软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2024,35(3):1074−1089 [doi: 10.13328/j.cnki.jos.007076]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于细粒度特征融合的部分多模态哈希*

殷崭祚[1], 李博涵[1,2,3], 王 萌[4], 黄瑞龙[1], 吴文隆[1], 王昊奋[4]

[1](南京航空航天大学 计算机科学与技术学院, 江苏 南京 211106)

[2](软件新技术与产业化协同创新中心, 江苏 南京 211106)

[3](空天地海一体化大数据应用技术国家工程实验室 (西北工业大学), 陕西 西安 710119)

[4](同济大学 设计创意学院, 上海 200092)

通信作者: 李博涵, E-mail: bhli@nuaa.edu.cn

摘 要: 多模态数据的指数级增长使得传统数据库在存储和检索方面遇到挑战, 而多模态哈希通过融合多模态特征并映射成二进制哈希码, 能够有效地降低数据库的存储开销并提高其检索效率. 虽然目前已经有许多针对多模态哈希的工作取得了较好的效果, 但是仍然存在着 3 个重要问题: (1) 已有方法偏向于考虑所有样本都是模态完整的, 然而在实际检索场景中, 样本缺失部分模态的情况依然存在; (2) 大多数方法都是基于浅层学习模型, 这不可避免地限制了模型的学习能力, 从而影响最终的检索效果; (3) 针对模型学习能力弱的问题已提出了基于深度学习框架的方法, 但是它们在提取各个模态的特征后直接采用了向量拼接等粗粒度特征融合方法, 未能有效地捕获深层语义信息, 从而弱化了哈希码的表示能力并影响最终的检索效果. 针对以上问题, 提出了 PMH-F^3 模型. 该模型针对样本缺失部分模态的情况, 实现了部分多模态哈希. 同时, 基于深层网络架构, 利用 Transformer 编码器, 以自注意力方式捕获深层语义信息, 并实现细粒度的多模态特征融合. 基于 MIR Flickr 和 MS COCO 数据集进行了充分实验并取得了最优的检索效果. 实验结果表明: 所提出的 PMH-F^3 模型能够有效地实现部分多模态哈希, 并可应用于大规模多模态数据检索.

关键词: 部分多模态哈希; 多模态数据检索; 细粒度特征融合

中图法分类号: TP301

中文引用格式: 殷崭祚, 李博涵, 王萌, 黄瑞龙, 吴文隆, 王昊奋. 基于细粒度特征融合的部分多模态哈希. 软件学报, 2024, 35(3): 1074−1089. http://www.jos.org.cn/1000-9825/7076.htm

英文引用格式: Yin ZZ, Li BH, Wang M, Huang RL, Wu WL, Wang HF. Partial Multimodal Hashing Based on Fine-grained Feature Fusion. Ruan Jian Xue Bao/Journal of Software, 2024, 35(3): 1074−1089 (in Chinese). http://www.jos.org.cn/1000-9825/7076.htm

Partial Multimodal Hashing Based on Fine-grained Feature Fusion

YIN Zhan-Zuo[1], LI Bo-Han[1,2,3], WANG Meng[4], HUANG Rui-Long[1], WU Wen-Long[1], WANG Hao-Fen[4]

[1](College of Computer Science and Technology, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China)

[2](Collaborative Innovation Center of Novel Software Technology and Industrialization, Nanjing 211106, China)

[3](National Engineering Laboratory for Integrated Aero-Space-Ground-Ocean Big Data Application Technology (Northwestern Polytechnical University), Xi'an 710119, China)

[4](College of Design and Innovation, Tongji University, Shanghai 200092, China)

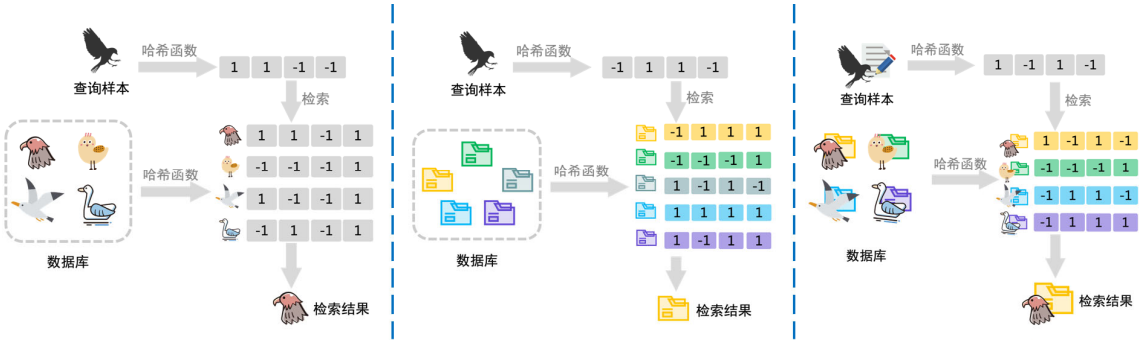
Abstract: Due to the exponential growth of multimodal data, traditional databases are confronted with challenges in terms of storage and

* 基金项目: 国家重点研发计划(2020YFB1708100); "十四五"民用航天技术预先研究项目(D020101); 国家自然科学基金(62172351); 高安全系统的软件开发与验证技术工业和信息化部重点实验室资助项目(NJ2018014); 河北省软件工程重点实验室项目
本文由"面向多模态数据的新型数据库技术"专题特约编辑彭智勇教授、高云君教授、李国良教授、许建秋教授推荐.
收稿时间: 2023-07-17; 修改时间: 2023-09-05; 采用时间: 2023-10-24; jos 在线出版时间: 2023-11-08
CNKI 网络首发时间: 2023-12-26

retrieval. Multimodal hashing is able to effectively reduce the storage cost of databases and improve retrieval efficiency by fusing multimodal features and mapping them into binary hash codes. Although many works on multimodal hashing perform well, there are also three important problems to be solved: (1) Existing methods tend to consider that all samples are modality-complete, while in practical retrieval scenarios, it is also common for samples to miss partial modalities; (2) Most methods are based on shallow learning models, which inevitably limits models' learning ability and affects the final retrieval performance; (3) Some methods based on deep learning framework have been proposed to address the issue of weak learning ability, but they directly use coarse-grained feature fusion methods, such as concatenation, after extracting features from different modalities, which fails to effectively capture deep semantic information, thereby weakening the representation ability of hash codes and affecting the final retrieval performance. In response to the above problems, the PMH-F^3 model is proposed. This model implements partial multimodal hashing for the case of samples missing partial modalities. The model is based on deep network architecture, and the Transformer encoder is used to capture deep semantics in self-attention manner, achieving fine-grained multimodal feature fusion. Sufficient experiments are conducted on MIR Flickr and MS COCO datasets and the best retrieval performance is achieved. The results of experiments show that PMH-F^3 model can effectively implement partial multimodal hashing and can be applied to large-scale multimodal data retrieval.

Key words: partial multimodal hashing; multimodal data retrieval; fine-grained feature fusion

大数据和人工智能的快速发展正掀起新一轮的信息革命，传统信息系统也正通过知识赋能进行数字化转型[1]，这不仅导致各行各业数据量的爆炸式增长，数据格式和类型也变得愈加丰富. 与传统的单模态数据相比，多模态数据可以提供更加丰富的信息表示，且基于多模态数据表示也有着更为广泛的应用[2]，如视觉问答、智能医疗和情感分析等. 与此同时，海量的多模态数据给传统数据库带来了检索方面的挑战. 与基于真实值的检索方法相比，基于哈希的检索方法通过将高维数据映射成紧凑的二进制哈希码，从而能够大幅度降低大规模数据的存储开销并提高数据库的检索效率. 为此，基于哈希的数据检索方法被提出. 按照样本所涉及的模态进行划分，基于哈希的检索方法可以分为单模态哈希[3−5]、跨模态哈希[6−8]和多模态哈希[9−14]. 如图 1 所示，多模态哈希与传统的单模态哈希和跨模态哈希针对的应用场景不同：单模态哈希涉及的所有样本均属于同一模态，例如通过图像样本检索数据库中其他相似的图像样本；跨模态哈希涉及的查询样本和检索结果则属于两种不同的模态，例如将图像样本映射成哈希码并检索数据库中与之语义相似的文本样本；而多模态哈希涉及的查询样本和检索结果都包涵多个模态，例如将图像和文本样本作为复合查询统一映射成哈希码，从而检索数据库中与之语义相似的图像和文本样本. 此外，跨模态哈希和多模态哈希的目标也有所差别：跨模态哈希的目标是确定一个共享空间，同一个样本的不同模态特征经过投影后在该共享空间中具有相似性；而多模态哈希旨在确定一个多模态互补空间，基于该空间充分挖掘各个模态间的互补信息，从而获得更全面的表示[15]. 因此，较之跨模态哈希，多模态哈希需要更加关注如何协同融合不同模态的特征.



(a) 单模态哈希 (b) 跨模态哈希 (c) 多模态哈希

图 1 3 种不同的基于哈希的检索方法

近年来涌现了一些针对多模态哈希的方法，但其中大部分方法更偏向于考虑所有样本在训练和查询阶段都是模态完整的[16−18]，而缺失部分模态的情况对于数据库中存储的多模态样本而言同样常见. 例如在真实的社交网络场景中，数据由各种用户自主上传，有些用户上传图像数据但缺失相应的文本信息，而有些用户上

传描述的文本信息但缺失相应的图像数据, 因此产生了大量的部分多模态数据[19], 从而在一定程度上限制了这些方法的实际应用. 如果舍弃不完整模态的样本而只利用模态完整的样本, 会大大减少训练样本的数量, 从而弱化最终学习到的哈希码的表示能力并影响检索效果. 因此, 需要依靠部分多模态哈希来应对样本模态不完整的情况.

除此之外, 大多数现有的针对多模态哈希的方法基于浅层学习模型, 这些方法受限于浅层模型的学习能力而无法有效捕获多模态数据中隐含的深层语义信息[20−22]. 虽然已经有部分方法[17,23]使用预训练的深层网络提取各个模态的原始特征信息, 并采用特定的策略融合所提取的多模态特征以生成哈希码, 但是它们忽略了对深层语义信息的建模, 导致不能充分发挥深度神经网络强大的学习能力. 具体而言, 这些方法利用深度网络模型提取多模态特征后, 直接采用简单的向量拼接或相加等方式进行模态特征融合, 但提取的特征仅能表示浅层语义信息, 采取的特征融合方式也只属于粗粒度的融合, 因此会弱化最终哈希码的表示能力并影响检索效果. 事实上, 每种模态的数据都蕴含丰富的语义信息, 不同语义信息之间也蕴含着相应的语义关系, 在浅层语义信息的基础上进一步学习, 能够提取更深层次的语义信息. 例如: 给定一张鹰的图片, 使用神经网络能够轻易提取到浅层的语义信息, 即该物体属于“鹰”这一类别; 如果进一步对浅层语义信息进行学习, 能够提取到深层语义信息, 即鹰属于一种鸟类. 那么, 所有有关鸟的图片在映射成哈希码时应该存在相似性. 此外, 不同模态的特征可以从多个角度描述样本并共享深层语义信息. 基于这些共享的深层语义信息, 样本的不同模态特征可以进行潜在对齐[16], 从而实现细粒度的多模态特征融合.

为了解决上述问题, 本文提出一种新的模型, 即基于细粒度特征融合的部分多模态哈希(partial multimodal hashing based on fine-grained feature fusion). 该模型能够有效应对样本缺失部分模态的情况, 并通过捕获深层语义信息实现细粒度的多模态特征融合. 具体而言: (1) 当样本缺失部分模态时, 利用缺失模态补齐模块能够补齐所缺失的模态特征; (2) 提出的模型完全基于深度神经网络架构, 利用 Transformer 编码器[24], 以自注意力机制为核心, 能够更有效地捕获深层语义信息并实现细粒度特征融合. 本文的主要贡献如下.

- 提出了一种部分多模态哈希模型, 简称为 PMH-F^3. 该模型充分利用所有样本, 用训练后的同模态补齐模块监督跨模态生成模块, 减少了补齐缺失模态特征时引入的噪声, 能够有效应对样本缺失部分模态的多模态数据检索场景.
- 设计了一种深层网络框架, 有效利用深度学习的强大能力捕获深层语义信息, 最终经过细粒度特征融合, 将多模态样本映射为二进制哈希码.
- 在两个广泛测试的多模态数据检索任务的数据集上进行了充分的实验, 并在整体上取得了最优的效果. 实验结果验证了本文所提出模型的优越性.

1 相关工作

1.1 多模态哈希

按照是否有监督的分类方式, 可将多模态哈希分为基于无监督学习的方法和基于有监督学习的方法. 基于无监督学习的方法不依赖类别标签和其他语义信息, 直接学习每个多模态样本对应的哈希码. Song 等人[9]提出了 MFH (multiple feature hashing)方法. 该方法能够保留每个特征的局部结构并一起考虑这些结构信息, 以此学习一组哈希函数, 将视频关键帧映射到汉明空间并生成相应的二进制代码, 从而支持近似重复的视频检索. Liu 等人[10]提出了无监督多视图对齐的方法 MAH (multiview alignment hashing). 该方法基于非负矩阵分解, 能够融合多个信息源, 并在找到一个紧凑的二进制代码的同时保留联合概率分布. Shen 等人[11]提出了 MVLH (multi-view latent hashing)方法. 该方法能够从统一核特征空间中学习二进制代码, 并根据每个视图的重建误差自适应地学习不同视图的权重. 此外, Shen 等人[12]还提出了 MvDH (multiview discrete hashing)方法. 该方法利用矩阵分解来生成哈希码, 并在此期间同时执行频谱聚类, 通过哈希码和聚类标签的联合学习增强所生成的哈希码的鉴别性.

不同于无监督学习的方法, 基于有监督学习的多模态哈希在生成哈希码时利用了样本的类别标签等其他

语义信息. Liu 等人[13]提出, 多特征哈希可以被公式化为具有最优线性组合的多核相似性保持问题, 并提出了一种有效的交替优化算法来学习哈希函数和最优核组合. Yang 等人[14]提出了 DMVH (discrete multi-view hashing)方法, 该方法直接利用多视图数据的丰富特征信息进行处理. 同时, 提出了一种新的相似矩阵构造方法以保留样本间的局部相似结构和语义相似性. 与之不同, Lu 等人[20]提出了名为 FOMH 的在线多模态哈希的方法, 该方法能够自适应地计算权重并以加权求和方式融合多模态特征, 并使用非对称监督方式学习哈希码. 同时, Lu 等人[21]提出的 SDMH (supervised discrete multi-view hashing)方法通过结合多视图数据进行潜在特征学习, 并通过融合视觉特征和灵活语义信息来学习一致的哈希码. Liu 等人[22]提出了名为 FDMH 的自适应多视图分析学习模型, 该模型能够巧妙地将不同模态的特征表示组合到一个潜在的公共特征空间中, 并在该空间中利用自主多视图加权策略, 很好地探索不同视图的互补性.

大多数基于浅层学习模型的多模态哈希主要依赖线性映射或矩阵分解对不同模态之间的语义信息进行建模. 相比之下, 基于深层学习模型的多模态哈希借助深层网络执行复杂的非线性投影, 以此提高多模态数据的检索性能. 同时, 基于深层学习模型的多模态哈希通过放松对哈希码的离散约束, 采用基于梯度下降的反向传播算法, 以此学习松弛的哈希码. Zhu 等人[23]首先提出了名为 DCMVH 的深度协同多视图哈希的方法, 该方法将不同的层分别与实例语义标签和成对语义标签相关联, 从而在深层网络架构下融合多视图特征并协同学习多视图哈希码. 与之不同, Tan 等人[16]提出了 BSTH (bit-aware semantic transformer hashing)方法, 该方法通过位感知的语义 Transformer 融合多模态特征并生成哈希码. 此外, 该方法能够通过挖掘可用的标签信息来监督模型的学习过程. Lu 等人[17]提出的 FGCMH (flexible graph convolutional multi-modal hashing)方法利用图卷积网络保持各个模态和融合模态间的结构相似性, 并用于学习二进制哈希码.

1.2 部分多模态哈希

在社交网络等实际场景下, 由于数据丢失、上传限制等因素, 样本往往不是模态完整的. 面对样本缺失部分模态的情况, 多模态哈希的检索效果不佳, 且直接舍弃不完整模态的样本显然不合理. 在跨模态哈希的研究中, 针对该问题提出了一些部分跨模态哈希的方法[25−27], 这些方法通过探究不同模态间的语义相关性来应对样本部分缺失模态情况下的跨模态哈希问题, 但是却较少有工作关注到部分多模态哈希. 据了解, 目前针对部分多模态哈希最具有代表性的两个工作分别是 SAPMH[28]和 NCH[29]. Zheng 等人[28]首先提出了名为 UAPMH 的无监督学习的方法, 该方法充分利用可用的局部视图, 并通过共享语义空间有效地保留哈希码中图像和文本的潜在关系. 为了进一步增强哈希码的判别能力, 研究者将 UAPMH 扩展到有监督学习范式, 提出了 SAPMH 方法. 但是该方法只使用样本的可用模态来生成哈希码, 未充分考虑其他模态, 从而可能导致所生成哈希码的语义不完整. Tan 等人[29]提出了一种名为 NCH 的方法, 该方法基于邻居节点来估算和补齐缺失模态. 但是该方法在生成哈希码时只使用了浅层网络模型, 因此不能捕获不同模态间的深层语义信息, 从而不能实现细粒度的特征融合.

1.3 缺失模态补齐

面对部分模态缺失的情况, 为了充分利用数据, 较为理想的方法是补齐样本所缺失的模态特征. 然而补齐的特征与实际值间必定存在误差, 从而导致引入额外噪声. 因此, 如何减少噪声的引入并准确补齐缺失模态特征成为研究重点. Wang 等人[30]提出了一种基于一致性 GAN (generative adversarial network)的部分多视图聚类方法. 与基于核方法或者非负矩阵分解的方法不同, 该方法利用某个视图的公共编码表示, 通过 GAN 生成相应视图的缺失数据. 虽然直观上 GAN 能够很好地适用于缺失模态特征的生成, 但是由于其网络结构的复杂性, 使得该模型的训练较为复杂, 较小的输入差异所引入的噪声也有较大波动. Guo 等人[31]提出了名为 APMC (anchor-based partial multi-view clustering)的部分多视图聚类方法, 该方法将具有完整视图的样本作为锚点, 计算样本点和锚点之间的相似性并融合视图内和视图间的相似性, 最后通过谱聚类的方法对融合以后的相似度矩阵进行聚类. Zeng 等人[32]面对缺失模态不定场景下的多模态情感分析问题提出了基于标签辅助的 TATE (tag-assisted transformer encoder)方法, 该方法没有直接补齐缺失模态特征, 而是用标签对样本缺失模态

的情况进行标记. 计算编码器编码后的特征与原始特征的差距以及解码器解码后的标签与原始标签的差距, 并最小化这种差距, 从而优化编码器对所有模态特征进行编码的效果. Ma 等人[33]针对样本严重缺失部分模态的情况(缺失部分模态的样本占总样本的比例高达 90%), 提出了基于贝叶斯元学习的 SMIL (severely missing modality)方法. 该方法对潜在特征空间添加扰动, 从而使得单个模态的特征表示近似于全模态的特征表示.

虽然目前越来越多的研究关注到多模态领域中重要的模态缺失的问题, 并且已经有许多缺失模态补齐的方法在多模态学习和多视图聚类、多模态情感分析等应用领域取得了不错的效果, 但是针对多模态哈希方向的研究仍在发展当中. 本文基于缺失模态补齐模块, 在充分利用缺失模态样本的同时, 最大限度地减少补齐过程中引入的噪声, 以保证特征的真实性和准确性, 从而提高最终哈希码的表示效果.

2 基于细粒度特征融合的部分多模态哈希模型

2.1 部分多模态哈希定义

多模态哈希的任务在于: 当多模态样本到达时, 首先提取各个模态的特征并融合, 然后将融合后的特征转化为二进制的哈希码, 以支持基于汉明距离的大规模多模态数据的快速检索. 而部分多模态哈希应对的是训练和查询阶段样本模态不完整的情况. 本文提出的方法可扩展至多种模态, 但一般地, 本文仅以图像和文本这两种模态为例进行说明. 假设训练集$\mathcal{O}=\{\mathcal{O}^p,\mathcal{O}^v,\mathcal{O}^t\}$由 3 个部分组成, 其中, $\mathcal{O}^p=\{(x_i^p,y_i^p,l_i^p)\,|\,i\in[1,N_p]\}$表示训练集样本中具有完整模态的部分, $\mathcal{O}^v=\{(x_i^v,miss,l_i^v)\,|\,i\in[1,N_v]\}$表示仅有图像模态而缺失文本模态的部分, $\mathcal{O}^t=\{(miss,y_i^t,l_i^t)\,|\,i\in[1,N_t]\}$表示仅有文本模态而缺失图像模态的部分. 其中: $x_i^*\in\mathbb{R}^{1\times d_v}$, $y_i^*\in\mathbb{R}^{1\times d_t}$, $*\in\{p,v,t\}$分别表示对应样本集中第 i 个样本的图像模态特征和文本模态特征, 它们都是原始数据经过特征提取后形成的特定维度的向量表示; N_*, $*\in\{p,v,t\}$则表示对应样本集包含的样本数量. 因为每个多模态样本可以属于多个类别, 所以标签 l_i^* 是一个多类别向量, 即: 假设训练集一共有 n 个类别, 则 $l_i^*\in\{0,1\}^{1\times n}$, 其中, $l_{ij}^*=1$则表示第 i 个样本属于第 j 个类别, 反之, $l_{ij}^*=0$. 最终能够为用于查询和等待检索的每个多模态样本学习一个紧凑的二进制表示 $b_i^*\in\{-1,1\}^{1\times K}$, 其中, K 是哈希码的长度.

2.2 PMH-F^{3}总体架构

本文提出的 PMH-F^3 模型的总体架构如图 2 所示.

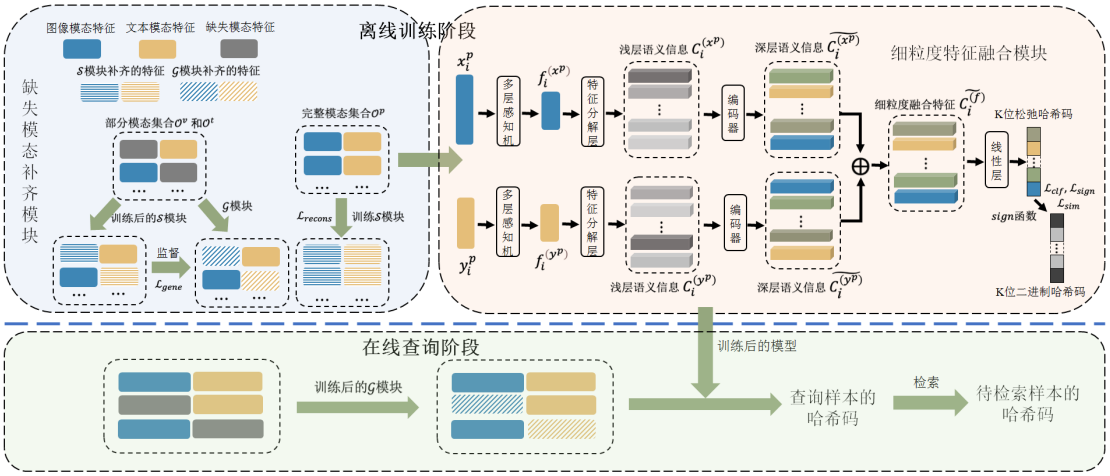


图 2 PMH-F^3 模型的总体架构

该模型分为离线训练阶段和在线查询阶段. 离线训练阶段主要由缺失模态补齐模块和细粒度特征融合模块组成. 对于缺失模态补齐模块, 模型预先在完整模态样本集中随机抽取若干具有完整模态的样本构成锚点

集, 若其余训练样本有任意一种模态缺失, 首先利用与之语义相似的锚点的同一模态特征补齐样本缺失的模态特征, 将这一部分记作同模态补齐模块(“$\mathcal{S}$”); 然后, 将补齐的模态特征作为“真实值”来监督训练依靠样本已有模态特征直接跨模态生成另一缺失模态特征, 将这一部分记作跨模态生成模块(“$\mathcal{G}$”). 对于细粒度特征融合模块, 模型首先将图像和文本特征映射成同一维度; 然后将特征分解成每个哈希位对应的 K 个浅层语义信息; 接着利用 Transformer 编码器模块对浅层语义信息进行建模, 利用自注意力方式自适应地捕捉浅层语义信息间的内在关系, 以此得到编码后的深层语义信息; 最后进行细粒度特征融合, 并映射成 K 位的二进制哈希码. 考虑在线查询阶段, 当新的多模态查询样本(或多个查询样本)到达时, 若查询样本有部分模态缺失, 则首先利用离线阶段训练的“$\mathcal{G}$”模块生成缺失的模态特征, 然后利用训练后的细粒度特征融合模块融合不同模态特征并生成该样本对应的哈希码, 以便将其应用于快速的多模态数据检索. 下面将具体介绍缺失模态补齐模块和细粒度特征融合模块.

2.3 缺失模态补齐模块

出于以下两个考虑: (1) 某种或多种模态特征相近的不同样本之间存在相似的语义关系; (2) 同一样本的不同模态之间存在相似的语义关系. 本文的模型利用第 2.2 节中提到的“$\mathcal{S}$”和“$\mathcal{G}$”模块联合进行缺失模态补齐. 其中: “$\mathcal{S}$”模块基于模态特征相近的不同样本之间存在的相似的语义关系, 利用与样本存在相似语义关系的锚点的同种模态特征进行缺失模态的补齐; 而“$\mathcal{G}$”模块则基于同一样本的不同模态之间存在相似的语义关系, 直接跨模态生成缺失的模态特征. 在补齐缺失模态的同时, 不可避免地会引入噪声. 为了尽可能地减少引入的噪声, 对于“$\mathcal{S}$”模块, 模型基于完整模态的样本集$\mathcal{O}^p$ 进行训练, 并且将训练后的“$\mathcal{S}$”模块拟补齐的缺失模态特征作为“真实值”来监督训练“$\mathcal{G}$”模块. 这样既能保证补齐的缺失模态特征尽可能地准确, 又能增加用于训练“$\mathcal{G}$”模块的样本数量. 值得注意的是: “$\mathcal{S}$”模块仅用于在离线阶段监督训练“$\mathcal{G}$”模块, 而在线查询阶段只利用训练后的“$\mathcal{G}$”模块进行缺失模态补齐. 这是由于锚点集中锚点的选择是随机的, 所以可能会出现下面的情况: 在查询阶段, 新样本与所有锚点恰好都不属于同一类别, 即新样本与所有锚点都不存在很强的语义相关性. 如果仅利用“$\mathcal{S}$”模块, 即使训练时完全依赖完整模态的样本集, 补齐的缺失模态特征与真实特征间也会存在较大偏差, 导致补齐缺失模态时引入较大的噪声. 同时, “$\mathcal{G}$”模块能够根据样本已有模态特征直接跨模态生成缺失模态特征, 而“$\mathcal{S}$”模块则需要通过度量特征之间的相似度来寻找与待补齐样本语义最相关的锚点, 因此, 只使用“$\mathcal{G}$”模块还能减少查询阶段所花费的时间.

本文统一考虑样本缺失图像模态而仅有文本模态的情况. 对于缺失文本模态而仅有图像模态的情况, 处理方法同样类似.

2.3.1 同模态补齐模块

假设某样本缺失图像模态而仅有文本模态, 同模态补齐模块“$\mathcal{S}$”旨在利用文本模态特征搜索其他与该样本语义相似的样本, 并用语义相似样本的图像模态特征来补齐该样本缺失的图像模态特征, 这一过程可以形式化地表述为

$$\widehat{x_i^t} = \mathcal{S}(y_i^t, Y^a, X^a; \theta_{\mathcal{S}}) \tag{1}$$

其中, y_i^t 是该样本的文本模态特征, $\widehat{x_i^t}$ 表示通过“$\mathcal{S}$”模块补齐的图像模态特征, $\theta_{\mathcal{S}}$ 是该模块中可训练的参数. 值得注意的是: 为了减小搜索范围, 需要首先从具有完整模态的样本集合$\mathcal{O}^p$ 中选取 N_a 个样本构成锚点集. 锚点的采样是随机的, 且锚点的个数应大于训练集中样本的种类数, 以此使得随机采样的锚点尽可能能够覆盖所有的样本种类, 即让样本在补齐缺失模态时尽可能与一个或者多个锚点具有较强语义相关性. 锚点集中的所有图像模态特征构成 $X^a \in \mathbb{R}^{N_a \times d_v}$, 所有文本模态特征构成 $Y^a \in \mathbb{R}^{N_a \times d_t}$. 这里介绍两种不同的方法来具体实现“$\mathcal{S}$”模块: 基于 K 最近邻(K-nearest neighbor, KNN)的方法和基于自注意力机制(self-attention)的方法.

(1) 基于 KNN 的方法

基于KNN的方法首先计算样本的文本模态特征y^t与锚点集中所有锚点的文本模态特征的距离, 然后基于距离进行排序, 距离越小, 则说明该样本与锚点集中对应锚点在文本模态上越接近, 从而说明该样本与该锚

点存在相似的语义关系. 然后对前 K 个语义最相似的锚点的图像模态特征计算平均值, 抑或是对距离的倒数归一化后作为权重进行加权求和, 则可以补齐该样本缺失的图像模态特征. 这一过程可以形式化地表述为

$$Distance(y_i^t,\{y_j^a\}_{j=1}^{N_a})\xrightarrow{\text{sort}}\{x_k^a,y_k^a\}_{k=1}^{K}\xrightarrow{\text{weight sum}}\widehat{x_i^t} \tag{2}$$

值得注意的是, 基于 KNN 的方法是无参数的.

(2) 基于自注意力机制的方法

自注意力机制[24]通过度量 query 和 key 之间的相似度作为注意力分数, 并对 value 进行加权求和作为编码结果, 以此来建模输入序列之间的相关性. 如图 3 所示, 这里可将其扩展为跨模态的自注意力机制. 具体而言, 将样本的文本模态特征 y_i^t 作为 query, 将锚点集的文本模态特征 Y^a 作为 key, 并将锚点集的图像模态特征 X^a 作为 value, 这可以形式化地表述为

$$q_i^t=y_i^tW_Q,\ K=Y^aW_K,\ V=X^aW_V \tag{3}$$

其中, W_Q, $W_K\in\mathbb{R}^{d_t\times d_k}$, $W_V\in\mathbb{R}^{d_v\times d_k}$, 都是可训练的参数; d_k 是特征嵌入后的维度.

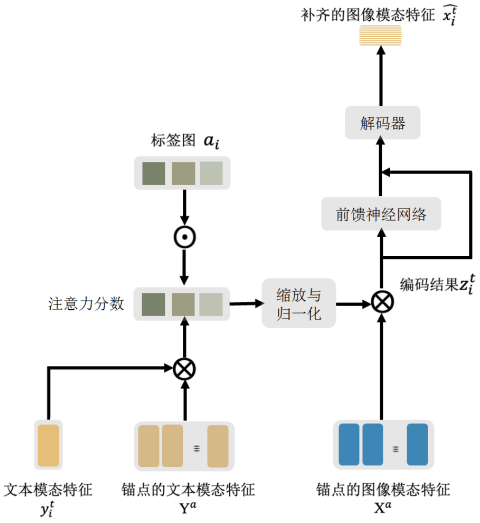


图 3 跨模态自注意力机制

接着, 计算 q_i^t 和 K 之间的相似度, 并对 V 进行加权求和, 得到编码后的结果. 此外, 为了提升基于自注意力机制的同模态补齐的效果, 本文引入了文献[29]中的标签图来优化注意力分数, 如式(4)所示.

$$z_i^t=softmax\left(\frac{q_i^tK^T}{\sqrt{d_k}}\odot a_i\right)V \tag{4}$$

其中, $\odot$ 是 Hadamard 乘积; z_i^t 是利用跨模态自注意力机制编码后的结果; a_i 是该样本与锚点集中所有锚点相似度的向量, 其每一个位置上的元素可以通过式(5)进行计算.

$$a_{ij}=\frac{2}{1+\mathrm{e}^{-l_i^t\cdot l_j^a}}-1 \tag{5}$$

类似于 Transformer 编码器[24]的架构, 之后将层归一化和前馈神经网络作用于编码器的输出. 最后, 用简单的全连接层作为解码器来产生最终的输出, 如式(6)所示.

$$\widehat{x_i^t}=\mathrm{Decoder}(z_i^t;\theta_{\mathrm{Decoder}}) \tag{6}$$

其中, $\theta_{\mathrm{Decoder}}$ 是解码器中可训练的参数; $\widehat{x_i^t}$ 是解码器的输出, 作为该模块补齐的图像模态特征.

2.3.2 跨模态生成模块

正如之前在第 2.3 节所提及, 由于锚点的选择是随机的, 因此会出现查询阶段新来的多模态样本与所有

锚点恰好都不属于同一类别的情况. 此时, “$\mathcal{S}$”模块补齐的缺失模态特征与真实特征会存在较大偏差, 导致引入较大的噪声. 与此同时, 同样基于上述所提及的考虑: 同一样本的不同模态之间存在相似的语义关系, 因此可以利用样本已有的模态特征直接跨模态生成缺失的模态特征. 本文将“$\mathcal{S}$”模块生成的图像模态特征作为“真实值”来监督训练跨模态生成模块“$\mathcal{G}$”, 旨在通过已有的文本模态特征直接跨模态生成缺失的图像模态特征. 跨模态生成的过程可以形式化地表述为

$$\widetilde{x_i^t} = \mathcal{G}(y_i^t;\theta_{\mathcal{G}}) \tag{7}$$

其中, $\widetilde{x_i^t}$ 表示由“$\mathcal{G}$”模块生成的样本缺失的图像模态特征, $\theta_{\mathcal{G}}$ 是该模块中可训练的参数. 值得注意的是: 本文只是简单地将“$\mathcal{G}$”模块设计为多个全连接层, 即通过非线性变换直接将样本的文本模态特征映射为图像模态特征. 虽然这种设计较为简单, 但是在“$\mathcal{S}$”模块生成的“真实值”监督下, 也可以对同一样本不同模态之间存在的相似语义关系进行准确建模. 此外, 简单的“$\mathcal{G}$”模块的设置能够减少在线查询阶段补齐查询样本缺失的模态所花费的时间, 进而有效缩短数据库的检索时间.

2.4 细粒度特征融合模块

基于浅层学习模型的多模态哈希无法有效捕获多模态数据中的深层语义信息, 同时, 直接通过向量相加或拼接的多模态融合方式仅属于特征层面上的粗粒度融合, 其融合效果不佳. 实际上, 每个哈希位都可以表示一种语义信息. 该模块首先将各个模态特征分解为 K 个向量, 代表 K 个浅层语义信息. 接着, 为了进一步捕获蕴含其中的深层语义信息, 为每个模态引入模态特定的 Transformer 编码器[24], 旨在以自注意力的方式对浅层语义信息间的关系进行建模, 从而自适应地捕捉深层语义信息. 之后, 对特征编码后的深层语义信息进行细粒度融合并, 最终映射到哈希码的每一位.

如图 2 所示, 为了减小细粒度特征融合模块的误差, 本文基于训练集中完整模态的样本集 $\mathcal{O}^p$ 来训练该模块. 对于 $\mathcal{O}^p$ 中的第 i 个样本 (x_i^p, y_i^p, l_i^p), 首先利用多层感知机(multi-layer perceptron, MLP)分别将不同模态特征映射成同一维度, 如式(8)所示.

$$f_i^{(\cdot)} = \mathrm{MLP}^{(\cdot)}(\cdot_i;\theta_{\mathrm{MLP}^{(\cdot)}}),\ \cdot \in \{x^p, y^p\} \tag{8}$$

其中, $\theta_{\mathrm{MLP}}{}^{(\cdot)}$是可训练参数, $f_i^{(\cdot)} \in \mathbb{R}^{1\times d_{\mathrm{common}}}$ 表示将图像和文本特征映射成同一维度后的特征向量.

之后, 将特征向量 $f_i^{(\cdot)}$ 分解成 K 个哈希位所对应的浅层语义信息的序列. 具体而言: 首先, 将 $f_i^{(\cdot)}$ 映射为 $K\times d_{\mathrm{common}}$ 维的向量; 然后, 将其重新转化为 $K\times d_{\mathrm{common}}$ 维的矩阵, 并记作 $C_i^{(\cdot)} = [c_{i_1}^{(\cdot)}, c_{i_2}^{(\cdot)}, ..., c_{i_K}^{(\cdot)}]$.

接着, 为每个模态引入模态特定的 Transformer 编码器[24]来捕获深层语义信息, 如式(9)所示.

$$\widetilde{C_i^{(\cdot)}} = \mathrm{Encoder}^{(\cdot)}(C_i^{(\cdot)};\theta_{\mathrm{Encoder}^{(\cdot)}}) \tag{9}$$

其中, Encoder 表示编码器模块. $\widetilde{C_i^{(\cdot)}} \in \mathbb{R}^{K\times d_{\mathrm{common}}}$ 表示编码器的输出, 即深层语义信息. 从形式上看, $\widetilde{C_i^{(\cdot)}} = [\widetilde{c_{i_1}^{(\cdot)}}, \widetilde{c_{i_2}^{(\cdot)}}, ..., \widetilde{c_{i_K}^{(\cdot)}}]$.

在捕获了深层语义信息后, 就可以将两种模态特征进行细粒度的特征融合, 如式(10)所示.

$$\widetilde{C_i^{(f)}} = \widetilde{C_i^{(x^p)}} + \widetilde{C_i^{(y^p)}} \tag{10}$$

其中, $\widetilde{C_i^{(f)}}$ 表示第 i 个样本进行多模态特征融合后的结果. 接着, 利用 K 个不同的映射函数将其逐位映射到对应哈希位上, 如式(11)所示.

$$h_i^p = (H_1(\widetilde{c_{i_1}^{(f)}};\theta_{H_1}), H_2(\widetilde{c_{i_2}^{(f)}};\theta_{H_2}), ..., H_K(\widetilde{c_{i_K}^{(f)}};\theta_{H_K})) \tag{11}$$

其中, $H_k(\widetilde{c_{i_k}^{(f)}};\theta_{H_k})$ 用于将第 k 个融合后的特征 $\widetilde{c_{i_k}^{(f)}}$ 映射为对应哈希位上的松弛值 $h_{i_k}^p$.

最后, 利用符号函数 $sign$ 将松弛的哈希码转化为最终的二进制表示, 如式(12)所示.

$$b_i^p = sign(h_i^p) \tag{12}$$

其中, $h_i^p \in \mathbb{R}^{1\times K}$, $b_i^p \in \{-1,1\}^{1\times K}$. 此外, 为了充分利用样本的标签信息, 本文在获得松弛的哈希码后, 利用其进

行简单的类别概率预测, 这可以形式化地表述为

$$\widetilde{l_i^p}=\sigma(\mathrm{FC}(h_i^p;\theta_{\mathrm{FC}})) \tag{13}$$

其中, FC 表示全连接层, θ_{FC} 是可训练参数, σ是 *sigmoid* 激活函数, $\widetilde{l_i^p}$ 的每一个元素都代表该样本属于对应类别的概率.

2.5 目标函数

为了减小缺失模态补齐模块中同模态补齐模块“$\mathcal{S}$”的训练误差, 使得“$\mathcal{S}$”模块补齐的特征与真实特征尽可能接近, 从而在训练跨模态生成模块“$\mathcal{G}$”时起到更好地监督效果, 本文在训练该模块时使用完整模态样本集 $\mathcal{O}^p$ 中的样本, 引入重构损失函数$\mathcal{L}_{\mathrm{recons}}$, 并将其定义为

$$\mathcal{L}_{\mathrm{recons}}=\|\widehat{x_i^p}-x_i^p\|_2^2+\|\widehat{y_i^p}-y_i^p\|_2^2 \tag{14}$$

其中, $\|\cdot\|_2$ 表示向量的 2-范数.

为了训练跨模态生成模块“$\mathcal{G}$”, 使其输出尽可能与“$\mathcal{S}$”模块的输出相接近, 本文引入$\mathcal{L}_{\mathrm{gene}}$ 损失函数, 并将其定义为

$$\mathcal{L}_{\mathrm{gene}}=\left\|\widetilde{x_i^*}-\widehat{x_i^*}\right\|_2^2+\left\|\widetilde{y_i^*}-\widehat{y_i^*}\right\|_2^2,\ *\in\{p,v,t\} \tag{15}$$

对于细粒度特征融合模块, 该模块旨在有效地融合多模态特征并映射成最终的哈希码. 为了提高学习的松弛哈希码 h_i^p 的表示能力, 本文充分利用类别标签信息, 引入分类损失函数$\mathcal{L}_{\mathrm{clf}}$, 并将其定义为

$$\mathcal{L}_{\mathrm{clf}}=\|\widetilde{l_i^p}-l_i^p\|_2^2 \tag{16}$$

在利用符号函数将松弛的哈希码转化为二进制代码时会引起量化误差, 为了最小化该量化误差, 本文引入量化损失函数$\mathcal{L}_{\mathrm{sign}}$, 并将其定义为

$$\mathcal{L}_{\mathrm{sign}}=\|h_i^p-b_i^p\|_2^2 \tag{17}$$

为了保持样本之间的成对相关性, 本文进一步引入$\mathcal{L}_{\mathrm{sim}}$ 损失函数:

$$\mathcal{L}_{\mathrm{sim}}=\|\cos(h_i^p,h_j^p)-S_{ij}^p\|_2^2,\ S_{ij}^p=\frac{2}{1+\mathrm{e}^{-l_i^p(l_j^p)^T}}-1 \tag{18}$$

其中, 矩阵 S 用于建模相关样本之间的细粒度关联.

因此, 细粒度特征融合模块的总损失函数为

$$\mathcal{L}=\alpha_1\mathcal{L}_{\mathrm{clf}}+\alpha_2\mathcal{L}_{\mathrm{sign}}+\alpha_3\mathcal{L}_{\mathrm{sim}} \tag{19}$$

其中, α_1、α_2、α_3 都是可调节的超参数.

2.6 多模态样本的在线查询

利用离线阶段训练的模型, 在线查询阶段将新来的多模态查询样本映射为二进制哈希码, 从而支持基于汉明距离的多模态数据的快速检索. 同时, 本文提出的方法也考虑到查询样本模态不完整的情况: 当新的多模态查询样本(或多个查询样本)到达时, 如果它缺失部分模态, 则首先利用训练后的跨模态生成模块“$\mathcal{G}$”生成缺失的模态特征; 然后, 对于模态完整的查询样本, 利用训练后的细粒度特征融合模块直接将其映射为对应的二进制哈希码, 以实现快速的多模态数据检索.

3 实验结果与分析

3.1 实验数据

本文在 MIR Flickr[34]和 MS COCO[35]这两个公开数据集上进行实验. 这两大数据集都包含了图像和文本模态的数据, 并在多模态数据检索任务中被广泛使用. 与先前的工作[16−19]类似, 预先使用预训练的 VGGNet[36]网络来提取 4 096 维的图像特征, 并分别用 1 386 维和 2 000 维的词袋(bag-of-words, BoW)向量作为

两个数据集的文本特征. 这两个数据集的统计信息见表 1. 下面对这两个数据集进行详细描述.

- MIR Flickr[34]数据集包含 25 000 个从 Flickr 网站上搜集的图像-文本对, 共有 24 个不同种类的类别. 从中挑选出 20 015 个图像-文本对, 并且每个图像-文本对至少属于 24 个类别中的 1 个类别. 遵循通常的实验设置[16,23,29], 在每个类别中随机选择 100 个样本, 去除重复样本后, 得到 2 243 个样本作为查询集, 将剩余的 17 772 个样本全部作为检索集, 并在检索集中随机挑选 5 000 个样本作为训练集.
- MS COCO[35]作为一个规模较大的数据集, 它一共包含了 80 个不同类别的图像-文本对. 遵循通常的实验设置[16,23,29], 在每个类别中随机挑选若干样本并去除重复后, 得到 5 981 个样本作为查询集, 将剩余的 82 783 个样本全部作为检索集, 并从中随机挑选 18 000 个样本作为训练集.

表 1 MIR Flickr 和 MS COCO 的统计信息

数据集	训练集样本个数	检索集样本个数	查询集样本个数	样本类别数量	图像模态的特征维度	文本模态的特征维度
MIR Flickr[34]	5 000	17 772	2 243	24	4 096 维	1 386 维
MS COCO[35]	18 000	82 783	5 981	80	4 096 维	2 000 维

3.2 实验设置

- 对于第 2.3.1 节中预先从完整模态的样本集合中随机挑选的锚点个数, N_a 取 300.
- 对于第 2.3.1 节中的同模态补齐模块, 本文采用基于自注意力机制的方法实现. 并且公式(4)中的 d_k 取 1 024.
- 对于第 2.3.2 节中的跨模态生成模块“$\mathcal{G}$”, 其包含 2 个全连接层, 每个全连接层都由一个线性映射层、一个批归一化层和一个 Tanh 激活函数组成, 并且隐藏层维度取 2 048.
- 对于第 2.4 节中的图像和文本模态编码的编码器, 即公式(9)中的 Encoder, 采用 2 个 Transformer 编码器模块[24]实现, 每个编码器具体由单头注意力层、归一化层和前馈网络层构成, 并且 d_{common} 取 128.
- 对于第 2.4 节中的逐位哈希映射函数, 即公式(11)中的 $H_k(\widetilde{c_{i_k}^{(f)}};\theta_{H_k})$, 其由 2 个线性层组成, 经过 128→64→1 的方式, 将 K 个细粒度融合特征映射成对应哈希位上的松弛值.

本文的实验代码使用 PyTorch 实现, 并使用单个 NVIDIA Tesla T4 进行训练. 在训练阶段, 批大小被设置为 256 和 512, 学习率根据经验被设置为 0.001, 并使用 Adam 优化器[37]通过标准的反向传播(back-propagation, BP)算法优化整个网络. 公式(19)中的可调节超参数在两个数据集上都取相同的值: α_1=1, α_2=0.01, α_3=1.

3.3 与现有方法的比较

3.3.1 完整多模态数据检索

遵循目前大多数多模态哈希的工作[16−19], 本文采用 mAP (mean average precision)作为标准评价指标, 定量评估所提出方法的检索效果. mAP 值越大, 表示模型的检索性能越好.

为验证本文所提出模型的优势, 本文与 8 种多模态哈希的基准模型进行比较, 具体包括 4 种基于浅层学习框架的方法: DMVH[14]、FOMH[20]、FDMH[22]、SAPMH[28]和 4 种基于深度学习框架的方法: DCMVH[23]、FGCMH[17]、BSTH[16]、NCH[29]. 这些方法都已经在第 1 节中进行了介绍. 为了公平起见, 除非额外标注, 其余实验结果均直接参考原始文献.

首先, 在所有样本都拥有完整模态的情况下进行实验. 分别考虑了哈希码取 16 位、32 位、64 位和 128 位的情况, 实验结果如图 4 和图 5 所示. 其中, NCH 在 MS COCO 数据集上的实验结果是参考源代码复现的. 由图可知: 随着哈希码位数的增加, 大部分方法的检索效果都得到了提高. 因为越多的位数使得哈希码能够表示的信息越多, 这与理论上的预期结果相一致. 同时, 我们发现一个有趣的现象: 在 MIR Flickr 数据集上, 当哈希码位数由 16 位扩大至 32 位时, 大部分方法的检索效果得到显著提升; 而继续扩大哈希码位数, 特别是由 64 位扩大至 128 位时, 其检索效果的提升幅度明显减小. 这是因为 MIR Flickr 数据集规模较小, 仅包含 24 个不同种类别, 因此蕴含的语义信息较少, 使用 32 位或者 64 位的哈希码就能拥有较好的表示能力. 而在规模

较大的 MS COCO 数据集上，这种现象便不明显，需要使用 64 位或者 128 位的哈希码才能保证较好的表示能力. 另一方面，随着哈希码位数的增加，训练阶段所花费的时间也明显增加. 例如：在相同的实验设置下，对于 MS COCO 数据集，当哈希码取 64 位时，模型的训练时间大约为取 32 位哈希码时的 2 倍；而当哈希码取 128 位时，模型的训练时间接近于取 32 位哈希码时的 3 倍. 通过考量检索效果以及相应的训练时长，在两个数据集上都选择取 64 位哈希码进行进一步的消融实验和超参数实验. 总体而言，相比于现有的最先进的基准模型，本文提出的方法在 MIR Flickr 和 MS COCO 数据集上都取得了最佳或次佳的效果. 具体而言，对于规模较小的 MIR Flickr 数据集，当哈希码位数大于等于 32 位时，我们的方法取得最优的检索效果；在规模较大的 MS COCO 数据集上，虽然我们的方法略逊于 BSTH，但是 BSTH 仅能处理样本具有完整模态的情况. 换言之，当样本缺失部分模态时，BSTH 在训练时不能利用不完整模态的样本，这将会大幅度降低其性能. 而 NCH 在 MS COCO 数据集上的表现稍逊于 BSTH 和我们的 PMH-F^3 方法. 分析原因主要是由于 NCH 没有着重关注深层语义信息的捕捉，从而在规模较大、具有较复杂语义信息的数据集上表现不佳.

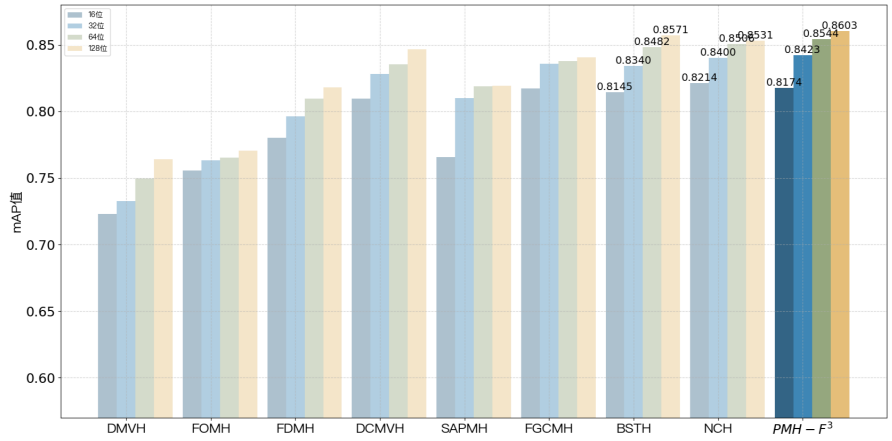


图 4　完整模态情况下 MIR Flickr 数据集上的 mAP 比较结果

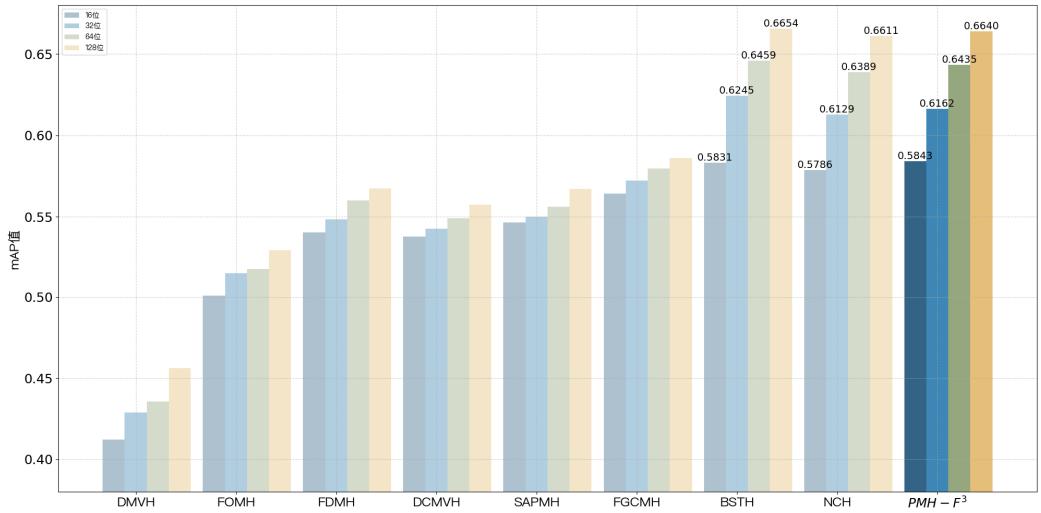


图 5　完整模态情况下 MS COCO 数据集上的 mAP 比较结果

3.3.2　部分多模态数据检索

考虑以下 3 种情况：训练集有样本缺失部分模态、查询集有样本缺失部分模态以及训练集和查询集都有样本缺失部分模态. 当训练集有样本缺失部分模态而查询集样本都有完整模态时，在查询阶段就不需要用

“$\mathcal{G}$”模块生成缺失模态, 比较的仅是不同方法下“$\mathcal{S}$”模块的性能, 因此将其放在消融实验部分. 由于目前针对部分多模态哈希的方法较少, 在完整多模态数据检索实验提到的 8 种方法中仅有 SAPMH 和 NCH 专门针对部分多模态数据检索的场景, 因此只着重对比这两种方法. 通过完整多模态数据检索的实验结果可知, 我们所提出模型的检索性能随着哈希码位数的增大而增强. 因此, 这里仅考虑哈希码位数都取 64 位的情况. 我们将不完整模态样本数占总样本数的比例称为 PDR (partial data ratio). 为了验证不完整模态样本的数量对检索性能的影响, 在不同 PDR 的情况下进行了充分的实验. 具体而言, 设置了 5 种不同的取值, 以 *PDR*=50%为例, 这表示训练集或查询集中有 50%的样本的模态是不完整的, 并且默认在不完整模态样本中缺失图像模态的样本和缺失文本模态的样本各占一半.

查询集有部分样本缺失模态的实验结果如表 2 所示(每种情况下最好结果用粗体标记, 第二好的结果用下划线标记), 其中, NCH 的实验结果是参考其源代码, 在相同的训练集和测试集划分的情况下复现的. 由表 2 可知: 随着 PDR 的增加, 所有方法的 mAP 值均下降. 与其他两个方法相比, 我们的方法总能取得最优的效果. 这说明虽然样本缺失部分模态会导致语义的损失, 从而影响检索的效果, 但是我们的方法可以有效地补齐缺失的模态, 以减轻性能下降幅度. 与 NCH 方法相比, 由于我们的方法在补齐缺失模态后采用更深层次的网络架构捕获多模态数据中的深层语义信息, 并且在细粒度特征层面融合多模态数据特征, 因此取得了更优的效果. 由于在规模较小的 MIR Flickr 数据集上 SAPMH 方法的效果总是明显不如 NCH 和 PMH-F^3 方法, 因此, 为了节约训练时间, 在规模较大的 MS COCO 数据集上, 只着重和 NCH 方法进行对比.

表 2　查询集有部分样本缺失模态的 mAP 比较结果

PDR (%)	MIR Flickr			MS COCO	
	SAPMH	NCH	PMH-F^3	NCH	PMH-F^3
10	0.812 2	0.845 8	**0.851 3**	0.623 5	**0.631 3**
30	0.803 2	0.836 4	**0.845 0**	0.616 4	**0.618 2**
50	0.793 6	0.827 2	**0.835 6**	0.607 3	**0.609 3**
70	0.783 1	0.818 7	**0.827 5**	0.588 3	**0.597 4**
90	0.772 0	0.811 4	**0.821 7**	0.575 2	**0.587 7**

训练集和查询集都有部分样本缺失模态的实验结果见表 3, 其中, NCH 的实验结果是参考其源代码,在相同的训练集和测试集划分的情况下复现的. 考虑到实际场景中训练集中缺失模态的样本所占的比例不会过高, 因此设置训练集 PDR 最高为 50%. 由表 3 可知, 我们的方法在大规模的 MS COCO 数据集上一直取得最优的效果. 对于规模较小的 MIR Flickr 数据集, 当训练集上的 PDR 大于 50%时, 我们的方法性能有所下降. 这是因为在训练细粒度特征融合模块时, 用的样本均为完整模态的样本, 当样本数量过少时会影响其性能. 但是实际场景中, 通常可以特意收集完整模态的样本作为训练集, 保证训练集上的 PDR 不会过大. 因此, 表 3 的结果是可接受的. 同样, 在规模较大的 MS COCO 数据集上, 只着重和 NCH 方法进行对比.

表 3　训练集和查询集都有部分样本缺失模态的 mAP 比较结果

PDR (%)	MIR Flickr			MS COCO	
	SAPMH	NCH	PMH-F^3	NCH	PMH-F^3
10	0.800 2	0.850 6	**0.856 2**	0.630 3	**0.638 0**
30	0.774 7	0.840 6	**0.843 7**	0.614 4	**0.618 1**
50	0.758 3	**0.827 1**	0.823 4	0.596 5	**0.602 3**
70	0.753 3	**0.817 3**	0.809 2	0.573 4	**0.578 2**
90	0.762 2	**0.803 2**	0.800 2	0.533 7	**0.542 7**

考量训练模型的时间代价, 虽然从直观上看, 我们的方法在补齐样本缺失的模态时需要额外的训练时间, 缺失模态的样本数越多, 需要的补齐时间也就越多. 但是从实验结果看, 随着 PDR 的增加, 模型整体的训练时长反而有所降低. 以 MS COCO 数据集为例, 在相同的实验设置下, 当训练集的 PDR 取 0.1 时, 训练时间为 19 min; 然而当训练集的 PDR 取 0.3 时, 训练时间缩短为 15 min; 当训练集的 PDR 取 0.5 时, 训练时间为 12 min. 这主要是由以下 3 个原因导致的.

- 首先, 为了减少补齐缺失模态时引入的噪声, 模型的“$\mathcal{S}$”模块是基于完整模态的样本进行预先训练

的, 它只用于监督训练“$\mathcal{G}$”模块生成缺失模态, 缺失模态的样本数越多, 训练“$\mathcal{S}$”模块所花费的时间越少.

- 其次, 为了减小噪声对于细粒度特征融合模块的影响, 该模块也是基于完整模态的样本进行训练的, 缺失模态的样本数越多, 训练该模块的时间也越少. 虽然这会减少“$\mathcal{S}$”模块和细粒度特征融合模块的训练样本的数量, 从而一定程度上影响模型的整体性能, 但是我们的方法仍然能够取得整体上最优的结果.
- 最后, 值得注意的是: 我们的模型分为离线训练阶段和在线查询阶段, 模型的训练完全是在离线阶段进行的, 即使需要较长的训练时间, 也是完全可接受的; 对于查询阶段到来的少量新样本, 只需要基于训练后的模型进行缺失模态补齐, 而用于补齐缺失模态的“$\mathcal{G}$”模块的构造简单, 其增加的时间代价几乎可以忽略不计.

综上所述, 可以认为, 我们的方法在训练的时间代价上是可接受的.

3.4 消融实验

因为本文所提出的方法主要用于样本模态不完整情况下的多模态哈希, 所以特意在严苛的环境下进行消融实验. 具体而言, 设置训练集 *PDR*=50%, 查询集 *PDR*=90%, 哈希码位数为 64 位. 设计了 5 种不同的变体: (1) 在实现“$\mathcal{S}$”模块时, 用 KNN 方法代替基于自注意力的方法, 将该变体记作 PMH-F^3-v1; (2) 移除“$\mathcal{S}$”模块并直接使用基于完整模态的样本集合训练后的“$\mathcal{G}$”模块进行缺失模态补齐, 将该变体记作 PMH-F^3-v2; (3) 将细粒度特征融合模块全部替换为浅层的 MLP 架构, 并将该变体记作 PMH-F^3-v3; (4) 将细粒度特征融合模块改为先进行特征融合再经过编码器编码, 并将该变体记作 PMH-F^3-v4; (5) 将细粒度特征融合模块中模态特定的 Transformer 编码器替换为模态共享的编码器, 即不同模态的编码器进行权值共享, 将该变体记作 PMH-F^3-v5. 消融实验的结果见表 4, 从中可以得出以下结论.

表 4 消融实验的 mAP 比较结果

方法	MIR Flickr	MS COCO
PMH-F^3-v1	0.788 1	0.522 7
PMH-F^3-v2	0.791 6	0.527 3
PMH-F^3-v3	0.762 3	0.498 1
PMH-F^3-v4	0.788 6	0.489 7
PMH-F^3-v5	0.789 0	0.521 9
PMH-F^3	**0.800 2**	**0.542 7**

- 对比 PMH-F^3-v1. 实验结果表明, 基于 KNN 方法实现的“$\mathcal{S}$”模块的性能不如基于自注意力的方法. 虽然 KNN 方法是无参数的, 这使得它具有运行时间短的优势, 但是它只是对锚点的特征进行简单的加权平均, 并不能像基于自注意力的方法一样, 自适应地对特征间的语义关系进行准确的建模.
- 对比 PMH-F^3-v2. 实验结果表明, 直接使用“$\mathcal{G}$”模块进行缺失模态补齐的效果不如在“$\mathcal{S}$”模块监督下的“$\mathcal{G}$”模块的补齐效果. 因为“$\mathcal{G}$”模块仅包含 2 个全连接层, 所以它对于同一样本的不同模态之间存在的相似语义关系的建模不够充分; 而“$\mathcal{S}$”模块以自注意力的方式可以准确地捕捉这类语义关系, 从而更好地引导“$\mathcal{G}$”模块. 此外, “$\mathcal{S}$”模块能够先补齐缺失的模态特征, 增加“$\mathcal{G}$”模块的训练样本的数目.
- 对比 PMH-F^3-v3. 实验结果表明, 简单地特征融合后 MLP 生成哈希码的效果不如细粒度特征融合模块. 因为基于 MLP 实现的变体仅能捕捉粗粒度的语义信息并在特征层面进行多模态融合, 而本文所提出的方法通过将特征分解为 K 个浅层语义信息, 并以自注意力的方式自适应地捕捉浅层语义信息间的关系, 从而得到深层语义信息, 实现了细粒度层面的多模态特征融合.
- 对比 PMH-F^3-v4. 实验结果表明, 先进行特征融合再经过编码器编码的设置会导致模型检索性能下降. 因为我们引入模态特定的编码器是为了以自注意力的方式对浅层语义信息间的关系进行建模, 从而自适应地捕捉深层语义信息. 而如果先进行特征融合, 则可能会破坏具体模态的深层语义信息.
- 对比 PMH-F^3-v5. 实验结果表明, 采用模态共享编码器会导致检索性能下降. 因为不同模态之间的差

异, 干扰了编码器对具体模态的深层语义信息的捕获.

3.5 超参数实验

第 2.5 节的公式(19)表明, 在细粒度特征融合模块中引入了 3 个超参数(α_1,α_2,α_3). 其中, α_1 是分类损失函数$\mathcal{L}_{\text{clf}}$的权重, 旨在通过充分利用标签信息提升哈希码的表示能力; α_2 和α_3 是分别是$\mathcal{L}_{\text{sign}}$ 和的$\mathcal{L}_{\text{sim}}$ 的权重, 旨在减小量化误差与保持样本间的相关性, 以此提升细粒度特征融合模块的性能. 为了探索 mAP 值受超参数变化的影响, 我们在 MS COCO 数据集上进行了超参数实验. 取训练集 *PDR*=50%, 查询集 *PDR*=90%. 设定超参数的可能取值范围为{0.0001,0.001,0.01,0.1,1,10,100}, 实验结果如图 6 所示. 总体上看, mAP 值随超参数的变化大致呈先增大后减小的趋势. 当α_1、α_2、α_3 分别取 1、0.01、1 时, mAP 值最大, 意味着此时模型的检索性能最佳. 如果某个超参数取值为 100, 此时总体损失函数近似于与该超参数相关的单个损失函数, 其余两个损失函数不能对模型进行优化, 因此mAP值迅速减小. 例如, α_1和α_3取100时的mAP值比α_2取100时的值大, 说明相较于$\mathcal{L}_{\text{sign}}$, $\mathcal{L}_{\text{clf}}$和$\mathcal{L}_{\text{sim}}$对模型优化起更大作用, 即: 在学习多模态数据的哈希码时, 有效地利用标签信息并保持样本间的成对相关性更加重要. 这也解释了为什么当α_1、α_2、α_3 分别取 1、0.01、1 时 mAP 值最大. 此外, 有一个有趣的现象: 当α_1 和α_2 取 100 时, 不同长度的哈希码对应的 mAP 值近似一样; 而α_3 取 100 时, mAP 值随着哈希码位数的增大而增大. 这可能是因为越长的哈希码对样本之间的成对相关性的建模能力越强.

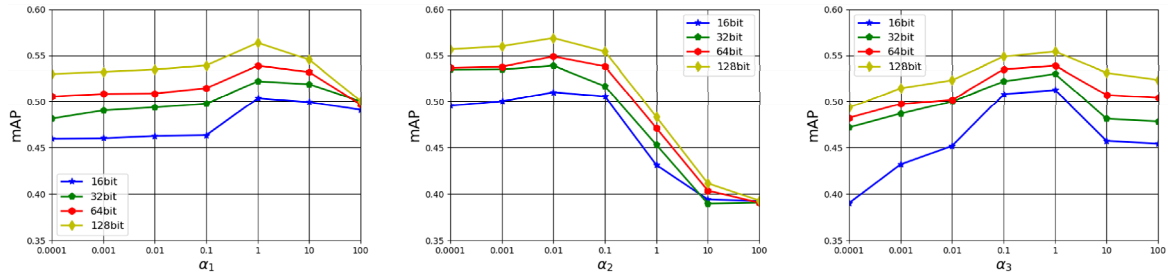


图 6 超参数实验的 mAP 比较结果

4 总 结

针对目前多模态哈希存在的模态缺失问题, 本文提出了 PMH-F^3 模型, 该模型的主要优点表现在以下两个方面: (1) 当样本缺失部分模态时, 该模型能够利用缺失模态补齐模块补齐所缺失的模态特征; (2) 提出的模型完全基于深度神经网络架构, 利用 Transformer 编码器, 以自注意力的方式捕获深层语义信息, 从而实现细粒度的多模态特征融合. 实验结果表明, 本文所提出的方法无论在完整多模态数据检索还是在部分多模态数据检索中都能取得最优的检索效果. 在未来的工作中, 我们将进一步探索如何在补齐缺失模态特征时减少引入的噪声, 以及其他解决模态缺失问题的方法, 例如借助标签辅助和利用知识蒸馏等. 此外, 我们将探索如何在实际的多模态数据检索场景应用所提出的方法.

References:

[1] Zhu D, Zhang BW, Cheng YQ, Liu XY, Wu WL, Wang TX, Wen H, Li BH. A survey of knowledge enabled new generation information systems. Ruan Jian Xue Bao/Journal of Software, 2023, 34(10): 4439−4462 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6884.htm [doi: 10.13328/j.cnki.jos.006884]

[2] Yang Y, Zhan DC, Jiang Y, Xiong H. Reliable multi-modal learning: A survey. Ruan Jian Xue Bao/Journal of Software, 2021, 32(4): 1067−1081 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/6167.htm [doi: 10.13328/j.cnki.jos.006167]

[3] Shen FM, Xu Y, Liu L, *et al*. Unsupervised deep hashing with similarity-adaptive and discrete optimization. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2018, 40(12): 3034−3044. [doi: 10.1109/TPAMI.2018.2789887]

[4] Song J, He T, Gao L, *et al*. Binary generative adversarial networks for image retrieval. In: Proc. of the 2018 AAAI Conf. on Artificial Intelligence. New Orleans: AAAI, 2018. 394−401. [doi: 10.1609/aaai.v32i1.11276]

[5] Zhang J, Peng Y. SSDH: Semi-supervised deep hashing for large scale image retrieval. IEEE Trans. on Circuits and Systems for Video Technology, 2019, 29(1): 212−225. [doi: 10.1109/TCSVT.2017.2771332]

[6] Bai C, Zeng C, Ma Q, *et al*. Deep adversarial discrete hashing for cross-modal retrieval. In: Proc. of the 2020 Int'l Conf. on Multimedia Retrieval. Dublin Ireland: ACM, 2020. 525−531. [doi: 10.1145/3372278.3390711]

[7] Meng M, Wang HT, Yu J, *et al*. Asymmetric supervised consistent and specific hashing for cross-modal retrieval. IEEE Trans. on Image Processing, 2020, 30: 986−1000. [doi: 10.1109/TIP.2020.3038365]

[8] Wang YX, Luo X, Xu XS. Label embedding online hashing for cross-modal retrieval. In: Proc. of the 28th ACM Int'l Conf. on Multimedia. Seattle: ACM, 2020. 871−879. [doi: 10.1145/3394171.3413971]

[9] Song JK, Yang Y, Huang Z, *et al*. Effective multiple feature hashing for large-scale near-duplicate video retrieval. IEEE Trans. on Multimedia, 2013, 15(8): 1997−2008. [doi: 10.1109/TMM.2013.2271746]

[10] Liu L, Yu MY, Shao L. Multiview alignment hashing for efficient image search. IEEE Trans. on Image Processing, 2015, 24(3): 956−966. [doi: 10.1109/TIP.2015.2390975]

[11] Shen XB, Shen FM, Sun QS, *et al*. Multi-view latent hashing for efficient multimedia search. In: Proc. of the 23rd ACM Int'l Conf. on Multimedia. Brisbane: ACM, 2015. 831−834. [doi: 10.1145/2733373.2806342]

[12] Shen XB, Shen FM, Liu L, *et al*. Multiview discrete hashing for scalable multimedia search. ACM Trans. on Intelligent Systems and Technology, 2018, 9(5): 1−21. [doi: 10.1145/3178119]

[13] Liu XL, He JF, Liu D, *et al*. Compact kernel hashing with multiple features. In: Proc. of the 20th ACM Int'l Conf. on Multimedia. Nara: ACM, 2012. 881−884. [doi: 10.1145/2393347.2396337]

[14] Yang R, Shi YL, Xu XS. Discrete multi-view hashing for effective image retrieval. In: Proc. of the 2017 Int'l Conf. on Multimedia Retrieval. Bucharest: ACM, 2017. 175−183. [doi: 10.1145/3078971.3078981]

[15] Zhu L, Zheng CQ, Guan WL, *et al*. Multi-modal hashing for efficient multimedia retrieval: A survey. IEEE Trans. on Knowledge and Data Engineering, 2024, 36(1): 239−260. [doi: 10.1109/TKDE.2023.3282921]

[16] Tan WT, Zhu L, Guan WL, *et al*. Bit-aware semantic transformer hashing for multi-modal retrieval. In: Proc. of the 45th Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. Madrid: ACM, 2022. 982−991. [doi: 10.1145/3477495.3531947]

[17] Lu X, Zhu L, Liu L, *et al*. Graph convolutional multi-modal hashing for flexible multimedia retrieval. In: Proc. of the 29th ACM Int'l Conf. on Multimedia. ACM, 2021. 1414−1422. [doi: 10.1145/3474085.3475598]

[18] Zheng CQ, Zhu L, Zhang Z, *et al*. Efficient semi-supervised multimodal hashing with importance differentiation regression. IEEE Trans. on Image Processing, 2022, 31: 5881−5892. [doi: 10.1109/TIP.2022.3203216]

[19] Sang JT, Gao Y, Bao BK, *et al*. Recent advances in social multimedia big data mining and applications. Multimedia Systems, 2016, 22(1): 1−3. [doi: 10.1007/s00530-015-0482-5]

[20] Lu X, Zhu L, Cheng ZY, *et al*. Flexible online multi-modal hashing for large-scale multimedia retrieval. In: Proc. of the 27th ACM Int'l Conf. on Multimedia. Nice: ACM, 2019. 1129−1137. [doi: 10.1145/3343031.3350999]

[21] Lu X, Zhu L, Li JJ, *et al*. Efficient supervised discrete multi-view hashing for large-scale multimedia search. IEEE Trans. on Multimedia, 2020, 22(8): 2048−2060. [doi: 10.1109/TMM.2019.2947358]

[22] Liu LY, Zhang Z, Huang Z. Flexible discrete multi-view hashing with collective latent feature learning. Neural Processing Letters, 2020, 52(3): 1765−1791. [doi: 10.1007/s11063-020-10221-y]

[23] Zhu L, Lu X, Cheng ZY, *et al*. Deep collaborative multi-view hashing for large-scale image search. IEEE Trans. on Image Processing, 2020, 29: 4643−4655. [doi: 10.1109/TIP.2020.2974065]

[24] Vaswani A, Shazeer N, Parmar N, *et al*, Polosukhin I. Attention is all you need. In: Proc. of the 31st Int'l Conf. on Neural Information Processing Systems. Long Beach: Curran Associates Inc., 2017. 6000−6010.

[25] Shen XB, Shen FM, Sun QS, *et al*. Semi-paired discrete hashing: Learning latent Hash codes for semi-paired cross-view retrieval. IEEE Trans. on Cybernetics, 2017, 47(12): 4275−4288. [doi: 10.1109/TCYB.2016.2606441]

[26] Liu H, Lin MB, Zhang SC, *et al*. Dense auto-encoder hashing for robust cross-modality retrieval. In: Proc. of the 26th ACM Int'l Conf. on Multimedia. Seoul: ACM, 2018. 1589−1597. [doi: 10.1145/3240508.3240684]

[27] Guo J, Zhu WW. Collective affinity learning for partial cross-modal hashing. IEEE Trans. on Image Processing, 2020, 29: 1344−1355. [doi: 10.1109/TIP.2019.2941858]

[28] Zheng CQ, Zhu L, Cheng ZY, *et al*. Adaptive partial multi-view hashing for efficient social image retrieval. IEEE Trans. on Multimedia, 2021, 23: 4079−4092. [doi: 10.1109/TMM.2020.3037456]

[29] Tan WT, Zhu L, Li JJ, *et al*. Partial multi-modal hashing via neighbor-aware completion learning. IEEE Trans. on Multimedia, 2023, 25: 8499−8510. [doi: 10.1109/TMM.2023.3238308]

[30] Wang QQ, Ding ZM, Tao ZQ, *et al*. Partial multi-view clustering via consistent GAN. In: Proc. of the 2018 IEEE Int'l Conf. on Data Mining. Singapore: IEEE, 2018. 1290−1295. [doi: 10.1109/ICDM.2018.00174]

[31] Guo J, Ye JH. Anchors bring ease: An embarrassingly simple approach to partial multi-view clustering. In: Proc. of the 2019 AAAI Conf. on Artificial Intelligence. Hawaii: AAAI, 2019. 118−125. [doi: 10.1609/aaai.v33i01.3301118]

[32] Zeng JD, Liu TY, Zhou JT. Tag-assisted multimodal sentiment analysis under uncertain missing modalities. In: Proc. of the 45th Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. Madrid: ACM, 2022. 1545−1554. [doi: 10.1145/3477495.3532064]

[33] Ma MM, Ren J, Zhao L, *et al*. SMIL: Multimodal learning with severely missing modality. In: Proc. of the 2021 AAAI Conf. on Artificial Intelligence. 2021, 35(3): 2302−2310. [doi: 10.1609/aaai.v35i3.16330]

[34] Huiskes MJ, Thomee B, Lew MS. New trends and ideas in visual concept detection: The MIR flickr retrieval evaluation initiative. In: Proc. of the Int'l Conf. on Multimedia Information Retrieval. Philadelphia: ACM, 2010. 527−536. [doi: 10.1145/1743384.1743475]

[35] Lin TY, Maire M, Belongie S, *et al*. Microsoft COCO: Common objects in context. In: Proc. of the European Conf. on Computer Vision. Zurich: Springer, 2014. 740−755. [doi: 10.1007/978-3-319-10602-1_48]

[36] Simonyan K, Zisserman A. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556, 2014.

[37] Kingma DP, Ba J. Adam: A method for stochastic optimization. arXiv:1412.6980, 2014.

附中文参考文献:

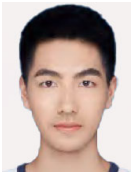
[1] 朱迪, 张博闻, 程雅琪, 刘昕悦, 吴文隆, 王铁鑫, 文浩, 李博涵. 知识赋能的新一代信息系统研究现状、发展与挑战. 软件学报, 2023, 34(10): 4439−4462. http:// www.jos.org.cn/1000-9825/6884.htm [doi: 10.13328/j.cnki.jos.006884]

[2] 杨杨, 詹德川, 姜远, 熊辉. 可靠多模态学习综述. 软件学报, 2021, 32(4): 1067−1081. http://www.jos.org.cn/1000-9825/6167.htm [doi: 10.13328/j.cnki.jos.006167]

殷崭祚(2000－), 男, 硕士生, CCF 学生会员, 主要研究领域为机器学习, 多模态学习.

黄瑞龙(2000－), 男, 硕士生, CCF 学生会员, 主要研究领域为机器学习, 语言模型, 自然语言处理.

李博涵(1979－), 男, 博士, 副教授, CCF 高级会员, 主要研究领域为时空数据库, 知识图谱, 大模型, 推荐系统.



吴文隆(2001－), 男, 本科生, CCF 学生会员, 主要研究领域为推荐系统, 机器学习.

王萌(1989－), 男, 博士, 副教授, CCF 专业会员, 主要研究领域为知识图谱, 多模态知识发现.

王昊奋(1982－), 男, 博士, 研究员, CCF 高级会员, 主要研究领域为知识图谱, 自然语言处理, 数据挖掘.