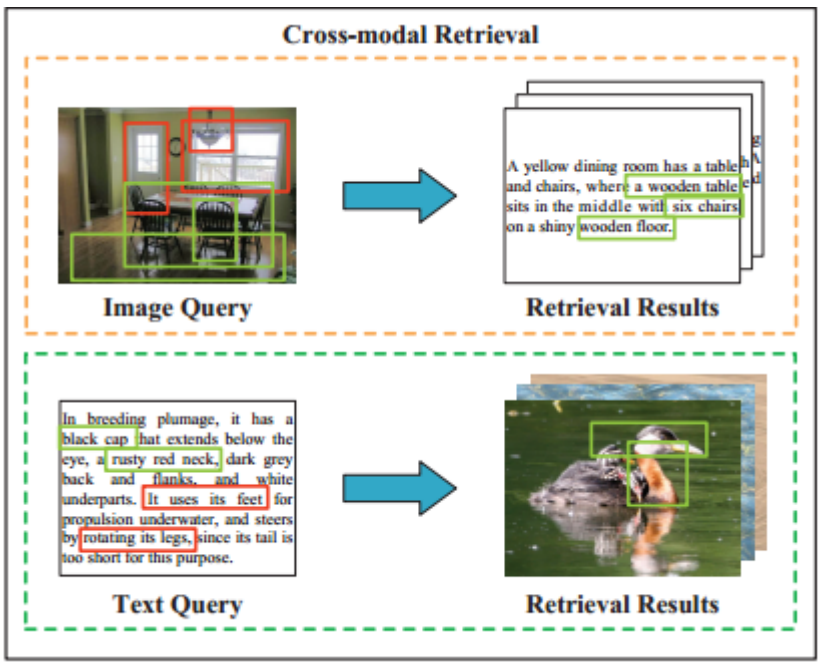
# A Report on Cross-media Retrieval Methods

*All the work is from the research group of Peng Yuxin, Peking University. All the methods reported are DNN-based methods.*

## 0. Problem definition

Cross-media retrieval scenario:




The main challenge of cross-media retrieval is to deal with the inconsistency between different modalities and learn the intrinsic correlation between them.

A key for cross-media retrieval is how to learn cross-modal common representations.

## 1. CBT

**【paper】**Cross-modal Bidirectional Translation via Reinforcement Learning

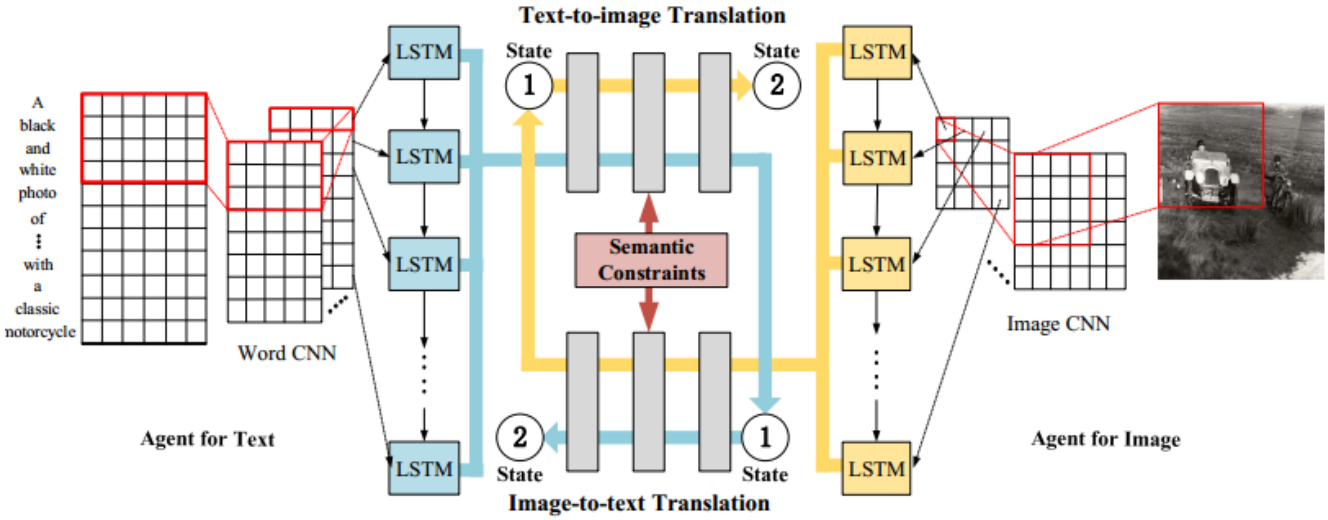
**【source】**IJCAI 2018

**【method】**Cross-modal Bidirectional Translation (CBT)

**【tag】**machine translation, reinforcement learning

In this paper, the authors treat images as a special kind of language. They attempt to conduct bidirectional transformation between image and text to further enhance cross-modal correlation.

**Network architecture**




Firstly use image CNN and Word CNN to extract sequence features and feed them to LSTM models. Then transform the LSTM hidden features from one modality to another and back again by fc layers. The middle layer is connected by weight sharing and a softmax loss layer to keep semantic consistency.

**Reinforcement training process**

Take image-to-text translation as example. The image feature representation $s^i$ from LSTM is transformed to text feature space to get translated feature $s^i_{mid}$, then $s^i_{mid}$ is transformed back to the original image feature space to get reproduced feature $s^i_{ori}$. The text feature representation is $s^t$.

- Inter-modality reward $r^{inter}$: similarity between $s^i_{mid}$ and $s^t$;
- Intra-modality reward $r^{intra}$: similarity between $s^i_{ori}$ and $s^i$.

Total reward:

$$r_p = \alpha r^{inter} + (1 - \alpha) r^{intra}$$

Text-to-image pipeline is similar.

The whole reinforcement training algorithm:

**Algorithm 1** Reinforcement training process of CBT

**Require:** Image training data $I_{tr}$, text training data $T_{tr}$, batchsize $N$, hyper-parameter $\alpha$, learning rate $\gamma$.

**repeat**
  Sample $N$ encoded image representations from the CNN-RNN based network.
  Generate $N$ translated representations for each image $s_p^i$ with $P(\cdot|s;\theta_{IT})$ as $s_{mid,1}^i, ..., s_{mid,N}^i$, and translate back with $P(\cdot|s;\theta_{TI})$ as $s_{ori,1}^i, ..., s_{ori,N}^i$.
  **for** $k = 1, ..., N$ **do**
    Set inter-modality reward $r_p^{inter}$ for the $k$-th sample with equation (7).
    Set intra-modality reward $r_p^{intra}$ for the $k$-th sample with equation (8).
    Set the total reward of the $k$-th sample $r_p$.
  **end for**
  Compute stochastic gradient of $\theta_{IT}$ by equation (9)
  Compute stochastic gradient of $\theta_{TI}$ by equation (10)
  Model updates:
  $\theta_{IT} \leftarrow \theta_{IT} + \gamma \bigtriangledown_{\theta_{IT}} E(r)$,
  $\theta_{TI} \leftarrow \theta_{TI} + \gamma \bigtriangledown_{\theta_{TI}} E(r)$.
  Go through the above process from step 2 to 11 symmetrically for the game beginning from text $s_p^t$.
**until** CBT converges
**return** Optimized CBT model.

## 2. MCSM

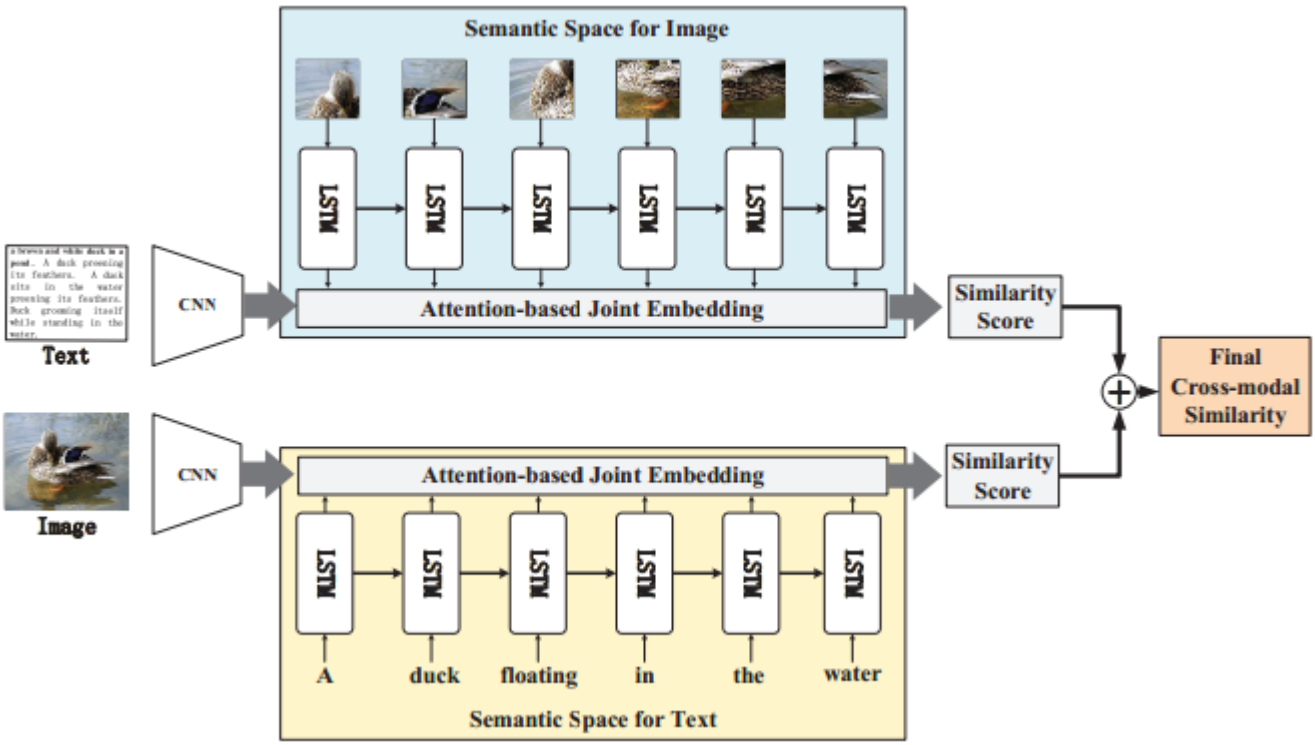
【**paper**】Modality-specific Cross-modal Similarity Measurement with Recurrent Attention Network

【**source**】TIP 2018

【**method**】Modality-specific Cross-modal Similarity Measurement (MCSM)

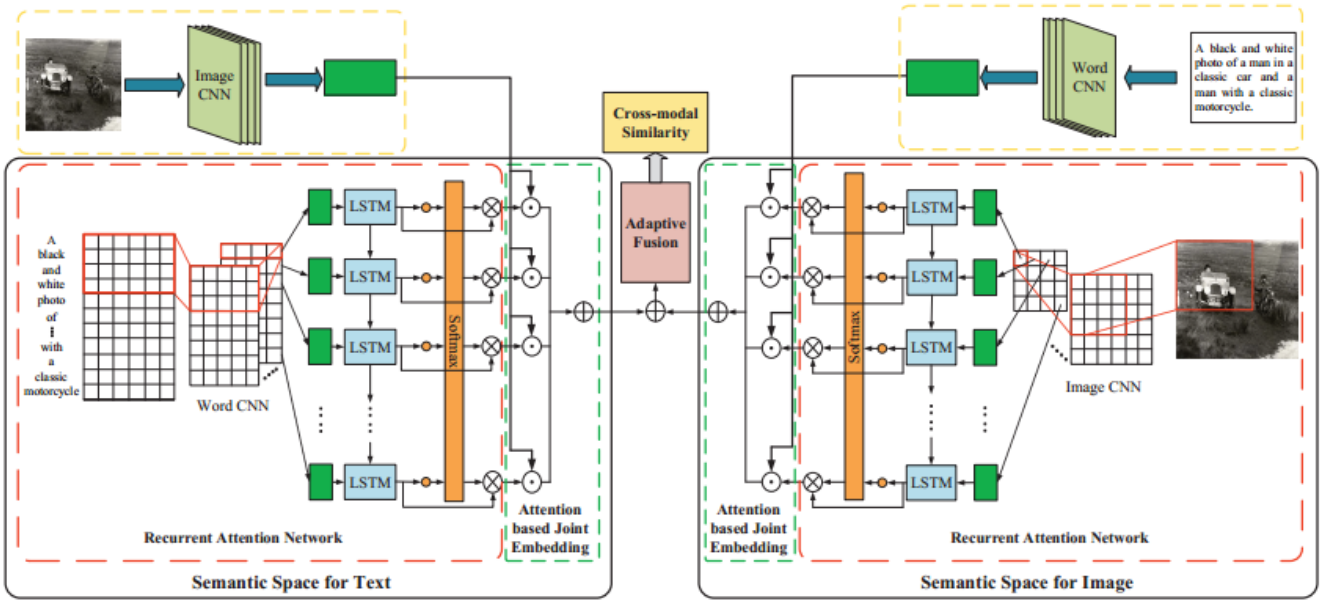
【**tag**】attention

**Overview**




The proposed approach does not construct explicit common feature spaces between image and text modality. It aligns image feature to text feature space and text feature to image feature space respectively in two branches.

**Architecture**




Take image modality as an example. The LSTM hidden outputs are embedded with attention weights. The attention weights are computed by:

$$M^i = \tanh\left(W_a^i H_i\right)$$
$$a^i = \mathrm{softmax}\left(w_{ia}^T M_i\right)$$

The recurrent attention network learns the weighted sum of local information in image modality, while WordCNN learns the global information of text modality. Then a similarity can be computed between them:

$$\mathrm{sim}_i(i_p, t_p) = \sum_{j=1}^{n} a_j^{i_p} h_j^{i_p} \cdot q_p^t$$

An joint embedding loss is defined by both consider matched and mismatched image-text pairs with the defined similarity:

$$L_i = \frac{1}{N} \sum_{n=1}^{N} l_{i1}\left(i_n^+, t_n^+, t_n^-\right) + l_{i2}\left(t_n^+, i_n^+, i_n^-\right)$$

where

$$l_{i1}\left(i_n^+, t_n^+, t_n^-\right) = \max\left(0, \alpha + \mathrm{sim}_i\left(i_n^+, t_n^+\right) - \mathrm{sim}_i\left(i_n^+, t_n^-\right)\right)$$
$$l_{i2}\left(t_n^+, i_n^+, i_n^-\right) = \max\left(0, \alpha + \mathrm{sim}_i\left(i_n^+, t_n^+\right) - \mathrm{sim}_i\left(i_n^-, t_n^+\right)\right)$$

Similarly, for text branch, the similarity score is:

$$\mathrm{sim}_t(i_p, t_p) = \sum_{j=1}^{m} a_j^{t_p} h_j^{t_p} \cdot q_p^i$$

and the jointly embedding loss is:

$$L_t = \frac{1}{M} \sum_{n=1}^{M} l_{t1}\left(t_n^+, i_n^+, i_n^+\right) + l_{t2}\left(i_n^+, t_n^+, t_n^-\right)$$

**Adaptive fusion**: the similarity score $\mathrm{sim}_i(i_p, t_p)$ and $\mathrm{sim}_t(i_p, t_p)$ are further fused to form a final cross-modal similarity, in order to **boost cross-modal retrieval performance.**

Firstly, $\mathrm{sim}_i(i_p, t_p)$ and $\mathrm{sim}_t(i_p, t_p)$ are min-max normalized as $r_i\left(i_p, t_p\right)$ and $r_t\left(i_p, t_p\right)$. Then the fused similarity is computed as:

$$\mathrm{Sim}(i_p, t_p) = r_t\left(i_p, t_p\right) \cdot \mathrm{sim}_i(i_p, t_p) + r_i\left(i_p, t_p\right) \cdot \mathrm{sim}_t(i_p, t_p)$$

The motivation is that larger similarity in one semantic space should lead to a higher importance of the corresponding pair in another semantic space.

## 3. MHTN

【**paper**】MHTN: Modal-adversarial Hybrid Transfer Network for Cross-modal Retrieval

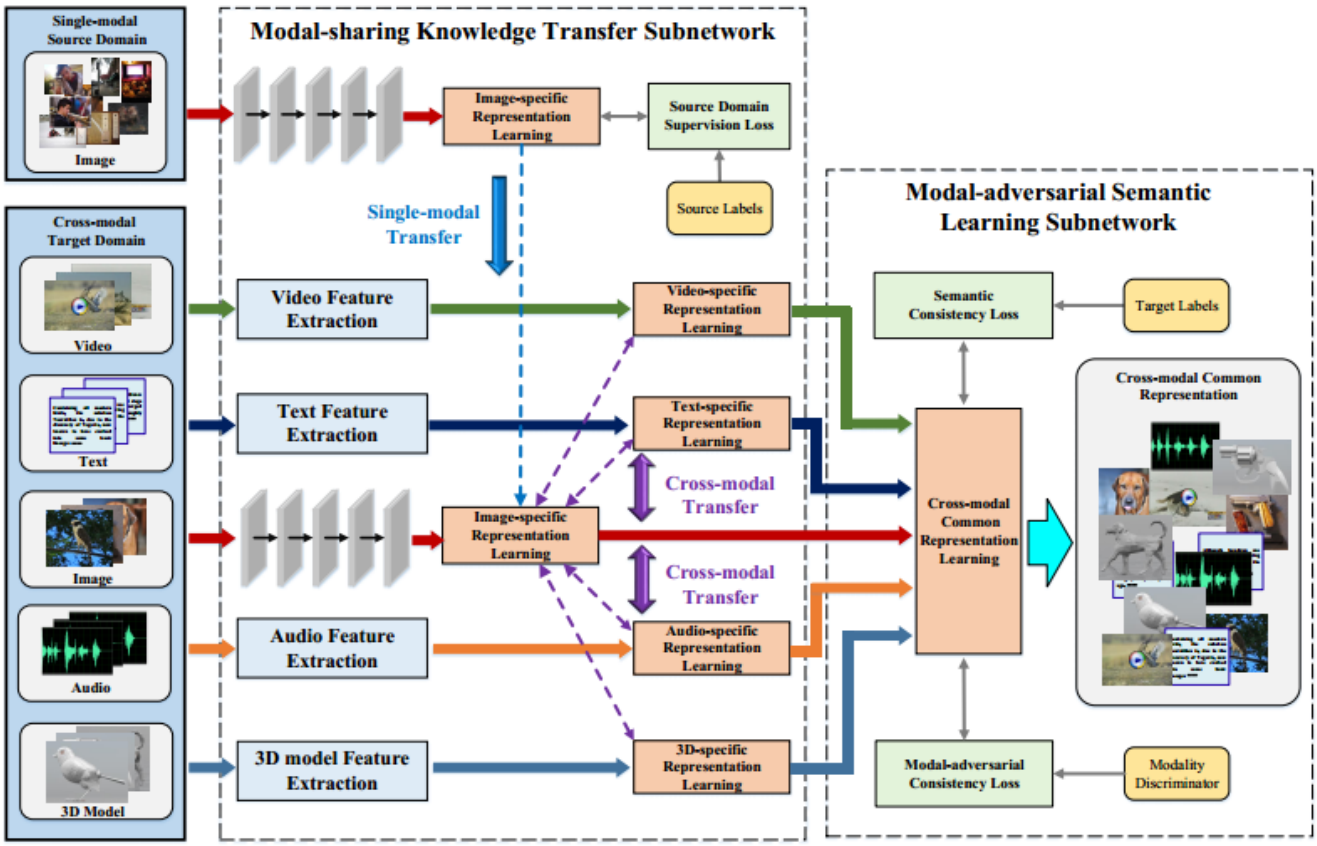
【**source**】TCYB 2018

【**method**】 Modal-adversarial Hybrid Transfer Network (MHTN)

【**tag**】adversarial learning

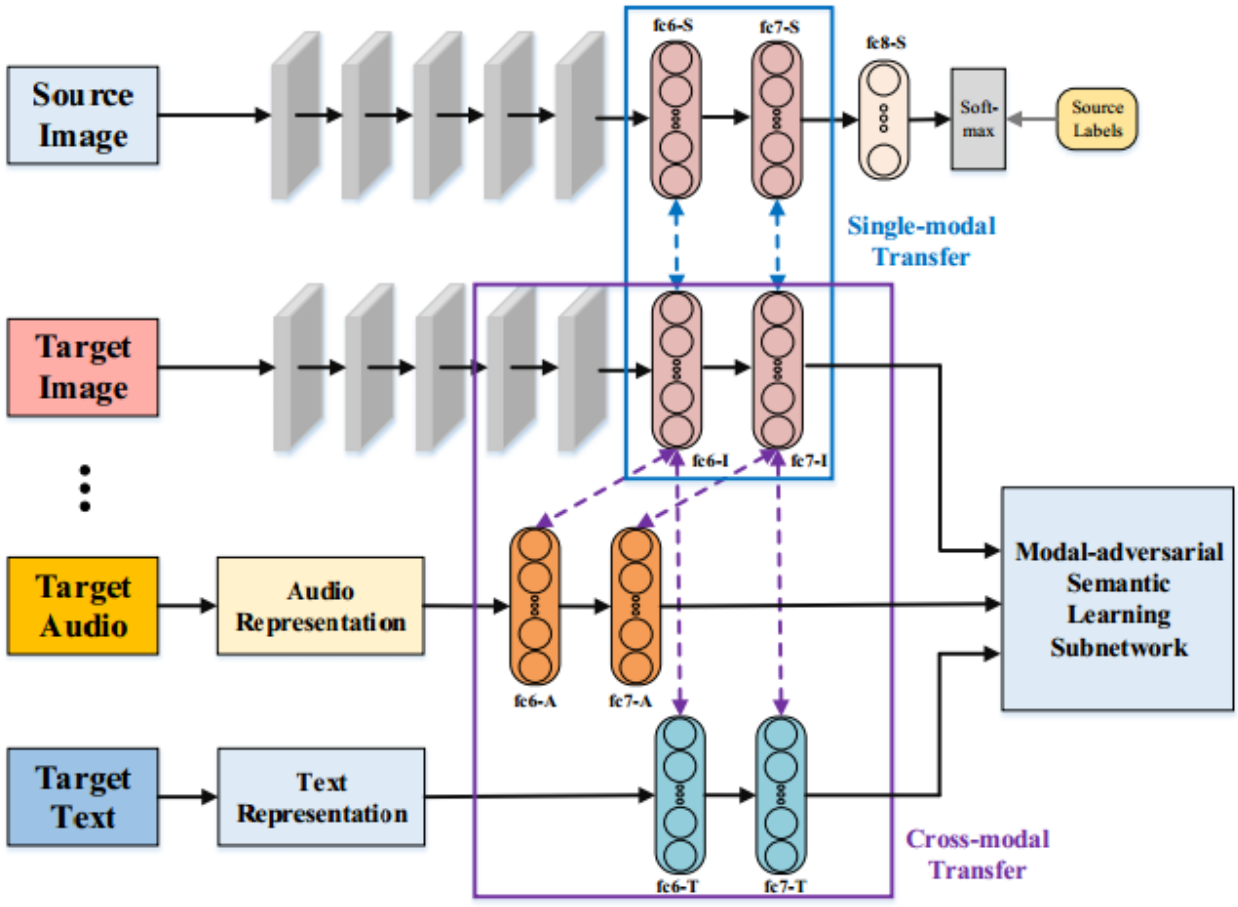
This studies **knowledge transfer from single-modal source domain to cross-modal target domain**. Five modalities are involved: image, audio, video, text, 3D object. The label spaces of source and target domain are **heterogeneous**.

**Architecture**

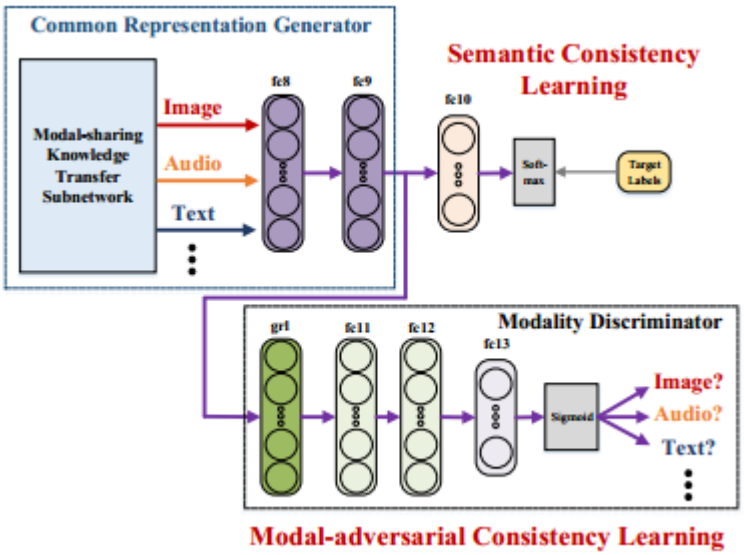



MHTN consists of two subnetworks:

- **Modal-sharing knowledge transfer subnetwork**:



    - **single-modal knowledge transfer**: transfer knowledge from source to target in shared modality (image); a **MMD loss** is added to minimize the discrepancy between source and target image features; a **source classification loss** is also added to preserve sematic constraints.
    - **cross-modal knowledge transfer**: transfer knowledge from the centric modality (image) to other modalities in target domain; a **l2 loss** is added to minimize the discrepancy between features of different modalities.
- **Modal-adversarial Semantic Learning Subnetwork**:




  This module takes the output of Modal-sharing knowledge transfer subnetwork as input. A modality discriminator is used to distinguish the modality of input features. The adversarial process is realized by **gradient reversal layer**. A **target classifier** is also added to preserve semantic consistency.

The total loss of the whole network consists of **five** parts. The final output of the network is a cross-modal common representation.

## 4. CM-GAN

**【paper】**CM-GANs: Cross-modal Generative Adversarial Networks for Common Representation Learning

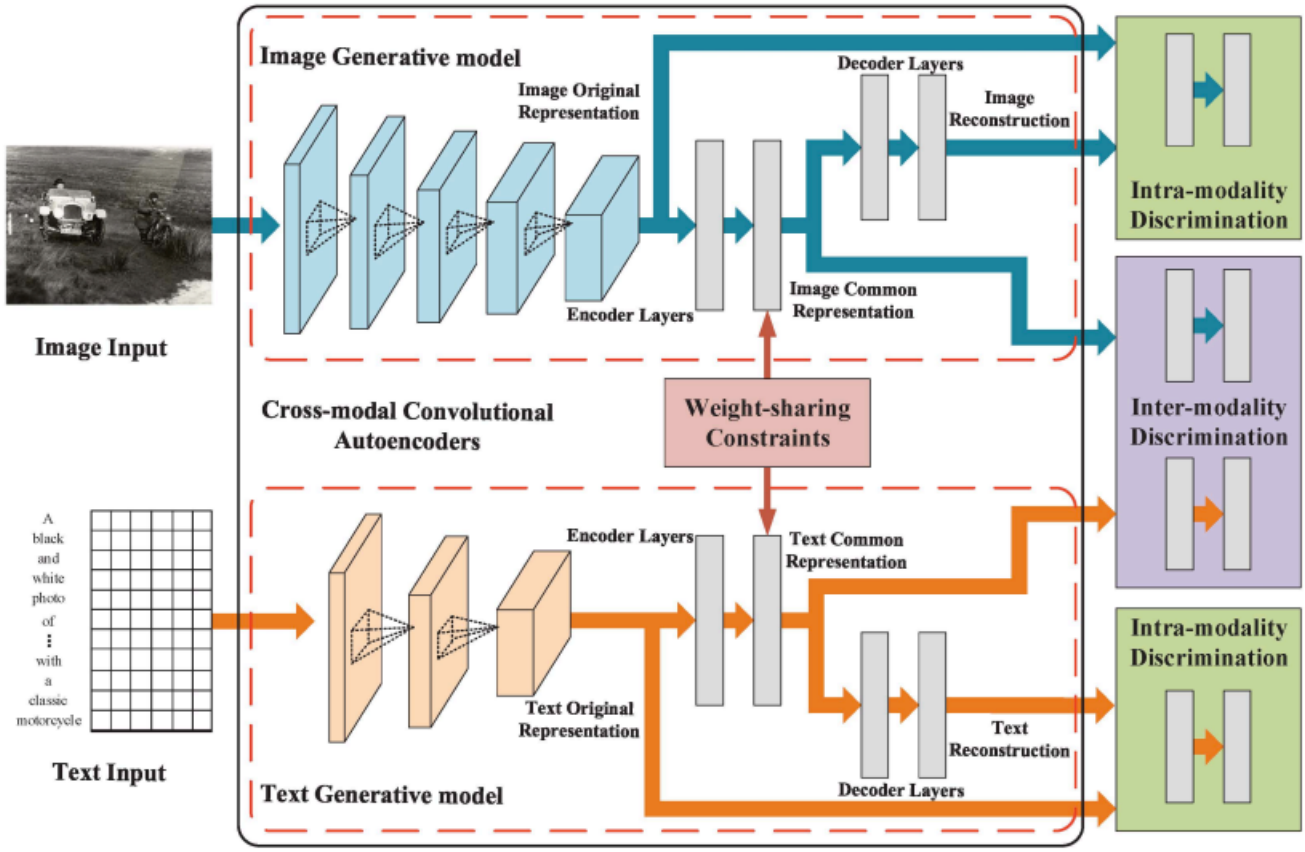
**【source】**TMM 2019

**【method】** Cross-modal Generative Adversarial Network (CM-GAN)

**【tag】**adversarial learning, GAN

This paper aims at learning a common representation for image and text modality via GAN-based adversarial learning.

**Architecture**




1) Generator:

The generator of CM-GAN is convolutional autoencoders. For both image and text inputs, the latent features (i.e., the output of encoder) are extracted by CNNs. The latent feature goes through several fc layers to get image or text common representations. The weights of the last layer are shared between two branches.

Then the common representation goes through several fully-connected decoder layers to reconstruct the original latent feature.

Therefore, for each branch, **three** representations are generated: original latent representations, common representations and reconstructed representations.

2) Discriminator:

- Intra-modality discrimination: discriminates original latent representation and the reconstructed representation of image and text modality respectively;
- Inter-modality discrimination: discriminates the image common representation and the text common representation.

## 5. TPCKT

【**paper**】TPCKT: Two-level Progressive Cross-media Knowledge Transfer

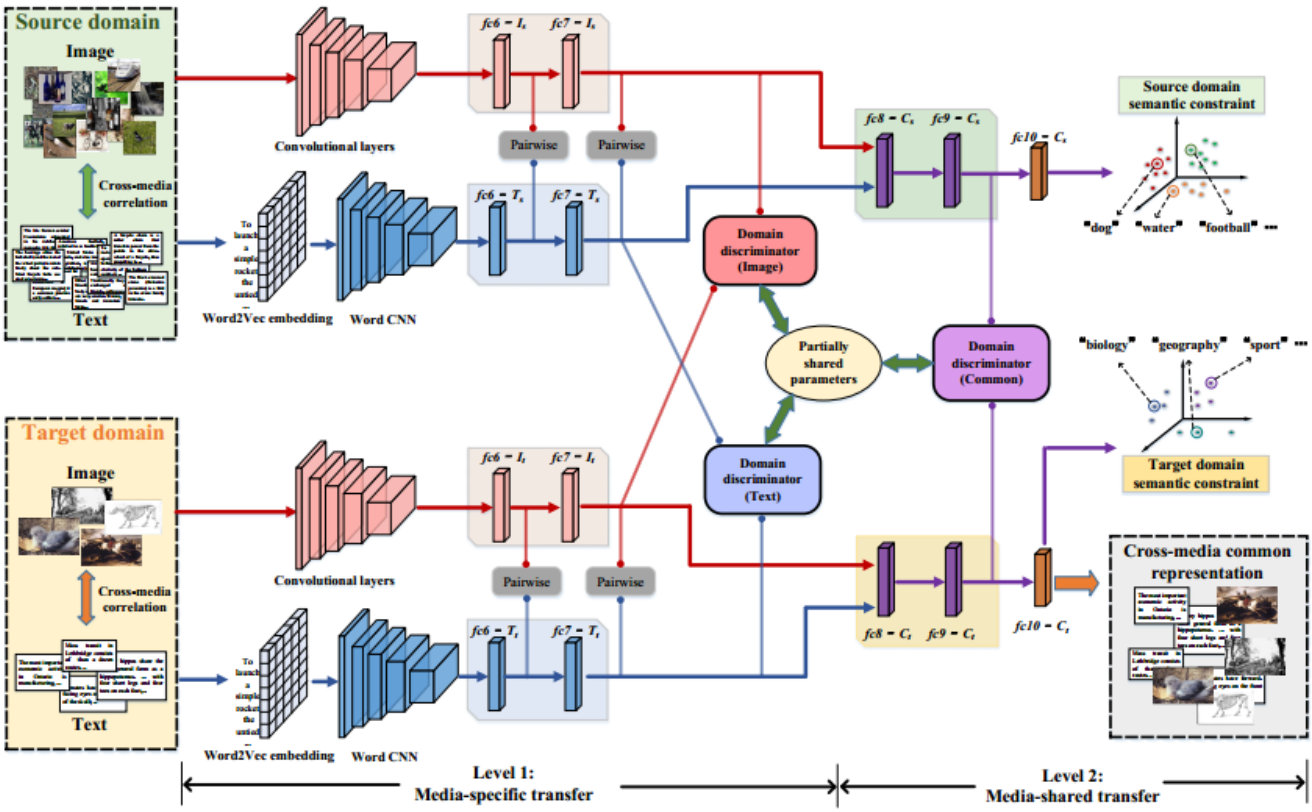
【**source**】TMM 2019

【**method**】 TPCKT

【**tag**】adversarial learning

This paper studies **knowledge transfer from cross-modal source domain to cross-modal target domain**. There are two main challenges: 1) inconsistency information between different modalities; 2) disjoint label space between source and target domain. For simplicity, image and text modality are chosen in this paper.

**Architecture**




1) Media-specific transfer

The media-specific transfer aims at align the data distributions of the same media type between source and target domain. A domain discriminator is adopted for each media type. One GRL layer is used for adversarial learning.

To preserve the intrinsic consistency between two medias, the authors proposed two strategies. First, the media-specific domain discriminators of image and text share the last fc layer. Second, a pairwise L2 loss is applied in fc6 and fc7 layers between the representations of two medias for each domain.

2) Media-shared transfer

The outputs of $fc7-I/fc7-T$ are passed through two fc layers, which are shared between two medias, to generate a common representation. A media-shared discriminator is used to distinguish the common representations of source and target domain.

Note that the last fc layer of this discriminator is also shared with the previous media-specific domain discriminators, in order to "enhance the consistency of media -specific and media-shared transfer processes".

Finally the classification loss of source and target domain is added for maintaining the semantically discriminative ability.

**Training**

The authors propose a **progressive semantic transfer mechanism** to train the whole network. The motivation is to avoid noisy and useless information in the early training period, i.e., to make the early training stage easier by gradually choose "harder" samples for the network.

Whether a category is easy or not is defined by its similarity with the other domain. If the similarity is high, then samples from that category will be chosen.

The output of fc9 layer (i.e., the common representation) is used for compute similarity. The similarity between a source class and the target domain is the mean of the similarities between that source class and all the target classes. Computation of the similarity between a target class and source domain is the similar.

Then the similarities are ranked in a descending order. Top-k classes in source domain and top-l classes are selected for the training in this iteration.

**Algorithm 1** : Progressive Semantic Transfer

**Require:** Training data of two domains $Src$ and $Tar$, maximal number of iteration $MI$, and training epoch number in each iteration $Ep$.

1: Pre-train the model as $Model(0)$. Set $iter = 1$.
2: **repeat**
3: Use $Model(iter - 1)$ to generate the media-shared representation as $C_s$ and $C_t$ for $Src$ and $Tar$.
4: Compute the mean vectors of each $C_s^i$ and $C_t^j$, as $\overline{C_s^i}$ and $\overline{C_t^j}$. Then obtain similarities $Sim(i, j)$ via Equation 16.
5: Compute $Con(C_s^i) = Average\{Sim(i, j)\}_{j=1}^{n}$, $Con(C_t^j) = Average\{Sim(i, j)\}_{i=1}^{m}$.
6: Compute $k$ and $l$ via Equation 15. Select top-$k$ categories in $Src$ with highest $Con(C_s^i)$. Select top-$l$ categories in $Tar$ with highest $Con(C_t^j)$.
7: Train model for $Ep$ epochs with data of selected categories, to get $Model(iter)$.
8: $iter = iter + 1$.
9: **until** $iter = MI$.
10: **return** $Model(MI)$.

## 6. $C^3L$

【**paper**】Show and Tell in the Loop: Cross-Modal Circular Correlation Learning

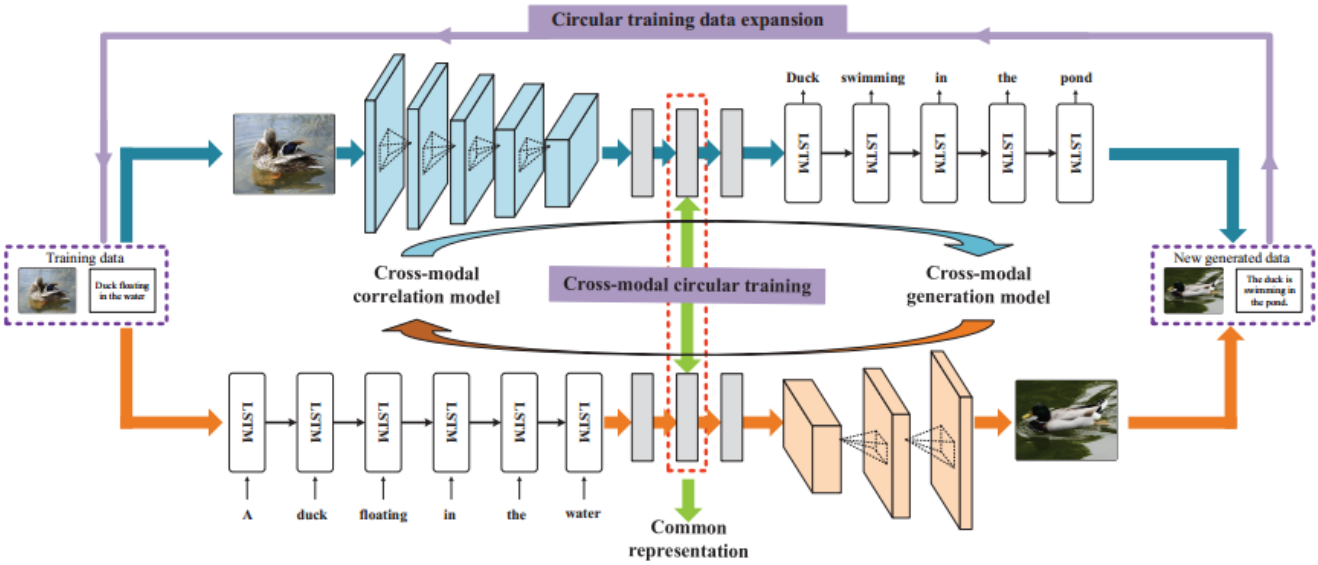
【**source**】TMM 2018

【**method**】 Cross-Modal Circular Correlation Learning ($C^3L$)

【**tag**】bidirectional generation

The authors propose a unified framework with a circular learning process for cross-modal retrieval, image-to-text caption and text-to-image synthesis.

**Framework**




The whole framework consists of two pathways:

- Image-to-text pathway: an encoder CNN and a decoder RNN for image-to-text caption;
- Text-to-image pathway: an encoder RNN and a decoder CNN for text-to-image synthesis.

Following the encoder CNN and encoder RNN, three fc layers are adopted. The intermediate layer is linked between two pathways for cross-modal common representation learning. Then the outputs of the last fc layer are fed to the decoder CNN or decoder RNN for generation.

The new generated image-text pairs are fed back as the inputs for data augmentation.

**Training**

In order to learn the common representation, a triplet loss is adopted:

$$L_{corr} = \frac{1}{N}\sum_{n=1}^{N} l_{corr_1}\left(i_n^+, t_n^+, t_n^-\right) + l_{corr_2}\left(t_n^+, i_n^+, i_n^-\right)$$

where

$$\begin{aligned} &l_{corr_1}\left(i_n^+, t_n^+, t_n^-\right) = \\ &\max\left(0, \lambda - \operatorname{sim}\left(i_n^+, t_n^+\right) + \operatorname{sim}\left(i_n^+, t_n^-\right)\right) \\ &l_{corr_2}\left(t_n^+, i_n^+, i_n^-\right) = \\ &\max\left(0, \lambda - \operatorname{sim}\left(t_n^+, i_n^+\right) + \operatorname{sim}\left(t_n^+, i_n^-\right)\right) \end{aligned}$$

The similarity is defined as the dot product on the common representation of image and text:

$$\operatorname{sim}(i_p, t_p) = c_p^i \cdot c_p^t$$

The image-to-text pathway is trained by the negative log likelihood objective function:

$$L_{txtGen}\left(c_p^i, S_p\right) = -\sum_{t=1}^{K} \log w_t\left(s_t\right)$$

The text-to-image pathway is trained in the form of a conditional GAN:

$$L_{imgGen} = \log D\left(s_q^i, c_q^t\right) + \frac{1}{2}\left(\log\left(1 - D\left(s_q^{i-}, c_q^t\right)\right) + \log\left(1 - D\left(\hat{s_q^i}, c_q^t\right)\right)\right)$$