Joint Similarity Guidance Hash Coding Based on Adaptive Weight Mixing Strategy For Cross-Modal Retrieval

Yaqi Sun **Jing Yun** **Zhuoqun Ma**
College of Data Science and Application, Inner Mongolia University of Technology, Huhhot, China
yunjing@imut.edu.cn

Abstract

There is a continuous and explosive growth of multimodal data. Efficient cross-modal hashing retrieval is of significant importance in conserving computational resources.To further enhance the attention to informative data within modalities and capture the semantic correlations in cross-modal data, we propose an enhanced deep Joint-Semantics Reconstructing Hashing algorithm, which is the Joint Similarity Guidance Hash Coding Based on Adaptive Weight Mixing Strategy(JSGHCA). The algorithm focuses on delving deeper into the correlations of the data in cross-modal. We introduce the adaptive weight mixing strategy to construct the semantic affinity matrix, so that the matrix can identify each modal data with specific weight in each batch. At the same time, in the process of the hash code generation, we introduce collaborative attention mechanism. It helps the model to pay more attention to the local information of each modality, thereby capturing the semantic features within each modality more accurately. Additionally, it enables the model to jointly process the attention across different modalities and extract shared semantic features more precisely. Experimental results show that the proposed model is significantly better than the deep joint semantic reconstruction hash algorithm on multiple benchmark datasets.

1 Introduction

With the rapid development of digital technology, multimodal information is omnipresent in our lives. On the internet, various modalities of data such as text, images, videos, and audio provide us with a rich information experience. The diverse structures of different modalities of data give rise to data heterogeneity. This heterogeneity poses a new challenge: how to effectively establish semantic associations and perform retrieval across different modalities?

Cross-modal retrieval is divided into cross-modal real-value retrieval and cross-modal hash retrieval. In cross-modal real-value retrieval, the dimensionality of real-valued vectors is relatively high, requiring large-scale matrix operations and high computational complexity. Cross-modal hash retrieval aims to eliminate differences between modalities and project them into a low-dimensional common Hamming space for retrieval tasks. In this regard, Deep Joint-Semantics Reconstructing Hashing (DJSRH (Su et al., 2019)) marks a significant milestone. It optimizes cross-modal semantic relationships by introducing a joint semantic affinity matrix. However, there is interference from irrelevant information of the two modalities during the synthesis of the joint semantic affinity matrix. Additionally, there are challenges in capturing intra-modal local information and utilizing cross-modal related information during hash code generation.

Considering these limitations, we propose a Joint Similarity Guidance Hash Coding Based on Adaptive Weight Mixing Strategy (JSGHCA) algorithm. The traditional fixed weight allocation strategy may be affected by irrelevant information when processing multimodal data, leading to a decrease in retrieval accuracy. To this end, we propose a dynamic weight allocation method that can dynamically adjust weights based on the importance of modal information within the data batch. Through this method, we

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 999-1010, Taiyuan, China, July 25 - 28, 2024.
Volume 1: Main Conference Papers

can more accurately identify and emphasize useful modal information when constructing a joint semantic affinity matrix, while filtering out useless information, thereby improving the accuracy and robustness of retrieval. When dealing with the local information within a single modality and capturing the complex semantic correlation between different modalities, the existing methods often have shortcomings. To address this issue, we introduce an enhanced local attention mechanism (Single Attention Mechanism, Si-Attention) that focuses more on key information within the modality during the feature extraction stage. Meanwhile, we also designed a cross modal semantic capture module (Joint Attention Mechanism, Jo-Attention) that can effectively capture and utilize the semantic associations between different modalities during the hash code generation process. Through these improvements, our model can generate richer and more accurate hash codes, thereby improving the performance of cross modal retrieval. The main contributions of this research are as follows:

- We introduce an adaptive weight mixing strategy. This approach dynamically adjusts weights based on the importance of modal information within each data batch, enhancing the retrieval process and filtering out irrelevant information.
- We introduce an attention mechanism (Collaborative Attention Mechanism) to ensure that the network primarily focuses on core information, thereby significantly enhancing the accuracy and relevance of hash encoding. We introduce an enhanced local attention mechanism Si-Attention that focuses on key information within individual modalities during the feature extraction phase. Additionally, we design a joint attention mechanism Jo-Attention to effectively capture and utilize the complex semantic correlations between different modalities in the hash code generation process.
- We conduct systematic experimental validation and performance evaluation, encompassing a series of comprehensive experiments aimed at thoroughly assessing the performance and effectiveness of JSGHCA. Through this approach, we aim to provide a more powerful, flexible, and practical hashing solution for cross-modal data, bringing new breakthroughs to the field of cross-modal information retrieval.

2 Related Work

Cross-modal retrieval, an extension of single-modal retrieval, tackles the intricate relationships within multimodal data. Recently, cross-modal hashing learning has been extensively employed to reduce feature storage and simplify retrieval.Cross-modal hashing learning can be divided into supervised and unsupervised methods based on the need for semantic supervision.

2.1 Supervised cross-modal hashing learning

Supervised cross-modal hashing learning utilizes source data and label information to explore the similarity between image and text modalities. For example, SDDH (Qin et al., 2021) employs orthogonal and equilibrium constraints to maintain modality similarity, QDCMH (Liu et al., 2021) constructs a quadruplet loss function to preserve semantic correlation, OLCH (Yi et al., 2021) acquires semantic labels through multi-class classification, NSDH (Yang et al., 2021) utilizes a three-layer network to extract features and generate hash codes, while $MSLF_1$ (Cheng et al., 2021) maximizes shared factors to obtain effective hash codes. Despite performing well on labeled cross-modal data, these methods heavily rely on large annotated datasets, thus limiting their practical applicability.

2.2 Unsupervised cross-modal hash learning

Unsupervised cross-modal hashing learns binary codes by leveraging modality correlations without the need for class labels. Some methods rely on constructing instance-level affinity graphs or similarity matrices to deeply analyze relationships between instances, revealing valuable similarity information. For instance, SIM (Jia et al., 2020) builds a node similarity graph, utilizing the triangle inequality principle and path averages to determine distances between nodes. SRCH (Wang et al., 2021) constructs K-nearest neighbor graphs for image and text instances, where the new graph only contains consistent relationships. UGACH (Zhang et al., 2018) creates graphs to represent relationships between image and

Proceedings of the 23rd China National Conference on Computational Linguistics, pages 999-1010, Taiyuan, China, July 25 - 28, 2024.
Volume 1: Main Conference Papers

text instances, typically requiring the pre-construction of the entire training set's graph structure, which is time-consuming and requires large storage space, limiting its application to large-scale data. In practical applications, such methods need to balance accuracy and computational feasibility.

Another type of unsupervised cross-modal hashing method first extracts real-valued features from different modal data and constructs a similarity matrix of these features through feature multiplication. Then, it maps the data to a hash space using a hash function and constructs a similarity matrix of hash features. In the loss function, this method compares these two similarity matrices and optimizes the hash function by minimizing the difference between them, ensuring that the generated hash codes better preserve the similarity relationships between the original data. For example, the DJSRH (Su et al., 2019) method integrates neighborhood information of multi-modal data by constructing a joint semantic correlation matrix and trains a network to generate binary codes that can reconstruct these semantic relationships to the maximum extent. DGCPN (Yu et al., 2021) extracts global similarity information by constructing a static KNN graph, combines local similarity of training batches, constructs a real-valued feature similarity matrix, then transforms it into a hash feature similarity matrix, and iteratively optimizes cross-modal hashing learning using both. DSAH (Yang et al., 2020) adopts an encoder-decoder structure to convert real-valued features of images and text into hash features, maintaining the similarity relationships between data by constructing and comparing similarity matrices of real-valued features and hash features in the loss function. Existing multi-modal retrieval methods still exhibit instability when dealing with large-scale, unbalanced, or noisy data, and many methods are limited in practical applications.

Compared to existing cross-modal retrieval methods, our approach integrates attention mechanisms, introduces a novel adaptive weighting strategy, and employs batch processing during model training, leading to performance improvements across various aspects.

3 Method

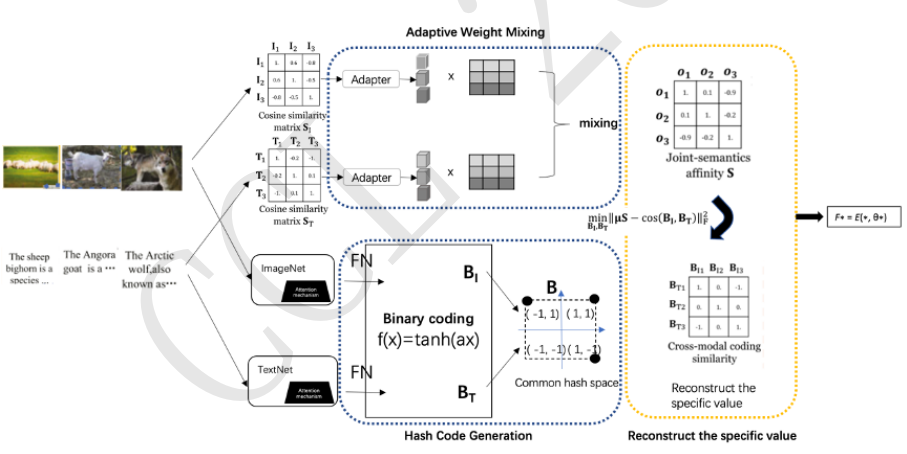

Figure 1: The structure of Cross-modal Hash Retrieval method JSGHCA

The structure of the JSGHCA method is shown in the figure1. At first, JSGHCA constructs similarity matrices for image features and text features based on the instances in each training batch. The model obtains the contribution degree of each modality in the overall similarity calculation by using the adaptive weighting strategy. We then combine the two similarity matrices according to the weights calculated within each batch, paying attention to the associations within and between modalities. In addition, JSGHCA uses pre-training network combined with collaborative attention mechanism to obtain hash features and construct hash similarity matrix, which is compared with the similar matrix of real-valued features. The model is trained by a stepwise training strategy. Firstly, the network of various modalities is trained separately, and then the cross-modal network is trained uniformly. The JSGHCA method is a comprehensive approach that combines the advantages of both deep learning and hashing techniques.

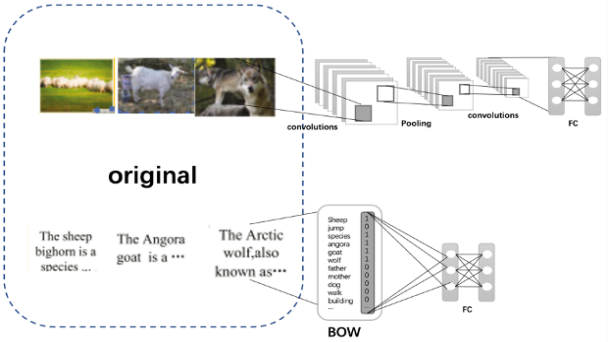

Figure 2: The extraction of the image and text features

We represent the image text pair as $p_i = (v_i, t_i)$, and the batch size is represented as m, then the whole set is $P = p_1, p_2, \cdots p_m$. For real-valued features, image and text are respectively represented as $F_V \in R^{m \times d}$ and $F_I \in R^{m \times d}$, where d represents dimension. For hash features, image and text are respectively represented as $B_V \in R^{m \times h}$ and $B_T \in R^{m \times h}$, where h represents the hash code dimension.

JSGHCA uses pre-trained model AlexNet (Krizhevsky et al., 2012) to extract image feature F_V, and the Bag-of-words (Gálvez-López and Tardos, 2012) model is used to extract text feature F_T. The formula is as follows:

$$F_* = E(*, \theta_*), * \in V, T \tag{1}$$

The hash feature length is set to a fixed value. Before converting a real-valued feature to a hash feature using function $sgn(\cdot)$, the dimension of the real-valued feature is reduced to the length of the hash code through the full join layer. The formula is as follows:

$$B_* = sgn(F_*), * \in V, T \tag{2}$$

When $sgn(\cdot)$ is used as an activation function, all non-zero inputs result in zero gradient, known as the vanishing gradient problem. To address this, we employ the scaled $\tanh(\cdot)$ function instead of $sgn(\cdot)$ because it indefinitely approximates $sgn(\cdot)$ as it approaches the limit.The formula is as follows:

$$sgn(x) = \begin{cases} 1, & \text{x} \geq 0 \\ -1, & \text{x} < 0 \end{cases} \tag{3}$$

$$\lim_{x \to \infty} \tanh(\delta x) = sgn(x) \tag{4}$$

where the parameter δ will increase with the increase of training times.

3.1 Adaptive weights Mixing strategy

The Unified Semantic Affinity Matrix is a two-dimensional matrix used for comparing multimodal data by mapping different data modalities into a shared semantic space. Assuming we have two modalities of feature representations F_V and F_T, we can initially compute the similarity between them. After normalizing F_V, F_T to F'_V, F'_T which has unit $\ell 2 - norm$ each row, we can calculate the cosine similarity matrices $S_I = F'_I F'^T_I \in [-1, +1]^{m \times m}$ and $S_T = F'_T F'^T_T \in [-1, +1]^{m \times m}$ to describe the original neighborhood structure for the input images and texts respectively. Once we have the similarity matrix for each modal, we can calculate the adaptive weights. These weights can be calculated according to the similarity matrix of each modal, which can reflect the degree of contribution of each modal to the overall similarity calculation. Specifically, we can calculate adaptive weights using the following formula:

$$w_i = \frac{S_{irowavg}}{S_{itotalavg}} \times \frac{S_{totalavg}}{S_{totalrowavg}} \tag{5}$$

Where w_i is the adaptive weight of modality I, S_i is the similarity matrix of modal I, and S_total is the sum of all modal similarity matrices. Through this formula, we can get the adaptive weight of each modality, which can be adjusted according to the similarity of each modality.

Algorithm1 Outlines the whole process of our proposed adaptive weight strategy.

Algorithm 1 Adaptive weight mixing strategy(AWM).

Input: Similarity matrix of two modalities SI and ST
Output: The similarity matrix after fusion FusionSimMatrix

1: Calculate the row average and total average for each modality:
$row\_avg1 = calculate\_row\_average(SimMatrix1)$
$total\_avg1 = calculate\_total\_average(row\_avg1)$
$row\_avg2 = calculate\_row\_average(SimMatrix2)$
$total\_avg2 = calculate\_total\_average(row\_avg2)$
2: Calculate the adaptive weights:
$adaptive\_weight1 = row\_avg1/total\_avg1$
$adaptive\_weight2 = row\_avg2/total\_avg2$
3: Fusion similarity matrix:
$FusionSimMatrix = zeros((len\_r(SimMatrix1), len\_r(SimMatrix2)))$
for i in range(len_r(SimMatrix1)):
for j in range(len_r(SimMatrix2)):
$FusionSimMatrix[i,j] = SimMatrix1[i,j] * adaptive\_weight1[i] + SimMatrix2[i,j] * adaptive\_weight2[i]$
4: **return** FusionSimMatrix

When we have adaptive weights for different modalities, we can utilize them to adjust and fuse the weights between modalities. By multiplying the weight of each modal by the corresponding adaptive weight, we obtain the final weight for each modal in the overall similarity calculation. Subsequently, combining the feature vectors and weights from different modalities allows us to achieve richer representations. Adjusting the weights and similarity calculation methods further optimizes the computation of the joint semantic affinity matrix, which reflects semantic similarity between modalities and guides subsequent tasks. JSGHCA introduces an adaptive weight strategy to dynamically assign weights to each modality, considering not only the inherent importance of features but also ensuring balanced representation across modalities. This approach enhances the performance of cross-modal retrieval, capturing inter-modality correlations effectively, and providing a more reliable foundation for hash coding.

3.2 Collaborative Attention Mechanism

The attention mechanism is used before hashing encoding forms a similarity matrix. Firstly, feature learning module is used to extract the features of image and text respectively. Considering that some information in the original data is useless for retrieval and in order to reduce the interference of these noises, this paper adds a local attention module to the network to extract the key information of each modality separately. Finally, semantic hash code is learned through semantic hash code generation module. In the process of hash code learning, joint attention mechanism is used to enhance the relationship between image and text.

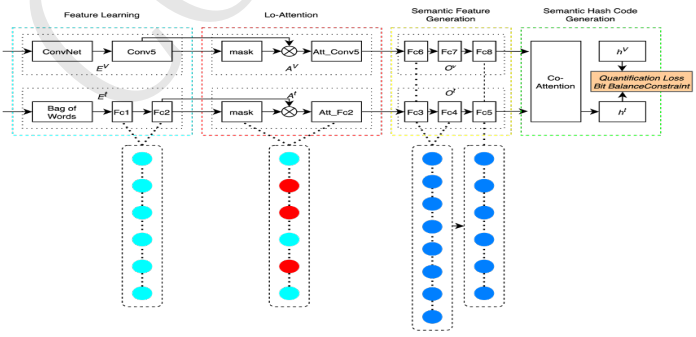

Figure 3: The structure of Collaborative Attention Mechanism

Si-Attention includes A^V and A^T, and extracts images using an attentional mechanism of local feature extraction. For the attention operation of the image part, the image feature map X_i^v obtained above is first input into the convolution layer with the convolution kernel size of 1×1 for compression, and the matrix $M_i^v = conv(X_i^v)$,$M_i^v \in R^{(H \times W)}$ is obtained. Then perform sigmoid activation function operation on M_i^v to get the attention mask $\hat{M}_i^v = sigmoid(M_i^v)$,$\hat{M}_i^v \in R^{(H \times W)}$. Finally, X_i^v dot the attention mask

$\hat{M}_i^v$ to get the key feature of the $i-th$ image $Y_i^v = \hat{M}_i^v \times X_i^v$,$Y_i^v \in R^{(H\times W)}$, which represents the Kronecker product. The whole process of attention processing for the $i-th$ image data can be simplified as $Y_i^v = A^v(X_i^v)$.

$$\begin{aligned} M_i^t &= fc(M_i^t) \\ \hat{M}_i^t &= sigmoid(M_i^t) \\ Y_i^t &= \hat{M}_i^t \times X_i^t \end{aligned} \tag{6}$$

For the attention operation of the text part, the key features Y_i^t of the $i-th$ text feature vector X_i^t can be obtained in the same way, which is implemented as follows(equation(6)):

Where fc is a fully connected layer with 8192 neurons, M_i^t, $\hat{M}_i^t$,$Y_i^t \in R(H \times W)$. The whole process of attention processing for the $i-th$ text feature vector can be simplified as $Y_i^t = A^t(X_i^t)$.

In order to generate more consistent hash code and reduce the information imbalance between image and text modules, a joint attention mechanism is added to this module, which uses the features learned by one module to guide the hash code training of another modal. Specifically, the generated image and text features Q_i^v and $Q_i^t$are input into the joint attention mechanism, and vectors $O_i^v = fc(Q_i^v)$,$O_i^v \in R^c$ and $O_i^t = fc(Q_i^t)$,$O_i^t \in R^c$ are generated through the full connection operation with c neurons and the ReLU activation function operation. The softmax function is then used for O_i^v and O_i^t respectively to obtain the attention distributions $\hat{O}_i^v$ and $\hat{O}_i^t$, i.e. $\hat{O}_i^v = softmax(Q_i^v)$, $\hat{O}_i^v \in R^c$ and $\hat{O}_i^t = softmax(Q_i^t)$, $\hat{O}_i^t \in R^c$. Finally, Q_i^v and Q_i^t features of the image and text are multiplied by the attention distribution $\hat{O}_i^t$ and $\hat{O}_i^v$ to update the features, respectively, and the semantic hash code $h_i^v \in R^c$ and $h_i^t \in R^c$ of the image and text is obtained, which is specifically expressed as follows:

$$\begin{aligned} h_i^v &= \hat{O}_i^t \diamond Q_i^v \\ h_i^t &= \hat{O}_i^v \diamond Q_i^t \end{aligned} \tag{7}$$

Where $\diamond$ represents the Hadamard Product. According to formula (7), it can be found that in the proposed joint attention mechanism, attention distribution $\hat{O}_i^t$ can be used to guide image semantic hash code h_i^v generation, while attention distribution $\hat{O}_i^v$ can be used to guide text semantic hash code h_i^t generation.

3.3 Joint similarity matrix guides hash coding

By constructing a joint semantic affinity matrix S, we aim to uncover the latent semantic relationships among batch input instances. This is achieved by minimizing the reconstruction error between the desired neighborhood matrix S and the hash code structure $cos(B_V, B_T)$ to be learned, thereby acquiring semantically relevant binary codes. The main functions of this approach are twofold: on the one hand, the introduction of hyperparameter μ enhances the flexibility of the reconstruction process, allowing it to better adapt to diverse data requirements; on the other hand, it constructs a specific similarity value among modalities to facilitate learning.

$$\begin{aligned} J &= \min_{B_V, B_T} \; || \, \mu S - cos(B_V, B_T) \, ||_F^2 \\ s.t. S &= C(S_V, S_T) \in [-1, +1]^{(m\times m)} \end{aligned} \tag{8}$$

When we use hash coding to represent the similarity between images and texts, for instance, there are two pictures, one of a sunset beach and the other of a sunrise beach. They are actually very similar, but according to strict similarity measures, they may not be classified into the same category. The hyperparameter μ acts like a "flexibility" adjuster. When we feel that the judgment criteria are too strict, we can increase μ to relax the standards; conversely, if we want to more accurately determine similarity, we can decrease μ. In this way, we can more accurately capture the true similarity relationship between images and texts.

The focus of the reconstruction framework is on maximizing the reconstruction of specific similarity values in S, rather than merely emphasizing their relative order. This approach is less sensitive to the composition of each randomly sampled training batch, making it more suitable for batch processing input methods. This article adopts the aforementioned method as the core model to establish comprehensive

and systematic training objectives. In addition to integrating the multimodal reconstruction components of B_V and B_T in the framework, it also strengthens the internal modal reconstruction. This is because according to previous research, fully considering the internal and inter-connectivity during cross-modal network training can effectively improve retrieval efficiency. Therefore, this article utilizes the following optimized training objectives for JSGHCA.

$$J = \min_{B_V, B_T} || \mu S - cos(B_V, B_T) ||_F^2 + \lambda_1 || \mu S - cos(B_V, B_T) ||_F^2 + \lambda_2 || \mu S - cos(B_V, B_T) ||_F^2 \quad (9)$$

λ_1 and λ_2 are tradeoff parameters for balancing inter-modal reconstruction and intra-modal reconstruction. Introducing the hyperparameter μ into the framework makes the reconstruction more flexible. Compared to the Laplacian-constrained model, selecting specific similarity values for reconstruction is more compatible with batch training.

3.4 Training and Objective Function

Image and text belong to two different modalities, but the previous work unified the training of the data of these two modalities, ignoring their respective rules and characteristics. Therefore, the JSGHCA method adopts a step-by-step training method, firstly training the image network and text network separately, and then uniformly training the whole network, and defines the parameter λ Used to adjust the loss function range. The overall procedure of our proposed JSGHCA is summarized in Algorithm2.

Algorithm 2 The training process of JSGHCA method.

Input: image dataset I, text dataset T, batch size m, hash code length c, maximum training batch N, parameter values λ,μ,...
Output: Feature extraction function $F* = E(*, \theta*), * \in \{I, T\}$

for $n = 1$ to N do
Extract real-valued features and hash features from image set I and text set T.
Construct the domain similarity matrix SI and ST of real-valued features.
Use adaptive weights, SI and ST to construct joint similarity matrix S.
Real-valued features are transformed into hash features through the attention mechanism.
Construct the similarity matrix BI and BT of hash features.
Construct the joint similarity matrix of hash matrix.
Calculate the objective function, backpropagate the gradient with chain rules, and update the whole parameter;
end for
return $F* = E(*, \theta*), * \in \{I, T\}$.

4 Experiments

4.1 Datasets

MIRFlickr (Huiskes and Lew, 2008) originates from the Flickr website, and the original MIRFLICKR 25K dataset comprises 25,000 image-text pairs, each belonging to multiple categories. In the experiment, only samples with text modalities of at least 20 words are used, resulting in a total of 20,015 image-text pairs, each belonging to 24 categories. The text is encoded into a 1386-dimensional Bag-of-Words (BOW) feature vector. The NUS-WIDE (Chua et al., 2009) dataset contains 269,648 network images, each accompanied by user-provided auxiliary tags, with each image labeled with one or more categories out of 81. Following the experimental setup in SSAH (Jin et al., 2020), image-text pairs without labels or auxiliary markers were removed, resulting in the selection of 190,421 pairs corresponding to 21 common categories. From this, 2,100 samples were randomly chosen as the query set, with the remaining 188,321 samples used as the database, and a further 10,500 samples randomly selected from the database for the training set.

Dataset	Training set	Database	Search set	Class	Image	Text
MIRFLICKR-25K	10000	18015	2000	24	4096-D	1368-D
NUS-WIDE	10500	188321	2100	81	4096-D	1000-D

Table 1: Statistics of the experimental dataset ("D" represents the feature dimension of the modality)

4.2 Evaluation Metrics

The most commonly used precision and recall rates are used as evaluation indicators in the experiment, and the corresponding top-K-precision curve can be obtained by using the obtained Precision and recall rates (abscissa: Top-k; Ordinate: precision).

The average precision AP value is the area under the precision-recall curve obtained by integrating, and the mAP value is the average AP value of all categories, calculated as follows:

$$AP(i) = \int_0^1 p(r)dr \tag{10}$$

$$mAP = \frac{\sum_{i=1}^{M} AP(i)}{M} \tag{11}$$

Where i refers to the class sequence number and M refers to the total number of classes.

4.3 Comparative experiment

Experimental Details:JSGHCA employs stochastic gradient descent optimizer with a momentum of 0.9 and weight decay of 0.0005. Evaluation during testing phase is conducted by calculating mAP values and Top-k-Precision curves. The overall evaluation criterion encompasses the mAP for image retrieval with text query and the mAP for text retrieval with image query. Parameter configuration should comprehensively consider the final experimental performance and training efficiency. After training the dataset for 10 epochs, the model demonstrates optimal training efficiency and retrieval performance. Learning rates are set to 0.001 for ImgNet and 0.01 for TxtNet, with a batch size of 32 chosen to balance performance and training efficiency. Hyperparameters are cross-validated, with $\eta = 0.4$ and μ=1.5 selected. For the NUS-WIDE dataset, $\beta = 0.6$ and $\lambda_1 = \lambda_2 = 0.1$. For the MIRFlickr dataset, $\beta = 0.9$ and $\lambda 1 = \lambda 2 = 0.1$.

Retrieval performance:To verify the effectiveness of the JSGHCA method, it was compared with state-of-the-art cross-modal hash retrieval methods, including AAH (Fang et al., 2021), HSIDHN (Chen et al., 2021), HNH (Zhang et al., 2021), DSPH (Yun et al., 2022), FDDH (Xin et al., 2021), SDDH (Qin et al., 2021), NSDH (Yang et al., 2021), MLSPH (Zhang et al., 2021), MSDH (Zhu et al., 2021), $MSLF_2$ (Song et al., 2021), QDCMH (Liu et al., 2021) and IRGR (Li et al., 2023). The results show that JSGHCA performs well on both the MIRFlickr and NUS-WIDE datasets. Although it may not achieve the best results in single searches, combining the mAP values of both search methods, JSGHCA and HNH perform the best. Compared to other advanced methods, JSGHCA demonstrates competitive performance on large datasets, indicating its effectiveness and robustness.

Method	I2T				T2I			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
AAH	0.7145	0.7230	0.7271	0.7283	0.8137	0.8198	0.8251	0.8281
DSPH	0.6473	0.6610	0.6703	-	0.6581	0.6781	0.6818	-
FDDH	-	0.7392	0.7578	0.7631	-	0.8022	0.8250	0.8357
HNH	-	0.8830	0.8950	0.9020	-	0.8540	0.8680	0.8780
HSIDHN	0.7978	0.8097	0.8179	-	0.7802	0.7946	0.8115	-
HSIDHN	0.7978	0.8097	0.8179	-	0.7802	0.7946	0.8115	-
MLSPH	0.8076	0.8235	0.8337	-	0.7852	0.8041	0.8146	-
MSDH	0.7836	0.7905	0.8017	-	0.7573	0.7635	0.7813	0.7959
$MSLF_2$	0.6988	0.7175	0.7222	0.7294	0.7572	0.7763	0.7892	0.8229
NSDH	0.7363	0.7561	0.7656	0.7712	0.7836	0.8014	0.8183	-
QDCMH	0.7635	0.7688	0.7713	-	0.7762	0.7725	0.7859	0.8328
IRGR	0.8310	0.8550	0.8770	0.8940	0.8250	0.8770	0.8820	0.8920
JSGHCA	0.830	**0.8897**	**0.9016**	0.9012	**0.8401**	**0.8832**	**0.9053**	**0.9102**

Table 2: Performance comparison of various methods on MIRFlickr dataset

It should be pointed out that IRGR method ,HNH method and JSGHCA method adopt the method of constructing similarity matrix for two modalities respectively, and further process the similarity matrix on this basis, but the processing methods of the similarity matrix are different between the two methods. The experimental results also show that IRGR method, HNH method and JSGHCA method have the best

Method	I2T				T2I			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
AAH	0.6409	0.6439	0.6515	0.6549	0.7379	0.7533	0.7595	0.7629
FDDH	-	0.6970	0.6910	0.7118	-	0.8133	0.8111	0.8244
HNH	-	0.8020	0.8160	0.8470	-	0.7760	0.7960	0.8020
HSIDHN	0.6498	0.6787	0.6834	-	0.6396	0.6529	0.6792	-
MLSPH	0.6405	0.6604	0.6734	-	0.6433	0.6633	0.6724	-
MSDH	0.6633	0.6859	0.7155	-	0.6359	0.6632	0.6934	-
$MSLF_2$	0.6213	0.6339	0.6374	0.6482	0.7212	0.7427	0.7578	0.7765
NSDH	0.6418	0.6604	0.6732	0.6791	0.7658	0.7892	0.7939	0.8011
SDDH	0.6510	0.6564	0.6670	0.6733	0.7638	0.7790	0.7945	0.7990
IRGR	0.7560	0.7930	0.8160	0.8390	0.7500	0.7830	0.8040	0.8170
JSGHCA	**0.7700**	**0.8203**	**0.8278**	**0.8602**	**0.7903**	**0.8357**	**0.8451**	**0.8500**

Table 3: Performance comparison of various methods on NUSWIDE dataset

performance among all comparison methods, which indicates that the construction of similarity matrix is conducive to the improvement of performance, and also indicates that the fine-grained processing of similarity matrix by JSGHCA method is logically reasonable and significantly improves the experimental performance.

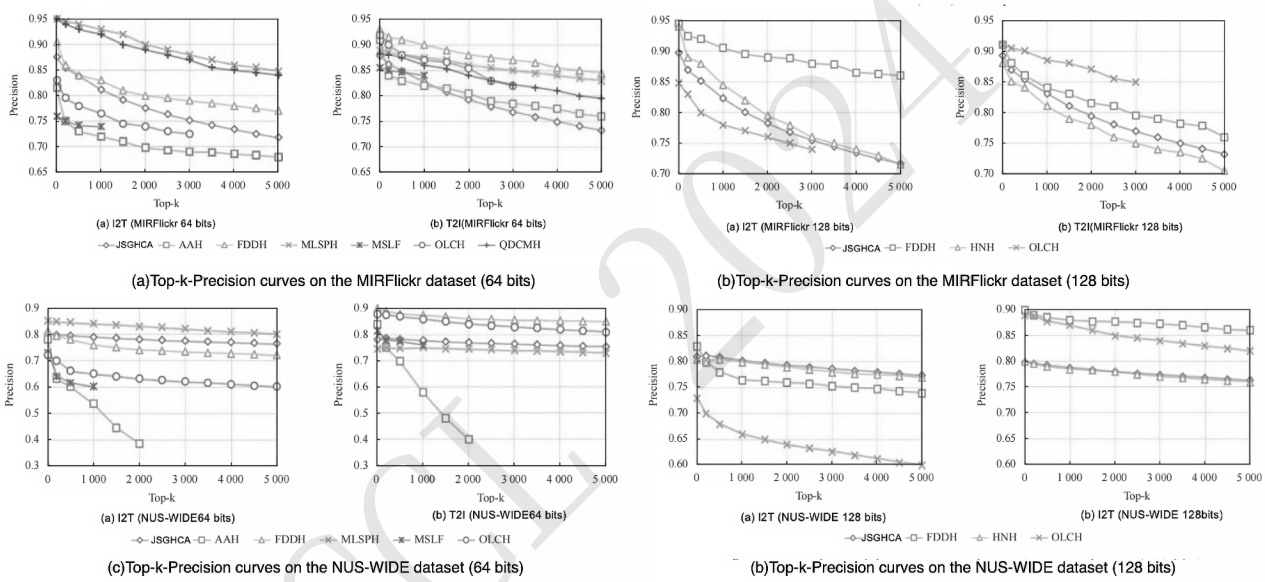

Figure 4: Top-k-Precision curves on the MIRFlickr and NUS-WIDE dataset (64 bits and 128bits)

To further compare the JSGHCA method with other cross-modal hash retrieval methods, we use a Top-k-Precision curve test. Here we compare the results of 64-bit and 128-bit hash lengths on MIRFlickr and NUS-WIDE datasets by several mainstream methods, as shown in Figure 4. As can be seen from the curve in the figure, although the accuracy of JSGHCA method on the same breakpoint may not be the highest on MIRFlickr dataset and NUS-WIDE dataset, it has relatively high performance and great advantages among similar methods, and is generally competitive.

4.4 Ablation Experiments

To demonstrate the effect of various innovations of the JSGHCA method, ablation experiments were performed on the data set. The ablation experiments of each module were conducted on the basis of JSGHCA, and their configurations were shown in Table 4. The JSGHCA-1 and JSGHCA-2 methods seperately used image similarity matrix and text similarity matrix to replace the joint semantic affinity matrix. The JSGHCA-3, JSGHCA-4, JSGHCA-5 and JSGHCA-6 methods do not use adaptive weight mixing strategy, Si-Attention Mechanism, Jo-Attention Mechanism and Collaborative Attention Mechanism, respectively.DJSRH is a benchmark model without adaptive weight strategy and collaborative attention Mechanism.

modal	setting
JSGHCA-1	$\| S_I - cos(B_I, B_T) \|_F^2$
JSGHCA-2	$\| S_T - cos(B_I, B_T) \|_F^2$
JSGHCA-3	No-AWM
JSGHCA-4	No-Lo-Attention Mechanism
JSGHCA-5	No-Co-Attention Mechanism
JSGHCA-6	No Mechanism

Table 4: Configuration of JSGHCA ablation experiments

Table 5-6 shows the experimental results of the ablation experiment. It can be seen from the data in the table that the JSGHCA-1 and JSGHCA-2 methods neglect the interaction and association between different modalities, focusing solely on the feature similarity within a single modality. Consequently, they may not perform well in multimodal retrieval tasks. JSGHCA-3 does not dynamically adjust weights based on the importance of different modalities, which may result in the inadequate utilization of information from various modalities.JSGHCA-4 and JSGHCA-5 methods neglect the importance of capturing crucial associations within a single modality (for JSGHCA-4) and across different modalities (for JSGHCA-5). Meanwhile, JSGHCA-6 ignores associations both within a single modality and across modalities as a whole, thus impeding its ability to fully comprehend multimodal data and generate accurate hash features.

Method	I2T				T2I			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
DJSRH	0.810	0.843	0.862	0.876	0.786	0.822	0.835	0.847
JSGHCA-1	0.6670	0.7082	0.7323	0.7745	0.6542	0.6835	0.7344	0.7632
JSGHCA-2	0.6634	0.6966	0.7222	0.7422	0.5573	0.5623	0.6438	0.6532
JSGHCA-3	0.8122	0.8623	0.8755	0.8802	0.8113	0.8582	0.8644	0.8723
JSGHCA-4	0.8186	0.8732	0.8911	0.8887	0.8096	0.8654	0.8873	0.8912
JSGHCA-5	0.8108	0.8555	0.8700	0.8864	0.8092	0.8487	0.8673	0.8774
JSGHCA-6	0.7902	0.8538	0.8792	0.8805	0.7756	0.8324	0.8606	0.8762
JSGHCA	**0.8300**	**0.8897**	**0.9016**	**0.9012**	**0.8401**	**0.8832**	**0.9053**	**0.9102**

Table 5: Comparison results on the MIRFlickr dataset

Method	I2T				T2I			
	16bits	32bits	64bits	128bits	16bits	32bits	64bits	128bits
DJSRH	0.724	0.773	0.798	0.817	0.712	0.744	0.771	0.789
DJSRH$_{S=S_I}$	-	-	0.717	0.741	-	-	0.712	0.735
DJSRH$_{S=S_T}$	-	-	0.702	0.734	-	-	0.606	0.581
JSGHCA-1	0.6753	0.6952	0.7265	0.7803	0.7098	0.7320	0.7696	0.7754
JSGHCA-2	0.6792	0.6866	0.7024	0.7865	0.7521	0.7542	0.7853	0.8042
JSGHCA-3	0.7423	0.7855	0.8025	0.8244	0.7652	0.7843	0.8413	0.8424
JSGHCA-4	0.7322	0.7753	0.7902	0.8265	0.7756	0.8324	0.8366	0.8382
JSGHCA-5	0.7532	0.7462	0.8321	0.8902	0.7653	0.8242	0.8234	0.8253
JSGHCA-6	0.7676	0.8032	0.8153	0.8439	0.7522	0.7900	0.8102	0.8199
JSGHCA	**0.7700**	**0.8203**	**0.8278**	**0.8602**	**0.7903**	**0.8357**	**0.8451**	**0.8500**

Table 6: Comparison results on the NUSWIDE dataset

To sum up , the shortcomings of these ablation experiments suggest that various innovations in JSGHCA are critical to improving the performance and accuracy of multimodal retrieval tasks. By considering the interaction and association of different modalities, using adaptive weight mixing strategies, and introducing Si-Attention, Jo-Attention, and collaborative-attention mechanisms, JSGHCA is able to better understand multimodal data and generate accurate hash features, thus achieving excellent performance in multimodal retrieval tasks.

5 Conclusion

In this paper, we introduced an improved deep joint semantics reconstructing hashing algorithm called JSGHCA. This algorithm combines adaptive weight strategies and attention mechanisms to enhance the

accuracy and descriptiveness of hash encodings for image and text modalities. Experimental results demonstrate that JSGHCA outperforms existing algorithms in multi-modal retrieval tasks, particularly with the incorporation of attention mechanisms and adaptive weight strategies. Additionally, we believe there is further room for exploration, such as integrating more advanced pre-trained models or optimizing attention mechanisms. We look forward to seeing the application of JSGHCA in other tasks, such as image and text generation, multi-modal translation, and beyond. JSGHCA opens up new possibilities in multi-modal retrieval and sets a new benchmark for future research.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62062055, Inner Mongolia University Youth Science and Technology Talent under Grant NJYT24061 and the basic Scientific Research Services of Colleges and Universities directly under Inner Mongolia Autonomous under Grant JY20220249.

References

Chen S, Wu S., Wang L. Hierarchical semantic interaction-based deep hashing network for cross-modal retrieval. PeerJ. Computer science, 2021, 7:e552.

Cheng X, Liu Z, Zhang Q. MSLF: Multi-scale legibility function to estimate the legible scale of individual line features. Cartography and Geographic Information Science, 2021, 48(2), 151-168.

Chua T S, Tang J, Hong R, et al. NUS-wide: a real-world web image database from National University of Singapore. Proceedings of the ACM international conference on image and video retrieval, 2009, pp. 1-9.

Fang X, Jiang K, Han N, et al. Average Approximate Hashing-Based Double Projections Learning for Cross-Modal Retrieval. IEEE Transactions on Cybernetics, 2021,PP(99), pp. 1-14.

Gálvez-López D, Tardos J D. Bags of binary words for fast place recognition in image sequences. IEEE Transactions on robotics, 2012, 28(5), pp. 1188-1197.

Huiskes M J, Lew M S. The mirflickr retrieval evaluation. Proceedings of the 1st ACM international conference on Multimedia information retrieval, 2008, pp. 39-43.

Jia M, Zhai Y, Lu S, et al. A similarity inference metric for RGB-infrared cross-modality person re-identification. arXiv preprint arXiv:2007.01504, 2020.

Jin S, Zhou S, Liu Y, et al. SSAH: Semi-supervised adversarial deep hashing with self-paced hard sample generation. Proceedings of the AAAI conference on artificial intelligence, 2020, 34(07), pp. 11157-11164.

Krizhevsky A, Sutskever I, Hinton G E. Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems, 2012, 25.

Li Z X, Hou C W, Xie X M. Unsupervised Cross-modal Hash Retrieval Fusing Multiple Instance Relations. Ruan JianXue Bao/Journal of Software, 2023, 34(11), pp. 4973–4988 (in Chinese).

Liu H, Xiong J, Zhang N, et al. Quadruplet-based deep cross-modal hashing. Computational Intelligence and Neuroscience, 2021, 2021.

Qin J, Fei L, Zhu J, et al. Scalable Discriminative Discrete Hashing For Large-Scale Cross-Modal Retrieval. ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2021.

Song W, Huan Z, Kei N. Learning a maximized shared latent factor for cross-modal hashing. Knowledge-Based Systems, 2021, 228.

Su Shupeng, Zhisheng Zhong, and Chao Zhang. Deep joint-semantics reconstructing hashing for large-scale unsupervised cross-modal retrieval. Proceedings of the IEEE/CVF International Conference on Computer Vision. 3027-3035, 2019.

Wang W, Shen Y, Zhang H, et al. Set and rebase: Determining the semantic graph connectivity for unsupervised cross-modal hashing. Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence, 2021, pp. 853-859.

Xin L, Xingzhi W, YiuMing C. FDDH: Fast Discriminative Discrete Hashing for Large-Scale Cross-Modal Retrieval. IEEE transactions on neural networks and learning systems, 2021.

Yang D, Wu D, Zhang W, et al. Deep semantic-alignment hashing for unsupervised cross-modal retrieval. Proceedings of the 2020 international conference on multimedia retrieval, 2020, pp. 44-52.

Yang Z, Yang L, Raymond, O I, et al. NSDH: A nonlinear supervised discrete hashing framework for large-scale cross-modal retrieval. Knowledge-Based Systems, 2021, 217, 106818.

Yi J, Liu X, Cheung Y, et al. Efficient online label consistent hashing for large-scale cross-modal retrieval. 2021 IEEE International Conference on Multimedia and Expo (ICME). IEEE, 2021, pp. 1-6.

Yu J, Zhou H, Zhan Y, et al. Deep graph-neighbor coherence preserving network for unsupervised cross-modal hashing. Proceedings of the AAAI conference on artificial intelligence, 2021, 35(5), pp. 4626-4634.

Yun L, Shujuan J, Qiang F, et al. An efficient dual semantic preserving hashing for cross-modal retrieval. Neurocomputing, 2022, 492, pp. 264-277.

Zhang J, Peng Y, Yuan M. Unsupervised generative adversarial cross-modal hashing. Proceedings of the AAAI conference on artificial intelligence, 2018, 32(1).

Zhang P, Luo Y, Huang Z, et al. High-order nonlocal Hashing for unsupervised cross-modal retrieval. World Wide Web, 2021, 24(2), pp. 1-21.

Zhang X, Wang X, E, M, et al. Multi-label semantics preserving based deep cross-modal hashing. Signal Processing: Image Communication, 2021, 93.

Zhu L, Tian G, Wang B, et al. Multi-attention based semantic deep hashing for cross-modal retrieval. Applied Intelligence, 2021, 51(8), pp. 1-13.