mathbf{S}]_l, F_{\theta^{s_*}_{\mathbf{y}}}, F_{\theta^{s_*}_{\mathbf{y}}}) \\ &\quad + \lambda \cdot \mathcal{L}_e([\mathcal{P}(f^t_{\mathbf{x}}, f^t_{\mathbf{y}})]_l, \mathcal{P}(f^{s_*}_{\mathbf{x}}), \mathcal{P}(f^{s_*}_{\mathbf{y}})), \\ s.t.\ &* \in \{1, 2\}. \end{aligned} \tag{13}$$

4 OPTIMIZATION

The optimization procedures are summarized in **Algorithm 1**. The learning starts from initializing the student and the peer by borrowing the learning experience from the cross-task teacher. After the initialization, the relative reliable models would be returned, which would provide comparative confident predictions to facilitate the training in the next step. In the refinery stage, the student and the peer are trained collaboratively under the supervision of progressively refined knowledge by combining the offline cross-task knowledge and the online knowledge from the EMA models. With such a joint-teaching scheme, the proposed method is expected to remove the noises in the cross-task knowledge, thereby stabilizing the training and improving the final performance of the student. In addition, as the $sign(\cdot)$ function for producing binary codes would cause the intractable back-propagation gradient problem, during training, we replace it with the $tanh(\cdot)$ function, while keeping it unchanged for inference.

Algorithm 1: Joint-teaching Framework

```
Require: Training data $O = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, mini-batch size $m$, hash code length $c$, iteration times $T_1, T_2$, balance parameters $\mu_{\mathbf{x}}, \mu_{\mathbf{y}}, \alpha, \beta, \lambda, \rho, \eta, \tau$;
Require: Randomly initialize $\theta^{s_*}_{\mathbf{x}}, \theta^{s_*}_{\mathbf{y}}, * \in \{1, 2\}$;
// Initialization
for $l = 1 \rightarrow T_1$ do
    $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m \sim O$ // Sample mini-batch;
    $\mathbf{S} \leftarrow \{(f^t_{\mathbf{x}}(\mathbf{x}_i), f^t_{\mathbf{y}}(\mathbf{y}_i))\}_{i=1}^m$ // Construct the cross-task similarity matrix;
    $[\theta^{s_*}_{\mathbf{x}}]_{l+1} \leftarrow [\theta^{s_*}_{\mathbf{x}}]_l - \alpha_f \nabla_{[\theta^{s_*}_{\mathbf{x}}]_l} \mathcal{L}^{s_*}_{ini}$,
    $[\theta^{s_*}_{\mathbf{y}}]_{l+1} \leftarrow [\theta^{s_*}_{\mathbf{y}}]_l - \alpha_f \nabla_{[\theta^{s_*}_{\mathbf{y}}]_l} \mathcal{L}^{s_*}_{ini}, * \in \{1, 2\}$ // Update the parameters of the student and the peer;
end
$[\theta^{s_*}_{\mathbf{x}}]'_0 \leftarrow [\theta^{s_*}_{\mathbf{x}}]_{T_1}, [\theta^{s_*}_{\mathbf{y}}]'_0 \leftarrow [\theta^{s_*}_{\mathbf{y}}]_{T_1}$// Initialize Mean-Teacher ensemble model weights;
// Refinery
for $l = 1 \rightarrow T_2$ do
    $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^m \sim O$ // Sample mini-batch;
    $\mathbf{S} \leftarrow \{(f^t_{\mathbf{x}}(\mathbf{x}_i), f^t_{\mathbf{y}}(\mathbf{y}_i))\}_{i=1}^m$ // Construct the cross-task similarity matrix;
    $[\theta^{s_*}_{\mathbf{x}}]'_l \leftarrow \{[\theta^{s_*}_{\mathbf{x}}]'_{l-1}, [\theta^{s_*}_{\mathbf{x}}]_l\}$,
    $[\theta^{s_*}_{\mathbf{y}}]'_l \leftarrow \{[\theta^{s_*}_{\mathbf{y}}]'_{l-1}, [\theta^{s_*}_{\mathbf{y}}]_l\}, * \in \{1, 2\}$// Update Mean-Teacher ensemble model weights;
    $[\mathbf{S}^{s_*}]_l \leftarrow \{F_{[\theta^{s_*}_{\mathbf{x}}]'_l}(\mathbf{x}_i), F_{[\theta^{s_*}_{\mathbf{y}}]'_l}(\mathbf{y}_i)\}_{i=1}^m, * \in \{1, 2\}$ // Construct ensembling similarity matrices ;
    $[\mathbf{S}]_l \leftarrow \{\mathbf{S}, [\mathbf{S}^{s_1}]_l, [\mathbf{S}^{s_2}]_l\}$ // Refine similarity;
    $[\theta^{s_*}_{\mathbf{x}}]_{l+1} \leftarrow [\theta^{s_*}_{\mathbf{x}}]_l - \alpha_f \nabla_{[\theta^{s_*}_{\mathbf{x}}]_l} \mathcal{L}^{s_*}_{ref}$,
    $[\theta^{s_*}_{\mathbf{y}}]_{l+1} \leftarrow [\theta^{s_*}_{\mathbf{y}}]_l - \alpha_f \nabla_{[\theta^{s_*}_{\mathbf{y}}]_l} \mathcal{L}^{s_*}_{ref}, * \in \{1, 2\}$ // Update the network parameters;
end
```

5 EXPERIMENTS

5.1 Datasets

We evaluated the JOG on two widely-used cross-modal datasets: MIRFlickr-25K [15] and NUS-WIDE [4]. In detail, MIRFlickr-25K consists of 25,000 image-text pairs, where each pair is annotated at least one of the 24 unique concepts. NUS-WIDE is a real-world large-scale dataset with 269,648 image-text pairs. Each pair is labelled at least one semantic concept. 186,577 instances are selected from the 10 most frequently used concepts from the original dataset as the final experimental dataset. For a fair comparison with other retrieval methods, 2,000 samples are randomly selected from each dataset as the query set while the rest as the database. 5,000 data instances are further randomly pick out from the database as the training set.

5.2 Baselines and Evaluation Metric

Seven state-of-the-art baselines are chosen for comparison, including non-deep methods: CVH [20], IMH [34], CCQ [26], CMFH [8], and deep methods: DJSRH [35], JDSH [24] and UKD-US [14]. For fairness, for non-deep methods, we extract deep features from the teacher models for training and testing. While for deep methods,

Table 1: Results of all methods on MIRFlickr-25K. The best results for each category are shown in boldface.

Task	**Teacher** / **Student**	**AlexNet**				**VGG19**				**ResNet50**			
		MAP		NDCG		MAP		NDCG		MAP		NDCG	
		32 *bits*	64 *bits*	32 *bits*	64 *bits*	32 *bits*	64 *bits*	32 *bits*	64 *bits*	32 *bits*	64 *bits*	32 *bits*	64 *bits*
I2T	CVH	0.619	0.612	0.300	0.299	0.632	0.629	0.315	0.310	0.610	0.605	0.294	0.286
	IMH	0.608	0.605	0.308	0.292	0.639	0.635	0.304	0.305	0.639	0.625	0.334	0.322
	CCQ	0.564	0.573	0.269	0.275	0.592	0.566	0.283	0.286	0.645	0.654	0.302	0.313
	CMFH	0.570	0.553	0.298	0.265	0.566	0.585	0.305	0.268	0.553	0.579	0.303	0.311
	DJSRH(M)	0.736	0.729	0.408	0.392	0.744	0.746	0.406	0.406	0.700	0.705	0.366	0.369
	JDSH(M)	0.713	0.732	0.393	0.406	0.727	0.756	0.385	0.409	0.735	0.731	0.381	0.392
	UKD-US(M)	0.720	0.729	0.384	0.389	0.746	0.768	0.410	0.431	0.731	0.731	0.395	0.395
	JOG(M)	**0.762**	**0.755**	**0.422**	**0.413**	**0.779**	**0.789**	**0.436**	**0.451**	**0.735**	**0.770**	**0.399**	**0.413**
	DJSRH(S)	0.681	0.725	0.340	0.384	0.711	0.714	0.369	0.377	0.696	0.698	0.360	0.361
	JDSH(S)	0.658	0.684	0.333	0.364	0.689	0.662	0.357	0.336	0.602	0.679	0.282	0.348
	UKD-US(S)	**0.683**	0.702	**0.349**	0.364	0.690	0.715	0.352	0.380	**0.731**	0.711	0.391	0.370
	JOG(S)	0.679	**0.759**	0.344	**0.413**	**0.711**	**0.761**	**0.374**	**0.411**	0.727	**0.720**	**0.399**	**0.383**
T2I	CVH	0.625	0.618	0.308	0.304	0.645	0.642	0.322	0.321	0.617	0.613	0.306	0.305
	IMH	0.613	0.609	0.282	0.284	0.649	0.643	0.317	0.314	0.644	0.634	0.340	0.324
	CCQ	0.587	0.588	0.290	0.291	0.595	0.594	0.291	0.294	0.619	0.641	0.335	0.347
	CMFH	0.562	0.557	0.296	0.269	0.567	0.577	0.302	0.271	0.543	0.560	0.293	0.308
	DJSRH(M)	0.722	0.720	0.399	0.387	0.728	0.729	0.394	0.394	0.715	0.703	0.378	0.366
	JDSH(M)	0.702	0.705	0.378	0.387	0.687	0.738	0.363	0.401	0.703	0.728	0.362	0.386
	UKD-US(M)	0.701	0.711	0.368	0.374	0.729	0.758	0.401	0.424	0.730	0.724	0.394	0.389
	JOG(M)	**0.736**	**0.746**	**0.406**	**0.409**	**0.757**	**0.787**	**0.426**	**0.451**	**0.740**	**0.738**	**0.405**	**0.422**
	DJSRH(S)	0.653	0.651	0.323	0.352	0.683	0.695	0.347	0.358	0.684	0.683	0.348	0.347
	JDSH(S)	0.623	0.660	0.305	0.334	0.659	0.645	0.306	0.319	0.589	0.638	0.280	0.309
	UKD-US(S)	0.657	0.687	0.324	0.340	0.675	0.694	0.338	0.359	**0.711**	0.695	**0.372**	0.354
	JOG(S)	**0.703**	**0.739**	**0.366**	**0.409**	**0.700**	**0.728**	**0.372**	**0.401**	0.687	**0.715**	0.346	**0.381**

we equip them with the same deep backbone as our student uses. In particular, for UKD-US, we take DJSRH as the teacher, where the same backbone as our teacher adopts is applied.

For evaluation, two widely-used metrics are chosen, i.e., Mean Average Precision (MAP) and Normalized Discounted Cumulative Gain (NDCG), to reflect both ranking information and precision.

5.3 Implementation Details

To comprehensively test the superiority of the proposed method, we perform experiments by adopting different CNN structures as the backbone of the student and the teacher. More specifically, on the image modality, we choose widely-used light models including MobileNet-V2 [31] and ShuffleNet-V2 [29] as the candidate by replacing the last layer of the original models with a fully connected layer (fc) with c hidden units. On the text, the Text CNN [18], a simple and light model is adopted by removing the last layer and adding a fc layer with the c hidden units. For the teacher, we choose the AlexNet [19], VGG19 [33] and ResNet50 [11] pre-trained on the Imagenet dataset [7] as the candidate on the image modal, while the hand-craft model, i.e., BoW [30], is take as the teacher on the text. Essentially, other deep and non-deep methods can also be used as the teacher as long as they can provide useful features. The peer adopts different structures from the student to provide diverse information. Specifically, it takes the AlexNet and TextCNN as the backbone of the image and text network, respectively.

The proposed JOG method is implemented with PyTorch on a workstation (with Intel XEON E5-2650 v3 @ 2.60GHz CPU, NVIDIA 1080Ti GPU). With regard to the parameter settings, we fix the batch size to 32, weight decay to $5e^{-4}$, set momentum as 0.9 and learning rates as 0.01. $\alpha = 0.999, \beta = 0.1, \lambda = 1000, \eta = 0.8, \tau = 10$, $\rho = \frac{p+l*n}{2*T_2*n}$, where n is the training size and p is related to the training progress in each epoch. On MIRFlickr-25K, $\mu_{\mathbf{x}} = 0.9, \mu_{\mathbf{y}} = 0.1, T_1 = 40, T_2 = 160$ while $\mu_{\mathbf{x}} = 0.6, \mu_{\mathbf{y}} = 0.4, T_1 = 40, T_2 = 260$ on NUS-WIDE. The compared methods take the same iteration times. With respect to the MAP and NDCG evaluations, the number of retrieved points is set to 500.

5.3.1 *Results and Analysis.*

The proposed JOG is compared with all baselines in the "I2T" (i.e., image query text) and "T2I" (i.e., text query image) search tasks with different backbones and teachers. Considering the space limitation, we only report the MAP values of 32 bits and 64 bits in Table 1 and 2. In the tables, for brevity, we use "(M)" and "(S)" to respectively represent that JOG is constructed on MobileNet-V2 and ShuffleNet-V2. From the results, we can have the following observations. First, the proposed JOG achieves the best overall performance over all compared methods. In terms of quantitative analysis, on MIRFlickr-25K, in the "I2T" task, JOG achieves an average 2.6 % increase while in the "T2I" task, JOG exceeds the compared methods by average 3.1 % on MAP. In terms of NDCG, the improvements in both tasks are 3.6 % and 6.7 %, respectively. On NUS-WIDE, the proposed method outperforms other methods by 2.1 % and 2.5 % on MAP and NDCG, respectively. The promising results imply that the proposed method can achieve more relevant samples given a query, which verifies the superiority of the proposed methods in cross-modal retrieval. Second, it is worth noting that in a few cases, the competitor UKD-US outperforms the proposed JOG. This is because that UKD-US takes a cross-modal

Table 2: Results of all methods on NUS-WIDE. The best results for each category are shown in boldface.

Task	Teacher / Student	AlexNet				VGG19				ResNet50			
		MAP		NDCG		MAP		NDCG		MAP		NDCG	
		32 *bits*	64 *bits*	32 *bits*	64 *bits*	32 *bits*	64 *bits*	32 *bits*	64 *bits*	32 *bits*	64 *bits*	32 *bits*	64 *bits*
I2T	CVH	0.430	0.425	0.260	0.258	0.462	0.458	0.254	0.253	0.426	0.419	0.253	0.254
	IMH	0.429	0.423	0.255	0.248	0.439	0.433	0.274	0.264	0.439	0.430	0.273	0.264
	CCQ	0.357	0.368	0.217	0.227	0.415	0.403	0.242	0.213	0.478	0.508	0.240	0.241
	CMFH	0.342	0.441	0.322	0.294	0.291	0.396	0.230	0.268	0.235	0.256	0.247	0.221
	DJSRH(M)	0.651	0.685	0.418	0.451	0.673	0.674	0.429	0.441	0.627	0.631	0.401	0.402
	JDSH(M)	0.668	0.652	0.428	0.415	0.664	0.681	0.426	0.439	0.636	0.642	0.416	0.405
	UKD-US(M)	0.639	0.660	0.405	0.431	0.703	0.707	0.459	0.467	**0.709**	0.671	**0.462**	0.433
	JOG(M)	**0.671**	**0.693**	**0.429**	**0.460**	**0.710**	**0.719**	**0.466**	**0.468**	0.700	**0.709**	0.448	**0.450**
	DJSRH(S)	0.596	0.637	0.372	0.410	0.600	0.610	0.378	0.382	0.595	0.581	0.369	0.363
	JDSH(S)	0.592	0.619	0.375	0.389	0.577	0.627	0.357	0.391	0.555	0.561	0.339	0.340
	UKD-US(S)	0.627	0.651	0.398	0.416	0.635	0.663	0.408	0.429	0.625	0.602	0.399	0.380
	JOG(S)	**0.644**	**0.668**	**0.420**	**0.440**	**0.659**	**0.670**	**0.418**	**0.439**	**0.644**	**0.651**	**0.408**	**0.419**
T2I	CVH	0.424	0.414	0.278	0.269	0.463	0.451	0.258	0.247	0.425	0.416	0.262	0.259
	IMH	0.429	0.420	0.247	0.258	0.397	0.448	0.264	0.266	0.448	0.433	0.253	0.251
	CCQ	0.392	0.372	0.244	0.257	0.432	0.398	0.245	0.241	0.451	0.480	0.272	0.270
	CMFH	0.334	0.428	0.313	0.290	0.279	0.385	0.235	0.267	0.236	0.263	0.247	0.222
	DJSRH(M)	0.645	0.665	0.419	**0.443**	0.674	0.672	0.444	0.453	0.632	0.641	0.409	0.420
	JDSH(M)	0.621	0.652	0.400	0.424	0.642	0.663	0.410	0.427	0.609	0.635	0.394	0.408
	UKD-US(M)	0.661	0.662	0.436	0.441	**0.698**	0.682	**0.469**	0.458	**0.706**	0.678	**0.474**	0.448
	JOG(M)	**0.688**	**0.668**	**0.454**	0.431	0.684	**0.700**	0.454	**0.460**	0.664	**0.690**	0.439	**0.458**
	DJSRH(S)	0.596	0.611	0.373	0.391	0.593	0.619	0.375	0.398	0.591	0.568	0.371	0.359
	JDSH(S)	0.568	0.620	0.358	0.390	0.601	0.611	0.380	0.386	0.473	0.572	0.281	0.357
	UKD-US(S)	0.622	0.624	0.395	0.403	0.649	0.680	0.417	0.451	0.623	0.595	0.393	0.379
	JOG(S)	**0.632**	**0.668**	**0.402**	**0.442**	**0.681**	**0.700**	**0.429**	**0.474**	**0.626**	**0.663**	**0.397**	**0.437**

Table 3: The ablation comparison on NUS-WIDE.

Task	Teacher / Student	AlexNet		VGG19		ResNet50	
		MAP	NDCG	MAP	NDCG	MAP	NDCG
I2T	JOG-1(M)	0.654	0.421	0.665	0.426	0.627	0.395
	JOG-2(M)	0.665	0.427	0.664	0.427	0.637	0.403
	JOG-3(M)	0.660	0.423	0.690	0.456	0.670	0.428
	JOG(M)	**0.671**	**0.429**	**0.710**	**0.466**	**0.700**	**0.448**
	JOG-1(S)	0.568	0.350	0.526	0.316	0.554	0.338
	JOG-2(S)	0.585	0.368	0.570	0.349	0.622	0.392
	JOG-3(S)	**0.663**	**0.433**	0.611	0.390	0.632	0.390
	JOG(S)	0.644	0.420	**0.659**	**0.418**	**0.644**	**0.408**
T2I	JOG-1(M)	0.640	0.423	0.654	0.423	0.647	0.421
	JOG-2(M)	0.641	0.424	0.643	0.417	0.647	0.423
	JOG-3(M)	0.666	0.449	**0.702**	**0.487**	**0.678**	**0.457**
	JOG(M)	**0.688**	**0.454**	0.684	0.454	0.664	0.439
	JOG-1(S)	0.540	0.329	0.501	0.301	0.544	0.330
	JOG-2(S)	0.587	0.372	0.566	0.347	0.608	0.383
	JOG-3(S)	0.623	0.397	0.610	0.386	0.601	0.371
	JOG(S)	**0.632**	**0.402**	**0.681**	**0.429**	**0.626**	**0.397**

retrieval model on the same task as the teacher, which naturally benefits the final results. Compared to UKD-US, JOG faces a more challenge cross-task scenario, where the large task discrepancy exists. Even so, it can be seen that JOG surpasses UKD-US in most cases, which well demonstrates the effectiveness of the proposed methods in terms of bridging the task gap in knowledge distillation.

5.3.2 Ablation Study. We attribute the success of the proposed JOG to the embedding-level distillation and the knowledge refinery. To investigate the contribution of them, we conduct extensive ablation studies on NUS-WIDE and report results of 32 bits in Table 3. “JOG-1” means the baseline model that does not use all the essential components but only the unified offline similarity matrix. “JOG-2” is a variant of JOG, which does not apply the embedding-level knowledge distillation. “JOG-3” represents a variant of the JOG that does not utilize the progressive refinery. From the results, we can see that the proposed strategies, i.e., the embedding-level distillation and knowledge refinery are both effective to improve the performance of a light model.

6 CONCLUSION

In this paper, we propose a novel cross-modal learning framework, JOint-teachinG (JOG), to enable the resource-limited retrieval by establishing a high-performing yet light-weight model. The proposed JOG achieves the purpose by borrowing the learning experience from a cross-task teacher to endow a light student with both reliable embedding ability and predictive power. To eliminate the noises in the distilled knowledge, an online refinement strategy is designed to progressively improve the cross-task knowledge. Considering the query speed, the proposed method learns to represent the original high-dimensional data with compact binary codes to accelerate the query. Extensive experiments demonstrate the effectiveness of the proposed method.

7 ACKNOWLEDGMENTS

This work is partially supported by Australian Research Council Discovery Project (ARC DP190102353, CE200100025), and China Scholarship Council.

REFERENCES

[1] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep canonical correlation analysis. In *Proceedings of International Conference on Machine Learning*. 1247–1255.

[2] Jimmy Ba and Rich Caruana. 2014. Do deep nets really need to be deep?. In *Proceedings of Advances in Neural Information Processing Systems*. 2654–2662.

[3] Yue Cao, Mingsheng Long, Jianmin Wang, Qiang Yang, and Philip S Yu. 2016. Deep visual-semantic hashing for cross-modal retrieval. In *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 1445–1454.

[4] Tat-Seng Chua, Jinhui Tang, Richang Hong, Haojie Li, Zhiping Luo, and Yantao Zheng. 2009. NUS-WIDE: a real-world web image database from national university of singapore. In *Proceedings of ACM International Conference on Image and Video Retrieval*. 1–9.

[5] Inseop Chung, SeongUk Park, Jangho Kim, and Nojun Kwak. 2020. Feature-map-level online adversarial knowledge distillation. In *Proceedings of International Conference on Machine Learning*. 2006–2015.

[6] Hui Cui, Lei Zhu, Jingjing Li, Yang Yang, and Liqiang Nie. 2019. Scalable deep hashing for large-scale social image retrieval. *IEEE Transactions on Image Processing* 29 (2019), 1271–1284.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: a large-scale hierarchical image database. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 248–255.

[8] Guiguang Ding, Yuchen Guo, and Jile Zhou. 2014. Collective matrix factorization hashing for multimodal data. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 2075–2082.

[9] Yixiao Ge, Dapeng Chen, and Hongsheng Li. 2020. Mutual mean-teaching: pseudo label refinery for unsupervised domain adaptation on person re-identification. *arXiv preprint arXiv:2001.01526* (2020).

[10] Qiushan Guo, Xinjiang Wang, Yichao Wu, Zhipeng Yu, Ding Liang, Xiaolin Hu, and Ping Luo. 2020. Online knowledge distillation via collaborative learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 11020–11029.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 770–778.

[12] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531* (2015).

[13] Yen-Chang Hsu, Zhaoyang Lv, and Zsolt Kira. 2017. Learning to cluster in order to transfer across domains and tasks. *arXiv preprint arXiv:1711.10125* (2017).

[14] Hengtong Hu, Lingxi Xie, Richang Hong, and Qi Tian. 2020. Creating something from nothing: unsupervised knowledge distillation for cross-modal hashing. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 3123–3132.

[15] Mark J Huiskes and Michael S Lew. 2008. The MIR flickr retrieval evaluation. In *Proceedings of ACM International Conference on Multimedia Information Retrieval*. 39–43.

[16] Qing-Yuan Jiang and Wu-Jun Li. 2017. Deep cross-modal hashing. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 3232–3240.

[17] Meina Kan, Shiguang Shan, Haihong Zhang, Shihong Lao, and Xilin Chen. 2015. Multi-view discriminant analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38, 1 (2015), 188–194.

[18] Yoon Kim. 2014. Convolutional neural networks for sentence classification. *arXiv preprint arXiv:1408.5882* (2014).

[19] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Proceedings of Advances in Neural Information Processing Systems*. 1097–1105.

[20] Shaishav Kumar and Raghavendra Udupa. 2011. Learning hash functions for cross-view similarity search. In *Proceedings of International Joint Conference on Artificial Intelligence*. 1360–1365.

[21] Jogendra Nath Kundu, Nishank Lakkakula, and R Venkatesh Babu. 2019. Um-adapt: unsupervised multi-task adaptation using adversarial cross-task distillation. In *Proceedings of IEEE International Conference on Computer Vision*. 1436–1445.

[22] Samuli Laine and Timo Aila. 2016. Temporal ensembling for semi-supervised learning. *arXiv preprint arXiv:1610.02242* (2016).

[23] Chuan-Xiang Li, Zhen-Duo Chen, Peng-Fei Zhang, Xin Luo, Liqiang Nie, Wei Zhang, and Xin-Shun Xu. 2018. SCRATCH: a scalable discrete matrix factorization hashing for cross-modal retrieval. In *Proceedings of ACM International Conference on Multimedia*. 1–9.

[24] Song Liu, Shengsheng Qian, Yang Guan, Jiawei Zhan, and Long Ying. 2020. Joint-modal distribution-based similarity hashing for large-scale unsupervised deep cross-modal retrieval. In *Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval*. 1379–1388.

[25] Yufan Liu, Jiajiong Cao, Bing Li, Chunfeng Yuan, Weiming Hu, Yangxi Li, and Yunqiang Duan. 2019. Knowledge distillation via instance relationship graph. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 7096–7104.

[26] Mingsheng Long, Yue Cao, Jianmin Wang, and Philip S Yu. 2016. Composite correlation quantization for efficient multimodal retrieval. In *Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval*. 579–588.

[27] Xu Lu, Lei Zhu, Zhiyong Cheng, Liqiang Nie, and Huaxiang Zhang. 2019. Online multi-modal hashing with dynamic query-adaption. In *Proceedings of ACM SIGIR International Conference on Research and Development in Information Retrieval*. 715–724.

[28] Yadan Luo, Yang Yang, Fumin Shen, Zi Huang, Pan Zhou, and Heng Tao Shen. 2018. Robust discrete code modeling for supervised hashing. *Pattern Recognition* 75 (2018), 128–135.

[29] Ningning Ma, Xiangyu Zhang, Hai-Tao Zheng, and Jian Sun. 2018. Shufflenet v2: practical guidelines for efficient cnn architecture design. In *Proceedings of European Conference on Computer Vision*. 116–131.

[30] Andrew Kachites McCallum. 1996. Bow: a toolkit for statistical language modeling, text retrieval, classification and clustering. (1996). http://www.cs.cmu.edu/ mccallum/bow.

[31] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. 2018. Mobilenetv2: inverted residuals and linear bottlenecks. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 4510–4520.

[32] Heng Tao Shen, Luchen Liu, Yang Yang, Xing Xu, Zi Huang, Fumin Shen, and Richang Hong. 2020. Exploiting subspace relation in semantic labels for cross-modal hashing. *IEEE Transactions on Knowledge and Data Engineering* (2020).

[33] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[34] Jingkuan Song, Yang Yang, Yi Yang, Zi Huang, and Heng Tao Shen. 2013. Inter-media hashing for large-scale retrieval from heterogeneous data sources. In *Proceedings of ACM SIGMOD International Conference on Management of Data*. 785–796.

[35] Shupeng Su, Zhisheng Zhong, and Chao Zhang. 2019. Deep joint-semantics reconstructing hashing for large-scale unsupervised cross-modal retrieval. In *Proceedings of IEEE International Conference on Computer Vision*. 3027–3035.

[36] Antti Tarvainen and Harri Valpola. 2017. Mean teachers are better role models: weight-averaged consistency targets improve semi-supervised deep learning results. In *Proceedings of Advances in Neural Information Processing Systems*. 1195–1204.

[37] Frederick Tung and Greg Mori. 2019. Similarity-preserving knowledge distillation. In *Proceedings of IEEE International Conference on Computer Vision*. 1365–1374.

[38] Laurens Van Der Maaten and Kilian Weinberger. 2012. Stochastic triplet embedding. In *Proceedings of IEEE International Workshop on Machine Learning for Signal Processing*. 1–6.

[39] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. 2017. Adversarial cross-modal retrieval. In *Proceedings of ACM International Conference on Multimedia*. 154–162.

[40] Weiran Wang, Raman Arora, Karen Livescu, and Jeff Bilmes. 2015. On deep multi-view representation learning. In *Proceedings of International Conference on Machine Learning*. 1083–1092.

[41] Han-Jia Ye, Su Lu, and De-Chuan Zhan. 2020. Distilling cross-task knowledge via relationship matching. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 12396–12405.

[42] Junho Yim, Donggyu Joo, Jihoon Bae, and Junmo Kim. 2017. A gift from knowledge distillation: fast optimization, network minimization and transfer learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 4133–4141.

[43] Sergey Zagoruyko and Nikos Komodakis. 2016. Paying more attention to attention: improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928* (2016).

[44] Xiaohua Zhai, Yuxin Peng, and Jianguo Xiao. 2013. Heterogeneous metric learning with joint graph regularization for cross-media retrieval. In *Proceedings of AAAI Conference on Artificial Intelligence*.

[45] Xiaohua Zhai, Yuxin Peng, and Jianguo Xiao. 2013. Learning cross-media joint representation with sparse and semisupervised regularization. *IEEE Transactions on Circuits and Systems for Video Technology* 24, 6 (2013), 965–978.

[46] Yibing Zhan, Jun Yu, Zhou Yu, Rong Zhang, Dacheng Tao, and Qi Tian. 2018. Comprehensive distance-preserving autoencoders for cross-modal retrieval. In *Proceedings of ACM International Conference on Multimedia*. 1137–1145.

[47] Peng-Fei Zhang, Chuan-Xiang Li, Meng-Yuan Liu, Liqiang Nie, and Xin-Shun Xu. 2017. Semi-relaxation supervised hashing for cross-modal retrieval. In *Proceedings of ACM International Conference on Multimedia*. 1762–1770.

[48] Ying Zhang, Tao Xiang, Timothy M Hospedales, and Huchuan Lu. 2018. Deep mutual learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 4320–4328.

[49] Liangli Zhen, Peng Hu, Xu Wang, and Dezhong Peng. 2019. Deep supervised cross-modal retrieval. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. 10394–10403.