LCM: A Surprisingly Effective Framework for Supervised Cross-modal Retrieval

Gohil Dwijesh*
Department of CSE, IIT Delhi
India
cs5170407@cse.iitd.ac.in

Shivangi Bithel*
Department of CSE, IIT Delhi
India
shivangi.bithel@cse.iitd.ac.in

Srikanta Bedathur
Department of CSE, IIT Delhi
India
srikanta@cse.iitd.ac.in

ABSTRACT

Due to its increasing importance, *cross-modal retrieval* (CMR), where the query from one modality is used to retrieve objects from a different modality, has gained a lot of attention. A plethora of techniques have been proposed for this task, with deep learnt multi-modal models being the dominant paradigm. While these techniques have become increasingly sophisticated in terms of learning representations of multi-modal objects in a common space, relatively less attention is paid to the overall computational costs involved while training the model and during retrieval.

In this work, we present LCM (Lightweight framework for Cross-Modal retrieval), a surprisingly effective approach with very low computational costs. It can work with any uni- and multi-modal representations that is available ranging from BoW/GIST to CLIP for text/image modality. In its training phase, LCM exploits the semantic labels with a combination of shallow modality-specific feed-forward network and a label auto-encoder such that embeddings in the common representation space that share labels are close to each other. During retrieval, LCM employs a novel 2-stage nearest neighbor (2Sknn) search to first rank candidate labels that are relevant to a query (stage-1), and then use this ranking to retrieve results from the indexed collection (stage-2). Experiments over 6 popular uni- and multi-label supervised CMR benchmarks show that LCM outperforms some of the very recent strong baselines by upto 20% gains in mAP values. Furthermore, we show that 2Sknn can benefit other baseline methods as well offering upto 50% mAP gains in some cases.

CCS CONCEPTS

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

KEYWORDS

Cross-modal retrieval, 2-Stage Retrieval, Representation Learning

ACM Reference Format:
Gohil Dwijesh, Shivangi Bithel, and Srikanta Bedathur. 2023. LCM: A Surprisingly Effective Framework for Supervised Cross-modal Retrieval. In *6th Joint International Conference on Data Science & Management of Data (10th ACM IKDD CODS and 28th COMAD) (CODS-COMAD 2023), January 4–7, 2023, Mumbai, India.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3570991.3571048

*Both authors contributed equally to this research.

CODS-COMAD 2023, January 4–7, 2023, Mumbai, India

ACM ISBN 978-1-4503-9797-1/23/01…$15.00
https://doi.org/10.1145/3570991.3571048

1 INTRODUCTION

The content produced today is increasingly multi-modal, with related images and text data produced in significant volumes. Consequently, it has become essential for retrieval systems to support cross-modal search. In cross-modal retrieval (CMR), a user can query the system in one modality –e.g., text query– and expect relevant results from another modality, such as image. In order to bridge the heterogeneity gap in cross-modal retrieval, a commonly used approach is to learn a common representation of objects from different modalities. In this common representation space, similar objects have similar representations, such that applying the nearest neighbor search will retrieve relevant cross-modal objects [11, 39].

Numerous techniques for cross-modal retrieval have been proposed ranging from deep learning-based models to those which learn hash representations for efficient indexing of cross-modal data [3, 10, 11, 16, 19, 36, 46–48, 52, 53]. Other approaches include the use of adversarial networks for learning modality invariant transformation of multi-modal data [16, 36, 44], learning of visual-semantic fine-grained features (e.g., word-level feature learning) using attention/transformers/RNN networks [2, 4, 15, 28], and more. Invariably most of these methods employ a large number of trainable parameters, with correspondingly high demand on computational resources, memory, training time, and, in some cases, inference time. Typically, the supervised cross-modal retrieval methods use label information to classify the learnt common representation into respective classes [11, 36, 44, 45, 52]. Only a few methods exploit the class information in a sophisticated way [10, 16]. They use an explicit label network to learn more discriminative label representations in the common space that guides the modality-specific common representations to preserve inter- and intra-modal similarity. Nearest neighbors are retrieved using *cosine*, *euclidean*, *Hamming* distances, or, in some cases, a custom nearest neighbor measure in the common space [49–51].

In this paper, we address the problem of *supervised cross-modal retrieval* and explore whether it is possible to achieve high quality retrieval *without* resorting to very costly models. We introduce a *Lightweight framework for Cross-Modal retrieval (LCM)* method that learns a lightweight non-linear transformation of embeddings into a shared space by optimizing the distance between embeddings with similar semantic relations. LCM utilizes an autoencoder to project semantic class labels into the common space and shallow feed forward networks for each modality to transform input embeddings to a common representation space. It uses a two-stage retrieval

method, inspired by the pseudo-relevance feedback techniques, called *2-Stage nearest neighbor (2Sknn)* search. In the first retrieval stage, 2Sknn efficiently retrieves an initial set of candidate results from this common space using scalable nearest neighbor indexes. The second stage uses the class label statistics on the candidates retrieved in the first stage to refine the retrieval and prioritize semantically relevant objects.

We conduct experiments using both uni-label (where objects are labeled as belonging to a single class) and multi-label benchmarks. An empirical comparison with recent state-of-the-art baselines such as DSCMR [52], CLIP4CMR [47] and PAN [48] shows that LCM is 9-23% better in mAP values than the closest baselines across the benchmark datasets. Finally, we also note that the idea of 2-stage retrieval can be used with other retrieval techniques to improve their retrieval performance by upto 50% in mAP scores.

LCM adopts the common representation learning objective of SRLCH [33] and improves over it significantly as follows: (a) we use lightweight modality-specific neural networks to transform the input features into the common space, (b) instead of using kernels to enforce non-linearity in transforming the class labels, we employ an auto-encoder network that can easily be used in both uni- and multi-label settings, and (c) we abandon the hashing based retrieval and instead use highly scalable nearest neighbor indexing technique (e.g., ScaNN [8] or FAISS[1]) without any loss in speed and better performance. Overall, the key contributions of the paper can be summarized as follows:

- We propose a novel lightweight LCM framework where we employ an autoencoder network for label projection in a common representation space to address the rarely touched issue of learning discriminative features for multi-label datasets.
- We propose a simple 2-Stage nearest neighbor (2Sknn) search technique that can be applied to any supervised cross-modal retrieval method that does representation learning in the common space.
- Experiments on 6 widely used cross-modal datasets demonstrate that our lightweight LCM framework achieves state-of-the-art results and is suitable for uni- and multi-label settings. Additionally, we exemplify the adaptability of the 2Sknn algorithm by applying it to recent baseline methods.

The rest of this paper is organized as follows. In Section 2, we review related work on cross-modal retrieval. We then introduce our LCM framework in Section 3. Section 4 presents the experimental setup and implementation details. Experiments are shown in Section 5. Finally, we conclude this paper in Section 6.

2 RELATED WORK

The cross-modal retrieval methods are designed to learn a common representation space in which the similarity of data from different modalities can be directly measured. Numerous ways to learn such a common representation space have been proposed in recent years. These methods can be broadly classified into two categories: 1) binary-valued representation learning, also known as cross-modal hashing, and 2) real-valued representation learning. Binary-valued representation learning methods aim to learn low-dimensional binary codes for high-dimensional input features such that the binary codes preserve inter- and intra-modal similarity. Traditional methods: [18, 33, 37, 45], in general, use the kernel trick and linear projection matrix to project the features into the common space. One commonly pointed out drawback of traditional methods is that they cannot exploit the non-linear correlations in the data. Owing to this, many deep hashing methods have also been proposed [3, 10, 16, 19, 46, 53].

The proposed method in this paper falls under real-valued representation learning methods. Real-valued methods aim to learn real-valued common representations for various modality items. Some representative methods are: [11, 36, 47, 48, 52]. Methods under this category can be further classified based on the information they utilize to learn the common representation space as 1) unsupervised methods and 2) supervised methods. The unsupervised methods [7, 13, 41], only utilize co-occurrence information to learn common representations across multi-modal data. Canonical Correlation Analysis (CCA) is one of the most popular and traditional unsupervised subspace learning methods. It learns a subspace in which the pairwise correlations between two sets of heterogeneous data are maximized [41]. KCCA [13] extends CCA by incorporating the kernel mappings. Despite the success of the unsupervised methods, labels provide more direct discriminative information. Hence, using labels can lead to a more discriminative common space. Supervised cross-modal retrieval approaches [24, 36, 47, 48, 52] use supervised semantic category information and enforce same/different category samples to have similar/dissimilar representations in the common space. Generalized multiview analysis (GMA)[32] and Multi-label CCA (ml-CCA) propose a supervised extension of the CCA method.

Beyond that, with advances in deep learning for representation learning, deep architectures have shown promising results in capturing non-linear relationships [14]. For example, Deep canonical correlation analysis (DCCA) [1] is proposed as a non-linear extension of CCA that incorporates DNN to learn the complex non-linear transformations for each modality. Similarly, SRLCH[33] is a traditional hashing method that uses the kernel trick and linear projection to learn a common space for images, text, and labels. To account for non-linearity, we propose a lightweight real-valued non-linear extension to the SRLCH method and show significant improvements over the current state-of-the-art baseline methods. Cross-media multiple deep network (CMDN) [21] also uses DNN to hierarchically combine the inter- and intra-modal representations to learn the rich cross-media correlation using a deeper two-level network strategy and finally get the shared representation using a stacked network style. Cross-modal correlation learning (CCL) [24] extends CMDN by using multi-grained fusion to learn more precise cross-modal correlation and multi-task learning strategies to adaptively balance intra-modality semantic category constraints and inter-modality pairwise similarity constraints. To learn the common space with semantic information, joint representation learning (JRL) [50] uses semi-supervised regularisation as well as sparse regularisation. Motivated by the success of autoencoder[9] networks in learning discriminative latent space, we utilize a lightweight autoencoder to project the labels into the common space. Correspondence Autoencoder (Corr-AE) [7] jointly incorporates representation learning and correlation learning errors into a single process and uses an autoencoder network per modality. In [36],

[1] https://faiss.ai

Adversarial Cross-Modal Retrieval (ACMR) method seeks an effective common subspace based on adversarial learning. In the adversarial learning-based CMR methods [36], the common space representations for image and text modality are given as input to the adversarial network. The idea is to fool the network in predicting the correct modality of the input representations. Such an approach helps learn the modality invariant features in the common space. Furthermore, Deep Supervised Cross-Modal Retrieval (DSCMR) [52] supervises the model in learning discriminative features by minimizing discrimination loss in both the label space and the common representation space. Prototype-based Adaptive Network (PAN) [48] learns a unified prototype for each semantic category and uses them as anchors to learn cross-modal representations. Additionally, CLIP4CMR [47] sheds light on the use of CLIP [29] as a representative of a vision-language pre-trained model for cross-modal retrieval under supervision. Using CLIP features enhances the common representation space's robustness to modality imbalance and sensitivity to dimension changes of the common representation space.

Some advanced DNN-based approaches use attention networks and transformers to learn fine-grained features for images and text [2, 4, 15, 28]. These methods assess image-sentence alignment for the MS-COCO-2014 and FLICKR30K datasets. Due to the lack of labels in these datasets, our approach cannot be trained on such datasets and no direct comparison is possible.

Most of the prior work uses the Cosine, Euclidean, or Hamming distance to find the nearest neighbors in the common space. However, a few methods come up with their own ways of finding nearest neighbors [49–51]. JGRHML [49] learns a metric to find the distance between two heterogeneous items, JRL [50] predicts whether the query and retrieved results belong to the same semantic category, and CDPAE [51] proposes a novel unsupervised similarity measure to calculate the distance between two representations using marginal probability. We propose a 2-Stage nearest neighbor search (2Sknn) method similar to these methods.

Only a few approaches use high-level label semantics in the common space to capture uni- and multi-label feature semantics. Furthermore, most methods use the naive method of finding the nearest neighbors in the common space. Also, the recent methods use complex non-linear models (attention, transformers, GANs, etc.) to learn a discriminative common space. In this work, we propose a novel lightweight framework that uses an autoencoder network for label projection to capture more meaningful representation for multi-label datasets and a simple yet effective 2-Stage nearest neighbor(2Sknn) search algorithm.

3 PROPOSED METHOD

Suppose we have n instances in our training dataset. Each instance is comprised of an image, a text, and corresponding label vector. Let's denote the image feature matrix as $X_v \in R^{n \times f_v}$, text feature matrix as $X_t \in R^{n \times f_t}$, and a label matrix as $Y \in \{0,1\}^{n \times C}$. Here f_v, f_t represents the feature dimension of image and text modality respectively and C represents the number of labels. Hence $(X_v^{(i)}, X_t^{(i)}, Y^{(i)})$ represents one training instance. In this paper, we will consider a query set and a retrieval set. We assume that for the query set, we do not have any label information available but for the retrieval set we know the ground truth labels. Similar to training dataset, we will represent the image and text feature matrices for the query set as Q_v and Q_t respectively. For the retrieval set, the image, text, and label matrices are represented as R_v, R_t, and Y_R.

Figure 1 shows the overall LCM framework. It comprises of two parts: Model training and retrieval. The model training first extracts features from the raw data. Then modality specific light-weight neural networks (3 layers each) learn the compressed representation for the corresponding input items. Finally, we utilize *pivots*, represented as B, which guide the dense representations to preserve inter- and intra-modal similarity in the common representation space. Once the model is trained, the retrieval happens in two stages. Given a query in one modality, the first stage retrieves items from the same modality and the second stage retrieves from other modality. The following sections elaborate more on the model training, the use of pivots and retrieval stages.

3.1 Common Representation Space Learning

In this framework, we will consider three types of neural networks. One for each modality and one for labels. We will call them F (image encoder), G (text encoder), and Z (label encoder). F and G are three layer feed forward neural networks. Z is an auto-encoder network. Let us refer to the encoder of Z as E and the decoder as D. The auto-encoder network aids in generating label projection vectors with greater discrimination. If the encoder E generates very different compressed representations for similar input vectors, it becomes difficult for the decoder to reconstruct the original input vectors. Hence, given a similar/dissimilar input to the encoder network, it will produce a similar/dissimilar compressed representation. In this framework, we focus on preserving inter- and intra-modal similarity using light-weight neural networks. The learning objective of the framework consists of three terms as follows:

$$\mathcal{L} = \mathcal{L}_{inter} + \mathcal{L}_{intra} + \mathcal{L}_{pivot} \tag{1}$$

3.1.1 Inter-modal loss ($\mathcal{L}_{inter}$). Inter-modal loss aims at minimizing the distance between the common space representation of semantically similar items that belong to different modality. We further break the inter-modal loss into two terms as follows:

$$\mathcal{L}_{inter} = \mathcal{L}_F + \mathcal{L}_G \tag{2}$$

$$\mathcal{L}_F = \frac{1}{n}\sum_{i=1}^{n} ||B^{(i)} - F(X_v)^{(i)}||^2 \tag{3}$$

$$\mathcal{L}_G = \frac{1}{n}\sum_{i=1}^{n} ||B^{(i)} - G(X_t)^{(i)}||^2 \tag{4}$$

Here, $\mathcal{L}_F$ and $\mathcal{L}_G$ corresponds to the loss function for neural network F and G respectively. We call $\mathcal{L}_F$ and $\mathcal{L}_G$ together the inter-modal loss term because it brings i^{th} common space representation of both image and text items closer via i^{th} pivot vector $B^{(i)}$ as can be seen in equations 3 and 4.

3.1.2 Intra-modal loss ($\mathcal{L}_{intra}$). Intra-modal loss aims at minimizing the distance between the common space representation of semantically similar items that belong to same modality. The L_{intra} is related to the loss terms corresponding to the auto-encoder network. The auto-encoder loss consists of two terms:

$$\mathcal{L}_{intra} = \mathcal{L}_{rec} + \sigma\mathcal{L}_{latent} \tag{5}$$

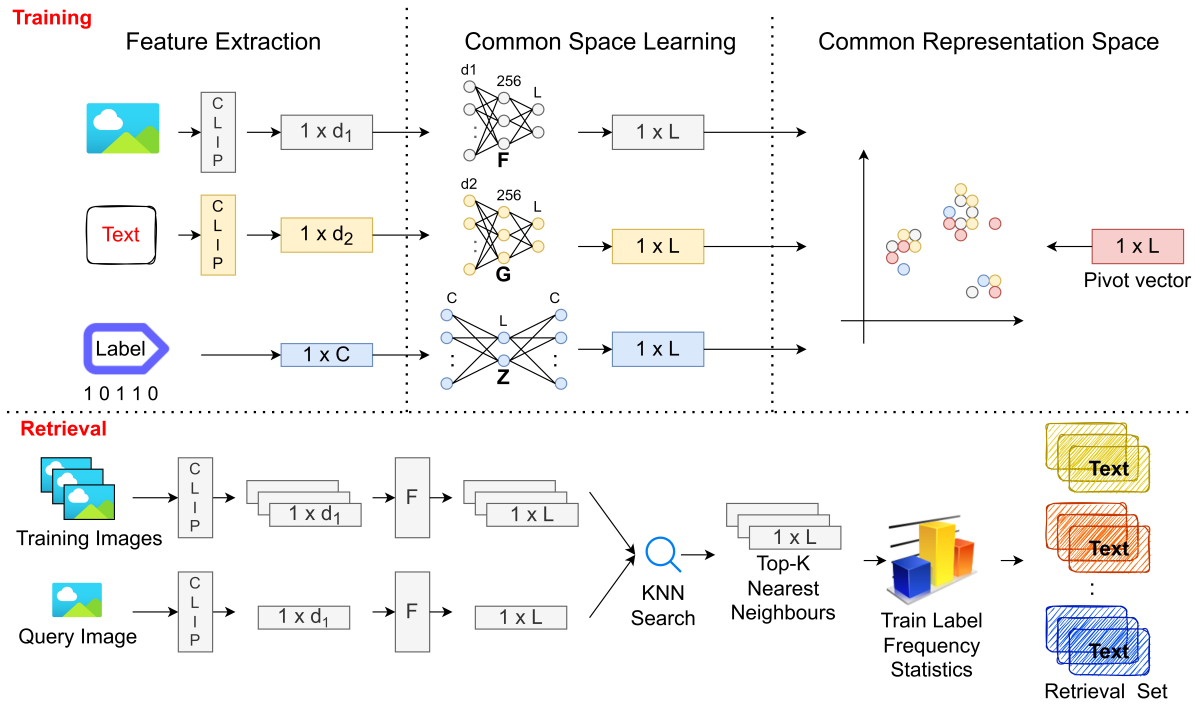


Figure 1: Framework for LCM. Using feed-forward neural networks F and G, we project images and text onto a shared space. Additionally, we project labels into shared space and attempt to minimise the distance between similar items within and across modalities. At inference time, we project the query into a common space, locate nearest neighbors in the same modality, rank labels according to their frequency, and lastly sort cross-modal items for recommendation.

Here σ is a hyper-parameter, $\mathcal{L}_{rec}$ is the reconstruction loss and $\mathcal{L}_{latent}$ is responsible to preserve the intra-modal similarity. The equations are defined as follows:

$$\mathcal{L}_{rec} = \frac{1}{n}\sum_{i=1}^{n}||\mathrm{D(E(Y))}^{(i)} - Y^{(i)}||^2$$

$$\mathcal{L}_{latent} = \frac{1}{n}\sum_{i=1}^{n}||B^{(i)} - E(Y)^{(i)}||^2$$

$\mathcal{L}_{latent}$ preserves the intra-modal similarity as follows. The encoder output of the auto-encoder network will be equal for items that have same label vectors. For such items, the corresponding pivot vectors will be forced to have the same value. As shown in equation-3 and 4, the similar pivot vectors will bring the corresponding intra-modal items together.

3.1.3 Pivot loss ($\mathcal{L}_{pivot}$). The pivot matrix consists of n pivot vectors. One pivot vector ($B^{(i)}$) is associated with one training instance ($\{X_v^{(i)}, X_t^{(i)}, Y^{(i)}\}$). The i^{th} pivot vector is a representative of the i^{th} training instance and they are expected to have similar representation in the common space. Pivot vectors guide training instances to preserve the inter- and intra-modal similarity. The pivot loss attempts to place the i^{th} pivot vector closer to i^{th} training instance as shown below:

$$\mathcal{L}_{pivot} = \left(a \mid\mid B - F(X_v) \mid\mid_F^2 + b \mid\mid B - G(X_t) \mid\mid_F^2 + c \mid\mid B - E(Y) \mid\mid_F^2\right)$$

Here a, b, and c are hyper-parameters. The optimization problem for the same is taken from SRLCH [33] paper and the direct solution for B is described below:

$$= \min_B \left(a \mid\mid B - F(X_v) \mid\mid_F^2 + b \mid\mid B - G(X_t) \mid\mid_F^2 + c \mid\mid B - E(Y) \mid\mid_F^2\right)$$

Now if we apply a constraint on B such that $B_{ij} \in \{-1, 1\}$ then the $Tr(B^T B)$ can be treated as constant. Also $F(X_v)^T F(X_v)$, $G(X_t)^T G(X_t)$, and $E(Y)^T E(Y)$ are constants. This leads to,

$$= \min_B \left(- Tr\left(B^T \left(aF\left(X_v\right) + bG\left(X_t\right) + cE\left(Y\right)\right)\right) - Tr\left(\left(aF\left(X_v\right)^T + bG\left(X_t\right)^T + cE\left(Y\right)^T\right) B\right)\right)$$

Using the property that $Tr(A^T B) = Tr(B^T A)$ for two $(m \times n)$ real matrices A and B

$$= \min_B \left(- 2Tr\left(B^T \left(aF\left(X_v\right) + bG\left(X_t\right) + cE\left(Y\right)\right)\right)\right)$$

Let $M = (aF(X_v) + bG(X_t) + cE(Y))$, where M is a $n \times L$ dimensional matrix:

$$= \min_B \left(- \left(\sum_{j=1}^{L}\left(\sum_{i=1}^{n} M_{ij} B_{ij}\right)\right)\right)$$

Given that $B_{ij} \in \{-1, 1\}$, the above expression is minimized when the product $M_{ij}B_{ij}$ is maximized and hence

$$B_{ij} = \text{sign}(M_{ij}) = \text{sign}(aF(X_v)_{ij} + bG(X_t)_{ij} + cE(Y)_{ij})$$

More generally,

$$B = \text{sign}(\ a\ F(X_v) + b\ G(X_t) + c\ E(Y)\ ) \tag{6}$$

In this framework, instead of an end-to-end training, we use alternate training where we update only one set of parameters and keep other parameters constant. We first train neural model F, then G and then Z using the back-propagation algorithm on loss terms $\mathcal{L}_F$, $\mathcal{L}_G$, and $\mathcal{L}_{intra}$ respectively. Then we update the pivots using the direct solution. We repeat the process until the total loss $\mathcal{L}$ in Equation 1 converges. For the sake of simplicity, we weigh each loss term equally.

3.2 2-Stage KNN search Algorithm (2Sknn)

We propose a simple yet effective search technique for cross-modal retrieval called 2-Stage nearest neighbor(2Sknn) search. It is a two stage retrieval process. In the initial stage, items from the same modality as the query are retrieved. Using the label statistics of the items retrieved in the first stage, the second stage retrieves items from a different modality. The detailed 2Sknn pipeline is depicted in the portion marked "Retrieval" of Figure 1

For a given query, the stage-1 retrieves training items from the same modality as the query. The retrieval process uses KNN search between the query and the training items. The KNN search can use any distance metric(Euclidean, Cosine, etc.). For example, consider image-to-text retrieval task. Let q_v be an image query. Now distance between the $F(q_v)$ and $\{F(X_v^{(i)})\ \ for\ \ all\ \ i\}$ is calculated and top-K training images with smallest distance value will be retrieved as 1^{st} stage retrieval items.

Since the first stage retrieval items belong to the train set, we know the corresponding ground truth labels. We will then calculate how many times a label occurred in the first stage retrieval items. All the labels are then sorted in descending order of label frequencies. For example, if there were two items in the retrieval set with label-2 and four items with label-5 then the ranking will be label-5 followed by label-2. As our task is multi-modal retrieval, we will collect all retrieval set items label-wise. In the above example, we will first collect all retrieval set items that belong to label-5, then collect all retrieval set items that belong to label-2. It is important to note that the collected items belong to the retrieval set whose labels are known and they are from the modality different than the query. So there will be as many collections as there are labels. These collections ordered in descending order of the corresponding label frequencies will be the recommendation of our framework. The pseudocode of the entire retrieval process is given in Algorithm 1.

Algorithm 1 LCM Retrieval (w.l.o.g., Image-to-Text Retrieval)

Input: q_v(=Image query), F, X_v, K
Output: $text\ \ recommendations$
$Normalize\ \ q_v$
Stage 1: KNN search
$top\_k\_image\_indices$ = arg sort([$dist(F(q_v), F(X_v^{(i)})\ \ for\ \ all\ \ i$])[:K]
$class\_freq$ = [0 $for\ \ i\ \ in\ \ range$(#$classes$)]
for (i=0; i<K; ++i) **do**
    $index = top\_k\_image\_indices[i]$
    for (j=0; j<#classes; ++j) **do**
        if $Y[index][j] == 1$ **then**
            $class\_freq[j]+ = 1$
        end if
    end for
end for
$ranked\_classes$ = arg sort($-class\_freq$)
Stage 2: Ranking retrieval set
$recommendations$ = [ ]
for (i=0; i<len(#classes); ++i) **do**
    $recommendations.add(texts \in ranked\_classes[i]\ \ and \notin recommendations)$
end for
return $recommendations$

4 EXPERIMENTAL SETUP

Datasets and Features:

To verify the efficacy of our proposed approach, we conducted experiments on six benchmark datasets, namely Wikipedia [31], Pascal-Sentence [30], NUS-WIDE-10K [5], XmediaNet [22, 26], MS-COCO [20] (2017 version), and MIRFlickr [12]. The dataset is divided into training, validation, retrieval, and query set. The retrieval set contains data from a different modality than the query set. We presume that the retrieval set contains labels, and the items do not need to be aligned to the query set. The statistical summary of the six datasets are summarised in Table 1.

Table 1: Statistics of datasets

Dataset	Classes	Train / Validation / Retrieval / Query
MS-COCO	90	117218 / 1500 / 2000 / 2000
MIRFlickr	24	17000 / 1359 / 2000 / 2000
Wikipedia	10	1942 / 215 / 462 / 462
Pascal Sentence	20	720 / 80 / 200 / 200
NUS-WIDE-10K	10	7200 / 800 / 2000 / 2000
XmediaNet	200	28800 / 3200 / 4000 / 4000

For the comparison with state-of-the-art baseline methods (Table-2 and 3), we represent our images and texts using CLIP[2] features. We follow the dataset partition and feature exaction strategies from CLIP4CMR [47] and take the mAP values for baseline methods on uni-label datasets from the CLIP4CMR [47] paper. Similarly, we follow the ALGCN [27] dataset partitioning scheme and take the mAP values for baseline methods on the multi-label dataset from the ALGCN [27] paper. For small datasets like Pascal Sentence and Wikipedia, end-to-end training cannot produce adequate unimodal representations. Thus for consistency in the paper, we are using pre-trained image and text features from CLIP for initializing all the datasets, and we see end-to-end training as the future for training large datasets.

Evaluation Metrics: In this paper, we consider image-to-text and text-to-image retrieval tasks. We use mAP to evaluate our method on various datasets. mAP value is the mean of all queries' average precision (AP). mAP is calculated over all retrieved results similar

[2]https://github.com/openai/CLIP

Table 2: Performance comparison in terms of mAP scores on four widely used uni-label datasets for cross-modal retrieval. △ denotes unsupervised methods. The bottom half of the table contains supervised methods.

Method	Wikipedia			Pascal Sentence			NUS-WIDE-10K			XmediaNet		
	I2T	T2I	Avg	I2T	T2I	Avg	I2T	T2I	Avg	I2T	T2I	Avg
CCA$^{\triangle}$ [41]	0.30	0.27	0.29	0.20	0.20	0.20	0.17	0.18	0.18	0.21	0.21	0.21
KCCA$^{\triangle}$ [13]	0.43	0.39	0.41	0.49	0.45	0.47	0.35	0.36	0.36	0.25	0.27	0.26
Corr-AE$^{\triangle}$ [7]	0.44	0.42	0.43	0.53	0.52	0.53	0.44	0.49	0.47	0.47	0.50	0.49
LCM$^{\triangle}$	**0.62**	**0.70**	**0.66**	**0.64**	**0.66**	**0.65**	**0.73**	**0.66**	**0.70**	**0.68**	**0.58**	**0.63**
CMDN [21]	0.49	0.43	0.46	0.54	0.53	0.54	0.49	0.54	0.52	0.49	0.52	0.51
JFSSL [38]	0.46	0.43	0.45	0.55	0.54	0.55	0.51	0.52	0.52	0.53	0.52	0.53
ACMR [36]	0.47	0.41	0.44	0.54	0.54	0.54	0.52	0.54	0.53	0.54	0.52	0.53
JLSLR [42]	0.47	0.44	0.46	0.57	0.55	0.56	0.54	0.53	0.54	0.54	0.55	0.55
MCMS [25]	0.52	0.46	0.49	0.60	0.60	0.60	0.52	0.55	0.54	0.54	0.55	0.55
CCL [24]	0.51	0.46	0.49	0.58	0.56	0.57	0.51	0.54	0.53	0.52	0.54	0.53
CM-GANS [23]	0.52	0.47	0.50	0.60	0.60	0.60	0.54	0.55	0.55	0.57	0.55	0.56
DSCMR [52]	0.52	0.49	0.51	0.67	0.67	0.67	0.56	0.59	0.58	0.64	0.65	0.65
PAN [48]	0.52	0.46	0.49	0.69	0.69	0.69	0.59	0.57	0.58	0.67	0.66	0.67
CLIP4CMR [47]	0.60	0.59	0.60	0.67	0.66	0.67	0.60	0.63	0.62	0.68	0.71	0.70
LCM	**0.65**	**0.84**	**0.75**	**0.74**	**0.78**	**0.76**	**0.81**	**0.71**	**0.76**	**0.84**	**0.75**	**0.80**

Table 3: Performance comparison in terms of mAP on multi-label MS-COCO dataset. △ denotes unsupervised methods. The bottom half of the table contains supervised methods.

Method	MS-COCO			MIRFlickr		
	I2T	T2I	Avg	I2T	T2I	Avg
CFA$^{\triangle}$ [17]	0.34	0.38	0.37	0.58	0.55	0.57
CCA$^{\triangle}$ [41]	0.65	0.66	0.66	0.71	0.72	0.72
Multimodal DBN$^{\triangle}$ [34]	0.36	0.33	0.35	0.58	0.56	0.57
Corr-AE$^{\triangle}$ [7]	0.65	0.67	0.66	0.71	0.73	0.72
DCCA$^{\triangle}$ [1]	0.64	0.63	0.64	0.74	0.75	0.75
LCM$^{\triangle}$	**0.87**	**0.86**	**0.87**	**0.74**	0.74	0.74
ml-CCA [6]	0.64	0.63	0.64	0.73	0.74	0.74
ACMR [36]	0.71	0.71	0.71	0.74	0.75	0.75
DCDH [40]	0.61	0.60	0.61	0.74	0.76	0.75
GCH [43]	0.56	0.56	0.56	0.76	0.79	0.78
DSCMR [52]	0.81	0.81	0.81	0.75	0.80	0.78
ALGCN [27]	0.84	0.83	0.84	0.80	0.82	0.81
CLIP4CMR [47]	0.77	0.78	0.78	0.72	0.74	0.73
LCM	**0.92**	**0.93**	**0.93**	**0.94**	**0.82**	**0.88**

to [47, 48, 52]. AP of a query is defined as follows:

$$AP = \frac{1}{T}\sum_{r=1}^{R} P_r \delta(r),$$

where R is the total number of items in the retrieval set and T is the number of relevant items in the retrieval set. P_r is precision of top-r retrieved items and $\delta(r)$ is an indicator function which takes value 1 if r^{th} retrieved item is relevant, otherwise it takes value 0. All presented mAP values for our method are averaged over three runs. For all the datasets, two items are considered as relevant if they share at least one label.

Implementation Details: In this method, for hyperparameters a, b, and c, we apply a grid search over the set $\{1, 10^{-1}, 10^{-2}, 10^{-3}\}$ and fix $a = 10^{-2}$, $b = 10^{-2}$, and $c = 10^{-1}$ for all datasets. For hyperparameter σ, we try values from set $\{10, 1, 10^{-1}, 10^{-2}, 10^{-3}\}$ and fix $\sigma = 10^{-2}$ for all dataset. Similarly, for K, we try values from the set $\{10, 25, 50, 75, 100, 200\}$ and fix $K = 50$ for all the datasets. We set $L = 32$ for all experiments.

The neural networks F and G are three-layer networks, with the first hidden layer having 256 neurons and a sigmoid activation unit. The second hidden layer consists of L neurons and a tanh activation unit. The first hidden layer uses batch normalization and dropout with a probability of 0.2. The encoder network of the autoencoder contains one layer with L neurons and a tanh activation unit, and the decoder network contains one layer with C neurons and a sigmoid activation unit. We use a learning rate of 10^{-3}, batch size of 64, 1000 maximum epochs, and a validation set for early stopping. We use the PyTorch framework and train our model on the CPU with an ADAM optimizer.

To initialize the pivot matrix, we tried both random and orthogonal initialization and found no significant performance difference. Hence, the pivot matrix is initialized using random samples from a uniform distribution over [0, 1) and we replace all values less than 0.5 with -1 and others with 1. Since our model learns real-valued common space, we use Euclidean distance to measure similarity.

5 EXPERIMENTAL RESULTS

To evaluate the performance of our proposed method, we first compare it to recent state-of-the-art methods in the experiments. Then, we provide additional analysis on the impact of various input features on the cross-modal retrieval task. Following that, we demonstrate the adaptability of the 2Sknn search. Finally, we provide a visualisation of naive similarity search and 2Sknn search in the learnt common representation space.

5.1 Comparison with Baselines

To demonstrate the efficacy of our proposed method, we compare our LCM with the results reported by several state-of-the-art methods. Overall, we consider the following baseline methods:

DSCMR[52], PAN[48], CLIP4CMR[47], ALGCN[27], GCH[43], CM-GANS[23], CCL[24], DCDH[40], MCMS [25], ACMR [36], JFSSL[38], CMDN[21], DCCA[1], Corr-AE[7], Multimodal DBN[34], ml-CCA[6], KCCA[13], and CCA[41]. A few of the baseline methods are unsupervised while the majority of the baseline methods are supervised. For fair comparison with the unsupervised methods, we consider a unsupervised variant of our method called LCM$^{\triangle}$. Further, we also present our comparison for uni-label datasets and for multi-label datasets separately as not all baselines were evaluated on both uni- and multi-label datasets.

Table 2 and Table 3 show the comparison of mAP values on image-to-text and text-to-image retrieval tasks on four uni-label datasets and two multi-label datasets for cross-modal retrieval respectively. In both, first half of the table contains unsupervised models, and the bottom half contains the supervised retrieval models. We can observe that LCM significantly outperforms all the baseline methods and is effective for both uni-label and multi-label datasets. Specifically, LCM outperforms the second-best baseline method with an improvement of 0.15, 0.07, 0.14, 0.1, 0.09, and 0.07 in terms of average mAP score on Wikipedia, Pascal Sentence, NUS-WIDE-10K, XMediaNet, MS-COCO, and MIRFlickr datasets, respectively.

We observe that there is a significant difference between the mAP values of I2T and T2I in Table-2 and 3 for our method. The reason is that the mAP values of LCM depend upon the first stage of our 2Sknn algorithm. The mean reciprocal rank (MRR) of the true class label in the ranked labels in stage-1 of the 2Sknn algorithm is 0.71/0.87, 0.77/0.78, 0.85/0.75, 0.86/0.77 for query images/texts from Wikipedia, Pascal Sentence, NUS-WIDE-10K, and XmediaNet respectively. Consequently, the difference in MRR between image and text queries results in a difference in the mAP values for the I2T and T2I tasks.

5.2 Comparison with Different Input Features

To provide insight into using the different input features for common representation space learning, we experimented with various combinations of feature types for images and texts. We experimented with GIST (1600D), Alexnet (512D), VGG19 (4096D), and CLIP (1024D) features for images and LDA, BoW, SBERT[3] (384D) and CLIP (1024D) features for text. For Wikipedia, Pascal Sentence, XmediaNet, MS-COCO and MIRFlickr datasets, we use 100D-1000D, 50D-512D, 256D-1000D, 256D-1000D, and 256D-1000D dimensional LDA-BoW features. For these experiments, we used the training set as the retrieval set. We tabulated our results in Table-4, where we highlighted bottom two mAP values with red and top two mAP values with green. Dark red represents the lowest and dark green represents the highest mAP.

We can observe that for all the datasets, image-to-text retrieval quality strongly correlates with the choice image feature type, *viz.*, CLIP, because the 2Sknn search first finds the nearest neighbors from the same modality. Hence, the model can learn superior intra-modal space using high-quality image representation, resulting in an accurate neighbor set and retrieval. Surprisingly, the choice of text feature has very little impact on the retrieval quality. Although a similar observation and reasoning holds for text-to-image retrieval,

[3]https://www.sbert.net/

Table 4: mAP values for different input feature combinations.

		Img-to-Txt				Txt-to-Img			
		lda	bow	sbert	clip	lda	bow	sbert	clip
MS-COCO	gist	0.64	0.64	0.65	0.64	0.92	0.93	0.94	0.93
	alexnet	0.79	0.79	0.80	0.80	0.92	0.93	0.93	0.93
	vgg19	0.84	0.85	0.85	0.85	0.92	0.93	0.93	0.93
	clip	0.91	0.91	0.91	0.92	0.92	0.93	0.93	0.93
MIRFlickr	gist	0.76	0.77	0.77	0.77	0.75	0.82	0.83	0.81
	alexnet	0.86	0.87	0.86	0.87	0.75	0.82	0.83	0.82
	vgg19	0.90	0.90	0.90	0.90	0.75	0.82	0.83	0.82
	clip	0.94	0.94	0.94	0.94	0.75	0.82	0.82	0.82
Pascal Sentence	gist	0.19	0.20	0.22	0.20	0.30	0.72	0.82	0.77
	alexnet	0.25	0.28	0.27	0.27	0.30	0.70	0.81	0.78
	vgg19	0.67	0.67	0.68	0.68	0.30	0.70	0.81	0.76
	clip	0.74	0.75	0.74	0.75	0.31	0.70	0.82	0.77
Wikipedia	gist	0.33	0.33	0.34	0.34	0.67	0.84	0.85	0.85
	alexnet	0.46	0.46	0.45	0.46	0.67	0.83	0.85	0.84
	vgg19	0.5	0.51	0.5	0.52	0.67	0.83	0.85	0.84
	clip	0.65	0.65	0.64	0.65	0.67	0.82	0.85	0.84
XmediaNet	gist	0.16	0.17	0.16	0.17	0.30	0.65	0.67	0.72
	alexnet	0.48	0.49	0.48	0.49	0.29	0.66	0.68	0.72
	vgg19	0.74	0.73	0.73	0.74	0.30	0.66	0.68	0.73
	clip	0.83	0.83	0.82	0.82	0.30	0.65	0.67	0.72

it is interesting to observe that for larger datasets like MIRFlickr and MS-COCO, the impact of text features is relatively low, except that LDA features are clearly a bad choice. On smaller Pascal Sentence the SBERT text features is better, while on XmediaNet, CLIP text features outperform SBERT features.

5.3 Applications of 2-Stage KNN Search

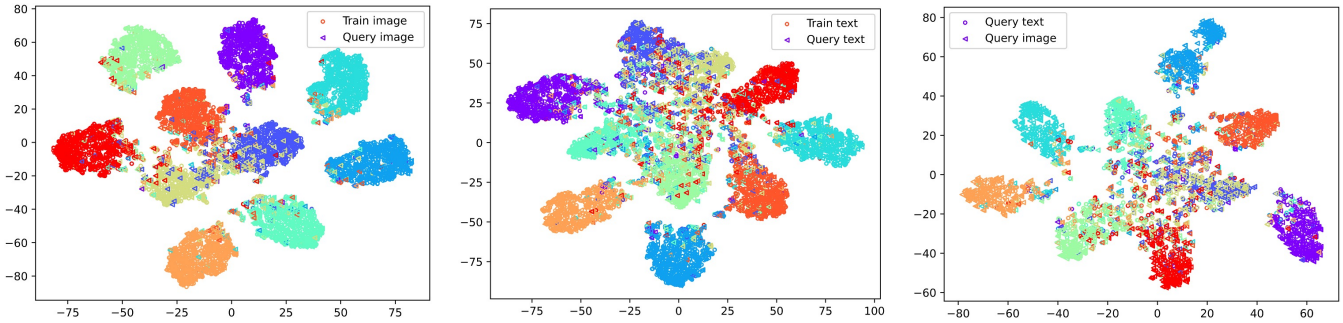
Our 2-Stage KNN Search method can be combined with any supervised common space learning. This section discusses the adaptiveness of the 2Sknn search on DSCMR, CLIP4CMR, and CLIP methods. For a fair comparison, we ran DSCMR[4], CLIP4CMR[5], CLIP, and our LCM using CLIP features and measured the mAP values before and after applying the 2Sknn search. Table-5 provides the MAP@all scores on MSCOCO, Pascal Sentence, NUS-WIDE-10K, Wikipedia, XmediaNet, and MIRFlickr datasets. Here, 2S-DSCMR, 2S-CLIP4CMR, 2S-CLIP and 2S-LCM represent corresponding methods with the 2Sknn search. The naive similarity search sorts the distance between the projected query and projected retrieval set items of other modality in increasing order to rank the retrieval set. At the same time, our 2Sknn search algorithm finds nearest neighbors from the same modality as that of the query in stage-1 and orders the relevant items according to the predicted class. We can observe an increase of 10%-20% when moving from naive similarity search to our proposed 2Sknn search on all retrieval benchmarks. It is probably because our proposed search method can sample correct labels for a query in stage-1 and order the relevant items at initial ranks. These enhancements demonstrate that the proposed 2Sknn can be used to improve the retrieval performance of a wide range of supervised CMR methods.

We compared the inference times of both the search algorithms and observed no significant difference. A query's average time to rank retrieval set items on Pascal Sentence, Wikipedia, NUS-WIDE-10K, and XmediaNet for naive similarity search is 0.00058, 0.0013, 0.0074, and 0.031, while 2Sknn search took 0.00046, 0.00047, 0.00094,

[4]https://github.com/penghu-cs/DSCMR
[5]https://github.com/zhixiongz/clip4cmr

Table 5: Performance comparison of methods with and without the 2-Stage knn search(2S-).

Method	MS-COCO			Pascal Sentence			NUS-WIDE-10K			Wikipedia			XmediaNet			MIRFlickr		
	I2T	T2I	Avg	I2T	T2I	Avg	I2T	T2I	Avg	I2T	T2I	Avg	I2T	T2I	Avg	I2T	T2I	Avg
DSCMR	0.81	0.82	0.82	0.70	0.70	0.70	0.60	0.59	0.60	0.62	0.60	0.61	0.73	0.73	0.73	0.75	0.80	0.78
2S-DSCMR	**0.92**	**0.92**	**0.92**	**0.82**	**0.82**	**0.82**	**0.84**	**0.73**	**0.79**	**0.72**	**0.88**	**0.80**	**0.90**	**0.79**	**0.85**	**0.93**	**0.82**	**0.88**
CLIP4CMR	0.77	0.78	0.78	0.67	0.66	0.67	0.60	0.63	0.62	0.60	0.59	0.60	0.68	0.71	0.70	0.72	0.74	0.73
2S-CLIP4CMR	**0.93**	**0.94**	**0.94**	**0.80**	**0.80**	**0.80**	**0.83**	**0.71**	**0.77**	**0.68**	**0.87**	**0.78**	**0.88**	**0.83**	**0.86**	**0.95**	**0.83**	**0.89**
CLIP	0.60	0.59	0.60	0.57	0.57	0.57	0.27	0.28	0.28	0.30	0.29	0.30	0.37	0.37	0.37	0.60	0.61	0.61
2S-CLIP	**0.91**	**0.93**	**0.92**	**0.69**	**0.76**	**0.73**	**0.74**	**0.68**	**0.71**	**0.63**	**0.81**	**0.72**	**0.84**	**0.83**	**0.84**	**0.93**	**0.83**	**0.88**
Naive-Our	0.67	0.67	0.67	0.60	0.61	0.61	0.58	0.59	0.59	0.55	0.52	0.54	0.54	0.50	0.52	0.66	0.65	0.66
2S-Our	**0.92**	**0.93**	**0.93**	**0.74**	**0.78**	**0.76**	**0.82**	**0.71**	**0.77**	**0.64**	**0.85**	**0.75**	**0.84**	**0.75**	**0.80**	**0.94**	**0.82**	**0.88**



(a) First stage of 2Sknn search between query image and train images in common representation space

(b) First stage of 2Sknn search between query text and train texts in common representation space

(c) Naive similarity search between query image and query text in common representation space

Figure 2: The visualisation of the similarity search algorithm on the NUS-WIDE-10K dataset using the t-SNE method [35]. The samples from the same semantic category are marked with the same colour.

and 0.028 seconds respectively. 2Sknn search uses ScaNN [8][6] for fast retrieval, and code for naive similarity search is used from DSCMR code.

5.4 Visualization using t-SNE plots

To visually understand the reason behind the surprising effectiveness of LCM, and 2Sknn search specifically, over naïve similarity search, we used t-SNE [35] plots. Figure 2 shows three different t-SNE plots with image and text embeddings in the common representation space for NUS-WIDE-10K data. In all these plots each class label is represented using a separate color. In Figure 2(a) (2(b)) we show the embeddings of all training images (text) —marked using circles— and the embeddings of *query images* (*query text*) obtained after passing them through the feed-forward network F (G) respectively. Figure 2(c) shows the embeddings of query images and text together [7].

Initial stage of 2Sknn search computes the nearest neighbors to the query from training items from *within the same* modality; their respective search spaces are depicted in Figures 2(a) and 2(b). As one can observe, there is a clear evidence for more queries to be located close to the clusters of training items from correct class. On the other hand, when we inspect Figure 2(c) which depicts the search space of naïve similarity search test-to-test cross-modal retrieval, fails to capture the correct class item from the other modality. In other words, the query image/text is frequently projected near the train image/text, whereas in Figure 2(c), many image and text samples are not falling in correct clusters. Consequently, 2Sknn search has a greater mAP than naive similarity search due to its effective nearest neighbor search.

[6]https://github.com/google-research/google-research/tree/master/scann

[7]Note that query items are nothing but elements of the test set from the benchmark.

6 CONCLUSION

In this paper, we proposed a lightweight framework called LCM to first learn a discriminative common representation space for uni-label and multi-label datasets using an autoencoder network for label projections. Subsequently, a 2-Stage nearest neighbor(2Sknn) search is used to get more than 9-23% improvements in retrieval performance over state of the art baselines across diverse multi-modal benchmark datasets. We note that 2Sknn method can be used independently with any supervised common space learning method to achieve more than 10-20% improvements in their mAP scores.

ACKNOWLEDGMENTS

This work was supported by a Huawei research grant, and a DS Chair of AI fellowship to Srikanta Bedathur.

REFERENCES

[1] Galen Andrew, Raman Arora, Jeff Bilmes, and Karen Livescu. 2013. Deep Canonical Correlation Analysis. In *Proceedings of the 30th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 28)*, Sanjoy

Dasgupta and David McAllester (Eds.). PMLR, Atlanta, Georgia, USA, 1247–1255. https://proceedings.mlr.press/v28/andrew13.html

[2] Jie Cao, Shengsheng Qian, Huaiwen Zhang, Quan Fang, and Changsheng Xu. 2021. Global Relation-Aware Attention Network for Image-Text Retrieval. *Proceedings of the 2021 International Conference on Multimedia Retrieval* (2021).

[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 the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (San Francisco, California, USA) *(KDD '16)*. Association for Computing Machinery, New York, NY, USA, 1445–1454. https://doi.org/10.1145/2939672.2939812

[4] Hui Chen, Guiguang Ding, Xudong Liu, Zijia Lin, Ji Liu, and Jungong Han. 2020. IMRAM: Iterative Matching With Recurrent Attention Memory for Cross-Modal Image-Text Retrieval. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2020), 12652–12660.

[5] 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 the ACM International Conference on Image and Video Retrieval* (Santorini, Fira, Greece) *(CIVR '09)*. Association for Computing Machinery, New York, NY, USA, Article 48, 9 pages. https://doi.org/10.1145/1646396.1646452

[6] B. Ding and Robert Gentleman. 2005. Classification Using Generalized Partial Least Squares. *Journal of Computational and Graphical Statistics* 14 (2005), 280 – 298.

[7] Fangxiang Feng, Xiaojie Wang, and Ruifan Li. 2014. Cross-Modal Retrieval with Correspondence Autoencoder. In *Proceedings of the 22nd ACM International Conference on Multimedia* (Orlando, Florida, USA) *(MM '14)*. Association for Computing Machinery, New York, NY, USA, 7–16. https://doi.org/10.1145/2647868.2654902

[8] Ruiqi Guo, Philip Sun, Erik Lindgren, Quan Geng, David Simcha, Felix Chern, and Sanjiv Kumar. 2020. Accelerating Large-Scale Inference with Anisotropic Vector Quantization. In *International Conference on Machine Learning*. https://arxiv.org/abs/1908.10396

[9] G. E. Hinton and R. R. Salakhutdinov. 2006. Reducing the Dimensionality of Data with Neural Networks. *Science* 313, 5786 (2006), 504–507. https://doi.org/10.1126/science.1127647 arXiv:https://www.science.org/doi/pdf/10.1126/science.1127647

[10] Peng Hu, Xu Wang, Liangli Zhen, and Dezhong Peng. 2019. Separated Variational Hashing Networks for Cross-Modal Retrieval. In *Proceedings of the 27th ACM International Conference on Multimedia* (Nice, France) *(MM '19)*. Association for Computing Machinery, New York, NY, USA, 1721–1729. https://doi.org/10.1145/3343031.3351078

[11] Peng Hu, Liangli Zhen, Dezhong Peng, and Pei Liu. 2019. Scalable Deep Multimodal Learning for Cross-Modal Retrieval. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Paris, France) *(SIGIR'19)*. Association for Computing Machinery, New York, NY, USA, 635–644. https://doi.org/10.1145/3331184.3331213

[12] Mark J. Huiskes and Michael S. Lew. 2008. The MIR Flickr Retrieval Evaluation. In *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval* (Vancouver, British Columbia, Canada) *(MIR '08)*. Association for Computing Machinery, New York, NY, USA, 39–43. https://doi.org/10.1145/1460096.1460104

[13] Pei Ling Lai and Colin Fyfe. 2000. Kernel and Nonlinear Canonical Correlation Analysis. *International journal of neural systems* 10 5 (2000), 365–77.

[14] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. 2015. Deep Learning. *Nature* (2015), 436–444. https://doi.org/10.1038/nature14539

[15] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. 2018. Stacked Cross Attention for Image-Text Matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[16] Chao Li, Cheng Deng, Ning Li, Wei Liu, Xinbo Gao, and Dacheng Tao. 2018. Self-supervised adversarial hashing networks for cross-modal retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4242–4251.

[17] Dongge Li, Nevenka Dimitrova, Mingkun Li, and Ishwar K. Sethi. 2003. Multimedia content processing through cross-modal association. In *MULTIMEDIA '03*.

[18] Kai Li, Guo-Jun Qi, Jun Ye, and Kien A. Hua. 2017. Linear Subspace Ranking Hashing for Cross-Modal Retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 9 (2017), 1825–1838. https://doi.org/10.1109/TPAMI.2016.2610969

[19] Zechao Li, Lu Jin, and Jinhui Tang. 2019. Deep Semantic Multimodal Hashing Network for Scalable Multimedia Retrieval. arXiv:1901.02662 [cs.CV]

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, and Piotr Dollár. 2015. Microsoft COCO: Common Objects in Context. arXiv:1405.0312 [cs.CV]

[21] Yuxin Peng, Xin Huang, and Jinwei Qi. 2016. Cross-Media Shared Representation by Hierarchical Learning with Multiple Deep Networks. In *IJCAI*.

[22] Yuxin Peng, Xin Huang, and Yunzhen Zhao. 2018. An Overview of Cross-Media Retrieval: Concepts, Methodologies, Benchmarks, and Challenges. *IEEE Trans. Cir. and Sys. for Video Technol.* 28, 9 (sep 2018), 2372–2385. https://doi.org/10.1109/TCSVT.2017.2705068

[23] Yuxin Peng and Jinwei Qi. 2019. CM-GANs: Cross-Modal Generative Adversarial Networks for Common Representation Learning. *ACM Trans. Multimedia Comput. Commun. Appl.* 15, 1, Article 22 (feb 2019), 24 pages. https://doi.org/10.1145/3284750

[24] Yuxin Peng, Jinwei Qi, Xin Huang, and Yuxin Yuan. 2018. CCL: Cross-modal Correlation Learning With Multigrained Fusion by Hierarchical Network. *IEEE Transactions on Multimedia* 20 (2018), 405–420.

[25] Yuxin Peng, Jinwei Qi, and Yuxin Yuan. 2018. Modality-Specific Cross-Modal Similarity Measurement With Recurrent Attention Network. *IEEE Transactions on Image Processing* 27 (2018), 5585–5599.

[26] Yuxin Peng, Jinwei Qi, and Yuxin Yuan. 2018. Modality-Specific Cross-Modal Similarity Measurement With Recurrent Attention Network. *Trans. Img. Proc.* 27, 11 (nov 2018), 5585–5599. https://doi.org/10.1109/TIP.2018.2852503

[27] Shengsheng Qian, Dizhan Xue, Quan Fang, and Changsheng Xu. 2021. Adaptive Label-aware Graph Convolutional Networks for Cross-Modal Retrieval. *IEEE Transactions on Multimedia* (2021), 1–1. https://doi.org/10.1109/TMM.2021.3101642

[28] Leigang Qu, Meng Liu, Jianlong Wu, Zan Gao, and Liqiang Nie. 2021. Dynamic Modality Interaction Modeling for Image-Text Retrieval. *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2021).

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021. Learning Transferable Visual Models From Natural Language Supervision. arXiv:2103.00020 [cs.CV]

[30] Cyrus Rashtchian, Peter Young, Micah Hodosh, and Julia Hockenmaier. 2010. Collecting Image Annotations Using Amazon's Mechanical Turk. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk* (Los Angeles, California) *(CSLDAMT '10)*. Association for Computational Linguistics, USA, 139–147.

[31] N. Rasiwasia, J. Costa Pereira, E. Coviello, G. Doyle, G.R.G. Lanckriet, R. Levy, and N. Vasconcelos. 2010. A New Approach to Cross-Modal Multimedia Retrieval. In *ACM International Conference on Multimedia*. 251–260.

[32] Abhishek Sharma, Abhishek Kumar, Hal Daume, and David W. Jacobs. 2012. Generalized Multiview Analysis: A discriminative latent space. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*. 2160–2167. https://doi.org/10.1109/CVPR.2012.6247923

[33] Heng Tao Shen, Luchen Liu, Yang Yang, Xing Xu, Zi Huang, Fumin Shen, and Richang Hong. 2021. Exploiting Subspace Relation in Semantic Labels for Cross-Modal Hashing. *IEEE Transactions on Knowledge and Data Engineering* 33, 10 (2021), 3351–3365. https://doi.org/10.1109/TKDE.2020.2970050

[34] Nitish Srivastava and Ruslan Salakhutdinov. 2012. Learning representations for multimodal data with deep belief nets. In *International conference on machine learning workshop*, Vol. 79. 3.

[35] Laurens van der Maaten and Geoffrey E. Hinton. 2008. Visualizing Data using t-SNE. *Journal of Machine Learning Research* 9 (2008), 2579–2605.

[36] Bokun Wang, Yang Yang, Xing Xu, Alan Hanjalic, and Heng Tao Shen. 2017. Adversarial Cross-Modal Retrieval. In *Proceedings of the 25th ACM International Conference on Multimedia* (Mountain View, California, USA) *(MM '17)*. Association for Computing Machinery, New York, NY, USA, 154–162. https://doi.org/10.1145/3123266.3123326

[37] Di Wang, Quan Wang, Yaqiang An, Xinbo Gao, and Yumin Tian. 2020. Online Collective Matrix Factorization Hashing for Large-Scale Cross-Media Retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval* (Virtual Event, China) *(SIGIR '20)*. Association for Computing Machinery, New York, NY, USA, 1409–1418. https://doi.org/10.1145/3397271.3401132

[38] K. Wang, Ran He, Liang Wang, Wei Wang, and Tieniu Tan. 2016. Joint Feature Selection and Subspace Learning for Cross-Modal Retrieval. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 38 (2016), 2010–2023.

[39] Kaiye Wang, Qiyue Yin, Wei Wang, Shu Wu, and Liang Wang. 2016. A Comprehensive Survey on Cross-modal Retrieval. arXiv:1607.06215 [cs.MM]

[40] Zijian Wang, Zheng Zhang, Yadan Luo, Zi Huang, and Heng Tao Shen. 2021. Deep Collaborative Discrete Hashing With Semantic-Invariant Structure Construction. *IEEE Transactions on Multimedia* 23 (2021), 1274–1286.

[41] David Weenink. 2003. Canonical correlation analysis. In *Proceedings of the Institute of Phonetic Sciences of the University of Amsterdam*, Vol. 25. Citeseer, 81–99.

[42] Jianlong Wu, Zhouchen Lin, and Hongbin Zha. 2017. Joint Latent Subspace Learning and Regression for Cross-Modal Retrieval. *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval* (2017).

[43] Ruiqing Xu, Chao Li, Junchi Yan, Cheng Deng, and Xianglong Liu. 2019. Graph Convolutional Network Hashing for Cross-Modal Retrieval. In *IJCAI*.

[44] Xing Xu, Li He, Huimin Lu, Lianli Gao, and Yanli Ji. 2019. Deep Adversarial Metric Learning for Cross-Modal Retrieval. *World Wide Web* 22, 2 (mar 2019), 657–672. https://doi.org/10.1007/s11280-018-0541-x

[45] Xing Xu, Fumin Shen, Yang Yang, Heng Tao Shen, and Xuelong Li. 2017. Learning Discriminative Binary Codes for Large-scale Cross-modal Retrieval. *IEEE Transactions on Image Processing* 26, 5 (2017), 2494–2507. https://doi.org/10.1109/TIP.2017.2676345

[46] Erkun Yang, Cheng Deng, Wei Liu, Xianglong Liu, Dacheng Tao, and Xinbo Gao. 2017. Pairwise Relationship Guided Deep Hashing for Cross-Modal Retrieval. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence* (San Francisco, California, USA) *(AAAI'17)*. AAAI Press, 1618–1625.

[47] Zhixiong Zeng and Wenji Mao. 2022. A Comprehensive Empirical Study of Vision-Language Pre-trained Model for Supervised Cross-Modal Retrieval. arXiv:2201.02772 [cs.CV]

[48] Zhixiong Zeng, Shuai Wang, Nan Xu, and Wenji Mao. 2021. *PAN: Prototype-Based Adaptive Network for Robust Cross-Modal Retrieval.* Association for Computing Machinery, New York, NY, USA, 1125–1134. https://doi.org/10.1145/3404835.3462867

[49] Xiaohua Zhai, Yuxin Peng, and Jianguo Xiao. 2013. Heterogeneous Metric Learning with Joint Graph Regularization for Cross-Media Retrieval. In *Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence* (Bellevue, Washington) *(AAAI'13)*. AAAI Press, 1198–1204.

[50] Xiaohua Zhai, Yuxin Peng, and Jianguo Xiao. 2014. Learning Cross-Media Joint Representation With Sparse and Semisupervised Regularization. *IEEE Transactions on Circuits and Systems for Video Technology* 24, 6 (2014), 965–978. https://doi.org/10.1109/TCSVT.2013.2276704

[51] 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 the 26th ACM International Conference on Multimedia* (Seoul, Republic of Korea) *(MM '18)*. Association for Computing Machinery, New York, NY, USA, 1137–1145. https://doi.org/10.1145/3240508.3240607

[52] Liangli Zhen, Peng Hu, Xu Wang, and Dezhong Peng. 2019. Deep Supervised Cross-Modal Retrieval. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. 10386–10395. https://doi.org/10.1109/CVPR.2019.01064

[53] Xitao Zou, Song Wu, Nian Zhang, and Erwin M. Bakker. 2022. Multi-label modality enhanced attention based self-supervised deep cross-modal hashing. *Knowledge-Based Systems* 239 (2022), 107927. https://doi.org/10.1016/j.knosys.2021.107927