# 基于哈希算法的异构多模态数据检索研究

陈 凤 蒙祖强

(广西大学计算机与电子信息学院 南宁 530000)

**摘 要** 随着大数据时代的发展,网络上的文本、图像、视频、音频等异构多模态数据呈指数级增长。在海量数据中进行异构多模态数据的检索,成为了热门的研究方向。但是,异构多模态数据检索面临两大挑战:1)数据存在“语义鸿沟”,即如何表达异构多模态数据之间的相似性;2)在海量数据中,如何进行准确高效的检索。针对哈希检索算法忽略了异构多模态数据之间语义一致性的问题,文中提出了一种基于 CCA(典型相关性分析)语义一致性的哈希检索算法(CCA-SCH)。该算法为了保持模态内的语义一致性,分别生成文本和图像数据的语义模型;为了保持模态间的语义一致性,通过 CCA 算法融合文本和图像语义,生成最大相关矩阵;同时引入 $\ell_{2,\rho}$ 范式来减少原始数据集的噪声和冗余信息,使哈希函数具有更好的鲁棒性。实验结果表明,CCA-SCH 算法在实验数据集上的均值平均准确率(Map)相较于基准算法提升了 10% 以上,体现了该算法更好的检索性能。

**关键词** 哈希函数,语义一致性,CCA 算法,异构多模态

**中图法分类号** TP391 **文献标识码** A **DOI** 10.11896/jsjkx.190100139

## Study on Heterogeneous Multimodal Data Retrieval Based on Hash Algorithm

CHEN Feng MENG Zu-qiang

(College of Computer and Electronics Information, Guangxi University, Nanning 530000, China)

**Abstract** The development of the era of big data has resulted in an exponentially growing of Internet heterogeneous multimodal data including text, images, video and audio. Therefore, heterogeneous multimodal data retrieval has become a hot direction in big data research. However, heterogeneous multimodal data retrieval encounters two major challenges. The first challenge is how to express the similarity between heterogeneous data while there is a “semantic gap”. The second challenge is how to achieve accurate and efficient retrieval in massive data. To solve the problem that the hash retrieval algorithm ignores semantic similarity of heterogeneous multimodal data, this paper proposed a hash retrieval algorithm based on canonical correlation analysis-semantic consistency, named CCA-SCH. In order to keep semantic consistency within the modality, the CCA-SCH algorithm separately generates semantic models of text and image data. In order to keep semantic consistency between modalities, the CCA algorithm is used to fuse semantics of text and image data to generate the maximum correlation matrix. At the same time, the paradigm $\ell_{2,\rho}$ is introduced to overcome the noise and redundant information of original datasets, so that the hash function has better robustness. Experiment results show that the mean average precision(Map) of CCA-SCH algorithm is increased by over 10% compared to benchmark algorithms' performances on experimental data sets, which embodies the better retrieval ability of proposed algorithm.

**Keywords** Hash function, Semantic consistency, Canonical correlation analysis algorithm, Heterogeneous multimodal

## 1 引言

现今,来自社交网络和物理空间的各种平台数据以不同模态混合在一起,展示了丰富的自然属性和社会属性。它们代表了全面的知识,反映了个人和群体的行为。因此,人们认识到一种新形式的数据,称其为异构多模态数据[1]。针对同一事件或场景描述,异构多模态数据往往具有低层特征结构差异大但高层语义却相似的特性,即低层特征与高层语义表达之间存在“语义鸿沟”:高层语义表达相似的数据时,其低层特征可能存在很大的区别,甚至当异构多模态数据的高层语义完全相同时,其低层特征根本没有可比性。“语义鸿沟”给异构多模态数据挖掘与分析带来了很大的影响。大数据时代,数据不仅在规模上呈指数级增长,而且其结构和类型也日趋复杂,数据的多模态化转变是数据发展的一种重要趋势。传统的基于单模态数据检索的学习方法难以被用来处理异构多模态数据。与单模态数据检索不同,异构多模态数据检索涉及了更多的问题。首先,如何将多模态数据生成统一的表达模式,实现跨越“语义鸿沟”的统一语义表达;其次,如何构

到稿日期:2019-01-17 返修日期:2019-03-29 本文受国家自然科学基金项目(61762009)资助。

陈 凤 女,硕士生,主要研究方向为多模态数据处理,E-mail:hhzcarl@126.com;蒙祖强 男,教授,主要研究方向为数据挖掘与知识发现、智能信息处理,E-mail:zqmeng@126.com(通信作者)。

造有效的检索算法，以达到准确、快速的检索效果。

单模态数据检索指数据集中只包含一种模态的数据检索，如用文本检索文本、用图像检索图像。针对大规模的图像检索，近几年已有许多相关研究，例如，毛晓蛟等[2]提出的基于子空间学习的哈希检索算法，加快了检索的速度；曹玉东等[3]通过改进 LSH 算法提出的图像检索算法和张梁[4]提出的基于 LSH 的近似近邻图像检索算法，在保证未降低图像检索性能的前提下，降低了内存的使用量；Liu 等[5]提出的深度监督哈希算法和文庆福等[6]提出的面向近似近邻查询的分布式哈希学习算法，进一步提高了检索速度。

在单模态检索的基础上，有的学者将单模态检索方法扩展到异构多模态数据检索上，主要使用哈希检索方法。多模态哈希检索主要是通过不同模态的语义相似性来生成哈希编码，然后用哈希编码来表达语义相似性。Tang 等[7]提出了基于监督的矩阵分解的跨模态哈希(Supervised Matrix Factorization Hashing，SMFH)算法，该算法考虑了图文数据对的标签信息，而且可以扩展到大规模数据集，但是其忽略了跨模态数据之间的语义一致性。语义一致性[8]指将异构多模态数据投影到低维的共享语义子空间中，得到多模态数据共享的语义特征，进而挖掘出异构多模态数据的有用信息。Wang 等[9]提出了语义增强跨模态哈希(Semantic Boosting Cross-Modal Hashing，SBCMH)算法，该算法通过将异构多模态数据特征映射到语义子空间中，考虑了跨模态数据之间的语义一致性，但忽略了同种模态数据的语义一致性。针对某些方法忽略了哈希学习过程中的判别属性导致不同类的哈希编码不可区分的问题，Wang 等[10]提出了多模态判别二元嵌入(Multimodal Discriminative Binary Embedding，MDBE)算法，其重点是学习判别式哈希函数，该算法可以保留哈希编码的可区别性和相似性，提高了多模态检索的准确性。Li 等[11]提出了一种基于核的潜在语义稀疏哈希(Kernel based Latent Semantic Sparse Hashing，KLSSH)方法，其可以生成非常短且有区别的哈希编码，减小了量化损失，使检索性能更好。

哈希算法不仅可以节省存储空间，还可以达到精确、快速的检索目的，因此本文主要研究基于哈希算法的异构多模态数据检索。针对异构多模态数据检索往往对模态间的相似性或者模态内的相似性有所忽略的问题，本文提出了一种基于 CCA 语义一致性的哈希检索算法，目的在于保持模态内和各模态间的语义一致性，提高了异构多模态数据检索的精度。

## 2 基于 $p$ 的局部敏感哈希算法

基于 $p$ 稳定局部敏感哈希算法是由 Datar 等[12]于 2004 年提出的，用于解决距离度量的问题。该算法把数据集中的数据点 $v$ 随机投影到某个方向矢量 $\boldsymbol{a}_i$ 上，$\boldsymbol{a}_i$ 指正态分布随机产生的向量。$\boldsymbol{a}_i$ 的所有数据点服从 $p$ 稳定分布的规则，即当 $p\in(0,2]$ 时，$\boldsymbol{a}_i$ 的所有数据点是稳定分布的。更具体地，当 $p=1$ 时，称作柯西分布；当 $p=2$ 时，称作标准高斯分布。基于 $p$ 稳定局部敏感哈希算法具有如下性质：假设两个变量是 $p$ 稳定分布，则这两个向量的线性组合也是 $p$ 稳定分布。基于 $p$ 稳定哈希函数$\{h:R^d\rightarrow Z\}$的定义如式(1)所示：

$$h(v)=\left\lfloor\frac{a\cdot v+b}{w}\right\rfloor \tag{1}$$

其中，符号$\lfloor\ \rfloor$表示对所得的结果向下取整；$a=\{\boldsymbol{a}_i\}$，$i=\{1,2,\cdots,n\}$，内积 $a\cdot v$ 表示数据点 $v$ 在向量 $\boldsymbol{a}_i$ 上的投影；参数 $w$ 表示投影窗口的量化宽度；参数 $b$ 是为了消除哈希桶边界带来的影响而赋予哈希函数随机性，其取值遵循$[0,w]$的均匀分布。比如：在标准高斯分布中，数据点 $v_1$ 和 $v_2$ 的映射距离为 $h(v)=a\cdot v_1-a\cdot v_2$。

对于数据点 $v_1$ 和 $v_2$，有：

$$\begin{aligned}p(c)&=\Pr(h_{a,b}(v_1)=h_{a,b}(v_2))\\&=\int_0^w\frac{1}{c}f_p\left(\frac{x}{c}\right)\left(1-\frac{x}{r}\right)\mathrm{d}x\end{aligned} \tag{2}$$

其中，参数 $c$ 服从 $\|v_1-v_2\|_p$ 分布，函数 $f_p(x)$ 是 $p$ 稳定分布绝对值的概率密度函数。

对于降低原始数据的维度，若降到 $k$ 维，则需要 $k$ 个哈希函数。$k$ 个哈希函数的形式如式(3)所示：

$$f_i(v)=\{h_{j,1}(v),\cdots,h_{j,k}(v)\},j=\{1,\cdots,l\} \tag{3}$$

数据点 $v$ 的哈希码是由 $k$ 个整数构成的，基于 $p$ 稳定局部敏感哈希族的表达形式为：$\Gamma=f:R^d\rightarrow Z^d$。

## 3 CCA 语义一致性哈希检索算法

(1)文本数据语义模型

首先通过 TF-IDF 算法获得文本特征向量，通过 BTM 模型对文本数据集进行训练，获得文档-主题分布和主题-词对分布；然后通过哈希函数进行学习，用哈希码来表示数据的主题分布。针对文本数据的哈希函数学习，当该文本包含某一主题时，相应的哈希码为 1，否则为 0。假设 $\boldsymbol{X}$ 是通过 TF-IDF 算法获得的 $d_1$ 维文本特征向量集合，$\boldsymbol{H}^1$ 是文本数据的哈希编码，$\boldsymbol{T}$ 是通过 BTM 模型获得的 $d_1$ 维的文本主题集合。其中，$\boldsymbol{X}=[\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n]\in R^{d_1}$，$\boldsymbol{H}^1=[\boldsymbol{h}_1^{(1)},\cdots,\boldsymbol{h}_n^{(1)}]\in R^{d_1\times n}$，$\boldsymbol{T}=[\boldsymbol{t}_1,\cdots,\boldsymbol{t}_n]\in R^{d_1\times n}$。参考文献[13]的文本数据语义建模方法，其表达式如式(4)所示：

$$\begin{aligned}&\min_{\boldsymbol{T},\boldsymbol{H}^1}\sum_{i=1}^{n}\sum_{j=1}^{k}h_{ij}^{(x)}\|\boldsymbol{x}_i-\boldsymbol{t}_i\|_2^2\\&\text{s.t. }h_{ij}^{(x)}\in\{0,1\},\sum_{j=1}^{k}h_{ij}^{(x)}=c\end{aligned} \tag{4}$$

其中，$\|*\|_2^2$ 表示 $\ell_2$ 正则项；$h_{ij}^{(1)}$ 是哈希编码 $\boldsymbol{h}_i^{(1)}$ 的第 $j$ 个元素；$k$ 是哈希编码的长度；$c$ 是小于或等于 $k$ 的正整数，表示 $h_i^{(x)}$ 中有多少个“1”。

式(4)利用文本特征向量和文本主题向量的正则项方法来学习文本数据语义模型，要求哈希码是 0 或者 1，并且哈希码中为“1”的个数等于 $c$；求其最小值是为了得到性能更好的文本数据语义模型，以保持文本数据的语义一致性。

语义模型指利用文本数据的相关性，将文本数据投影到一个低维的共享子空间，消除文本数据的底层特征异构性，从而获得文本数据的高层语义信息。本文将文本数据中具有相同主题的不同特征向量映射到同一个哈希桶中，并且保证文本数据中具有不同主题的特征向量在不同的哈希桶中，同时采用基于 $p$ 稳定局部敏感哈希算法的哈希函数生成哈希编码，并生成文本数据的语义模型，从而获得文本数据之间的高层语义信息。然而，文本主题集合 $\boldsymbol{T}$ 中的 $\boldsymbol{t}_i$ 和 $\boldsymbol{t}_j$ 分布相似会

造成文本数据主题冗余,使得随机分配相应的哈希编码时不同语义的文本数据不能通过哈希编码区分。为了解决这个问题,增加一个关于主题多样性的正则化项。如果哈希编码 $\boldsymbol{h}_i^{(1)}$ 与 $\boldsymbol{h}_j^{(1)}$ ($i,j\in\{1,\cdots,k\}$) 在语义子空间很接近,则 $\boldsymbol{h}_i^{(x)}(\boldsymbol{h}_i^{(x)})^{\mathrm{T}}$ 的值很大,而且 $\boldsymbol{t}_i$ 和 $\boldsymbol{t}_j$ 有很大可能是相似的,会造成主题冗余。此语义子空间指利用文本数据的相关性将文本数据投影到一个低维的共享子空间,消除不同模态数据的特征异构性,从而获得不同模态数据的高层语义信息。因此,关于主题多样性的正则化可以避免主题 $\boldsymbol{t}_i$ 与 $\boldsymbol{t}_j$ 相似。关于主题多样性的正则化表达式[13]如式(5)所示:

$$\begin{aligned}&F_r(\boldsymbol{TWT}^{\mathrm{T}})=\min_{T}\sum_{i=1,j=1}^{k}w_{ij}(\boldsymbol{t}_i)^{\mathrm{T}}\boldsymbol{t}_j\\&\text{s. t } w_{ij}=\begin{cases}\boldsymbol{h}_i^{(x)}(\boldsymbol{h}_i^{(x)})^{\mathrm{T}}, & i\neq j\\0, & i=j\end{cases}\end{aligned}\tag{5}$$

其中,$w_{ij}$ 是两个哈希编码 $\boldsymbol{h}_i^{(1)}$ 与 $\boldsymbol{h}_j^{(1)}$ 的向量相似性度量;$(\boldsymbol{t}_i)^{\mathrm{T}}\boldsymbol{t}_j$ 是两个文本主题 $\boldsymbol{t}_i$ 和 $\boldsymbol{t}_j$ 的向量相似性度量。当 $w_{ij}$ 的值相对较大时,主题 $\boldsymbol{t}_i$ 与 $\boldsymbol{t}_j$ 的分布相似。当式(5)满足约束条件时其最小值表示 $(\boldsymbol{t}_i)^{\mathrm{T}}\boldsymbol{t}_j$ 是最小的,即主题 $\boldsymbol{t}_i$ 和 $\boldsymbol{t}_j$ 的分布彼此不相同,可以避免主题冗余,生成有区分力的哈希编码。

文本数据语义模型的目标函数可以通过组合式(4)和式(5)获得:

$$\begin{aligned}&\min_{T,H}F_D=\sum_{i=1}^{n}\sum_{j=1}^{k}h_{ij}^{(x)}\parallel\boldsymbol{x}_i-\boldsymbol{t}_i\parallel_2^2+F_r(\boldsymbol{TWT}^{\mathrm{T}})\\&\text{s. t. } h_{ij}^{(x)}\in\{0,1\},\sum_{j=1}^{k}h_{ij}^{(x)}=c\end{aligned}\tag{6}$$

式(6)通过利用文本数据的相关性,来消除底层特征异构性,将高层语义相同的数据映射到同一个哈希桶,并保证高层语义不同的数据映射到不同的哈希桶,这不仅保持了文本数据之间的语义一致性,还利用主题多样性的正则化方法避免了主题冗余,生成了有区分力的哈希码。

(2)图像数据语义模型

与文本数据相比,隐藏在图像中的高级语义信息更难获得。文献[14]通过协同矩阵分解来发现图像中的语义信息。协同矩阵分解的思想是同时对多个矩阵进行分解,并在分解的过程中挖掘存在于不同矩阵中的共享潜在语义信息。文献[14]中的方法通过共享潜在语义信息更好地保持了图像数据间的语义一致性。协同矩阵分解通常利用最小二乘法的思想最小化平方损失函数(Quadratic Loss Function)。最小二乘法的基本思想是:找到一条直线,使得各数据点到直线的距离平方和最小。假设样本个数为 $n$,某个点 $x$ 的机器模型输出结果为 $y$,其实际输出结果为 $f(x)$,则平方损失函数的表达式如式(7)所示:

$$L(y,f(x))=\sum_{i=1}^{n}(y-f(x))^2\tag{7}$$

互联网产生的大规模异构多模态数据集包含了大量的噪声和异常值,而平方损失函数对噪声数据和异常数据很敏感。因此,协同矩阵分解引入正则项来生成具有鲁棒性的矩阵。与 $\ell_2$ 正则项相比,$\ell_{2,p}$ ($0<p\leqslant 2$) 正则项没有对样本重建误差,因此在防止过拟合时对噪声和异常值具有鲁棒性。假设 $\boldsymbol{Y}=[\boldsymbol{y}_1,\cdots,\boldsymbol{y}_n]\in R^{d_2\times n}$ 是通过 SIFT 算法提取的 $d_2$ 维的图像特征向量集合。协同矩阵分解的过程是给定一个矩阵,分解出两个潜在信息矩阵,其表示形式如(8)所示:

$$\min_{\boldsymbol{U}_i,\boldsymbol{V}}\parallel\boldsymbol{Y}_i-\boldsymbol{U}_i\boldsymbol{V}\parallel_{2,p}=\sum_{i=1}^{n}\parallel\boldsymbol{Y}_i-\boldsymbol{U}_i\boldsymbol{V}\parallel_F^2\tag{8}$$

其中,$\parallel*\parallel_F^2$ 表示 Frobenius 范数;$n$ 表示有 $n$ 个矩阵;$\boldsymbol{U}_i$ 表示包含第 $i$ 个矩阵的潜在信息的矩阵,$\boldsymbol{U}_i=[u_1,\cdots,u_n]\in R^{d_1\times n}$;$\boldsymbol{V}$ 是共享潜在信息的矩阵,$\boldsymbol{V}=[v_1,\cdots,v_n]\in R^{k\times n}$;$k$ 是哈希码的长度。协同矩阵分解不仅可以让不同的矩阵在分解的过程中共享潜在信息的矩阵 $\boldsymbol{V}$,还可以学习在分解过程中矩阵特性的一些相似表达 $\boldsymbol{U}_i$。式(8)取最小值是为了得到性能更好的图像数据语义模型,并保持图像数据的语义一致性。

通常,嵌入 $\ell_{2,p}$ 正则项的损失函数比嵌入 $\ell_2$ 正则项的损失函数更难生成,因此将式(8)重写为:

$$\min_{\boldsymbol{U}_i,\boldsymbol{V}}\parallel\boldsymbol{Y}_i-\boldsymbol{U}_i\boldsymbol{V}\parallel_{2,p}=F_r\{(\boldsymbol{Y}_i-\boldsymbol{U}_i\boldsymbol{V})\boldsymbol{D}^*(\boldsymbol{Y}_i-\boldsymbol{U}_i\boldsymbol{V})^{\mathrm{T}}\}\tag{9}$$

其中,$\boldsymbol{D}^*$ 表示一个对角矩阵,只有对角元素是非零的,其第 $i$ 个对角线元素的计算式如式(10)所示:

$$(\boldsymbol{D}^*)_{ii}=\frac{1}{\parallel\boldsymbol{Y}_i-\boldsymbol{U}_i\boldsymbol{V}\parallel_2}\tag{10}$$

综上所述,协同矩阵分解主要是通过对子空间中的特征向量进行矩阵分解,消除图像数据的底层特征异构性,从而获取图像数据之间的潜在语义共享信息,保持图像数据内的语义一致性。图像语义建模的目标函数如式(11)所示:

$$\begin{aligned}&\min_{\boldsymbol{U}_i,\boldsymbol{V}}F_I=F_r\{\boldsymbol{Y}_i-\boldsymbol{U}_i\boldsymbol{V})\boldsymbol{D}^*(\boldsymbol{Y}_i-\boldsymbol{U}_i\boldsymbol{V})^{\mathrm{T}}\}\\&\text{s. t. } \boldsymbol{U}_i\geqslant 0,\boldsymbol{V}\geqslant 0\end{aligned}\tag{11}$$

(3)图像-文本语义融合

在异构多模态数据检索过程中,Hotelling[15]于 1936 年提出了典型相关分析(Canonical Correlation Analysis,CCA)方法。CCA 是利用综合变量的相关关系来反映两组变量之间的整体相关性的多元统计分析方法,其基本思想是:首先分别在两组变量中获取有代表性的两个综合变量 $A$ 和 $B$($A$ 和 $B$ 分别是两组变量中各变量的线性组合),然后通过这两个综合变量之间的相关关系来反映两组变量之间的整体相关性。对于文本数据 $\boldsymbol{X}=[x_1,\cdots,x_n]\in R^{d_1}$ 和图像数据 $\boldsymbol{Y}=[y_1,\cdots,y_n]\in R^{d_2\times n}$,假设两个变量的第 $i$ 对组合为 $A_i=\boldsymbol{X}^{\mathrm{T}}\boldsymbol{\alpha}_i$ 和 $B_i=\boldsymbol{Y}^{\mathrm{T}}\boldsymbol{\beta}_i$,其典型变量分别是 $\boldsymbol{\alpha}_i=(\alpha_{1i},\cdots,\alpha_{ni})^{\mathrm{T}}$ 和 $\boldsymbol{\beta}_i=(\beta_{1i},\cdots,\beta_{ni})^{\mathrm{T}}$。CCA 的表达式如式(12)所示:

$$\begin{aligned}&\max_{\boldsymbol{\alpha}_i,\boldsymbol{\beta}_i}\rho(M_i,N_i)=\boldsymbol{\alpha}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{12}\boldsymbol{\beta}_i\\&\text{s. t. } \mathrm{Res}(M_i)=\boldsymbol{\alpha}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{11}\boldsymbol{\alpha}_i=1,\\&\qquad\mathrm{Res}(N_i)=\boldsymbol{\beta}_i^{\mathrm{T}}\boldsymbol{\Sigma}_{11}\boldsymbol{\beta}_i=1\end{aligned}\tag{12}$$

其中,$\boldsymbol{\Sigma}_{11}$ 是第一组变量 $\boldsymbol{X}$ 的协方差矩阵,$\boldsymbol{\Sigma}_{22}$ 是第二组变量 $\boldsymbol{Y}$ 的协方差矩阵,$\boldsymbol{\Sigma}_{12}$ 是 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的协方差矩阵。式(12)取最大值是为了获得图像数据和文本数据的高层语义的最大相关性。协方差矩阵是一个矩阵,其中每个元素是各个向量元素之间的协方差。例如,对于 $\boldsymbol{X}=[\boldsymbol{x}_1,\cdots,\boldsymbol{x}_n]\in R^{d_1\times n}$,其表达式如式(13)所示:

$$\boldsymbol{\Sigma}_{ij}=E[(\boldsymbol{x}_i-E[\boldsymbol{x}_i])(\boldsymbol{x}_i-E[\boldsymbol{x}_j])^{\mathrm{T}}]\tag{13}$$

其中,$E[\boldsymbol{x}_i]$表示第 $i$ 个元素的期望值。

语义建模是将具有相似语义信息的异构多模态数据映射

到某个共同的低维潜在空间。图像数据和文本数据的语义是密切相关的，其表达式如式(14)所示：

$$\min_{\boldsymbol{P}} F_C = \| \boldsymbol{H} - \boldsymbol{P} \|_F^2 \tag{14}$$

其中，$P_i = \max_{\alpha_i,\beta_i} \rho(M_i, N_i) \in R^{k\times k}$。

$P$ 是文本数据和图像数据之间的相关矩阵。式(14)使得文本数据和图像数据对具有相同的哈希码，保持了多模态之间的语义一致性，用于异构多模态数据的检索。

CCA-SCH 算法的目标函数是结合式(6)表达的文本语义建模、式(11)表达的图像语义建模和式(14)表达的图像与文本语义相关性而来，如式(15)所示：

$$\min_{\boldsymbol{T},\boldsymbol{H},\boldsymbol{U},\boldsymbol{V},\boldsymbol{P}} F = \lambda F_D + (1-\lambda) F_I + \mu F_C + \gamma R(\boldsymbol{F},\boldsymbol{U},\boldsymbol{V},\boldsymbol{P})$$
$$\text{s.t. } h_{ij} \in \{0,1\}, \sum_{i=1}^{k} h_{ij} = c \tag{15}$$

其中，$\lambda$，$\mu$，$\gamma$ 是权衡参数，$R(*) = \| * \|_F^2$ 是避免过度拟合的正则化术语。

由式(15)可知，CCA-SCH 算法不仅将图像数据和文本数据映射到低维的共享语义子空间中，还保持了模态内和模态间的语义一致性。

CCA-SCH 算法的流程如算法 1 所示。

**算法 1** 基于 CCA 语义一致性的哈希检索算法

输入：文本特征向量 x；图像特征向量 y；参数 λ，μ 和 γ；哈希码长度 k

输出：具有语义一致性的文本数据或者图像数据

```
1. for i=1,2,…,n
2.    根据式(6)生成文本数据语义模型；
3.    根据式(11)生成图像数据语义模型；
4.    根据式(13)融合图像与文本数据的高层语义；
5. end for
6. 初始化每一个哈希函数，令其个数 count(h)=0
7. for i=1,2,…,n do
8.    for j=1,2,…,n do
9.       if h_j(x_i)=h_j(y_i) then
10.          count(h_i)=count(h_j)+1
11.      end if
12.   end for
13. end for
14. 按升序排列 count(h_j)，得到 l 个哈希表 T_1,T_2,…,T_l
15. 输入查询样例 x_e 或者查询样例 y_e，假设 p 点
16. for j=1,2,…,l do
17.   if p∈T_j中的第 m 个桶，在该桶中查找最近邻数据 q
18.      if 数据 p 和 q 满足查询条件
19.         则数据 q 为相似数据
20.      end if
21.   end if
22. end for
23. stop
```

## 4 实验结果与分析

为了验证 CCA-SCH 算法的有效性，基于 Wiki 数据集和 NUS-WIDE 数据集，将其与 SCM 算法[16]、SePH 算法[17]进行对比分析。Wiki 数据集是从 Wikipedia 数据集中抽取形成的 2866 个文本-图像对。NUS-WIDE 数据集是一个网络图像数据库，共有 81 组不同的真实样本，包含 269648 个标签-图像特征对，有 5018 个独一无二的标签。

### 4.1 度量标准

(1)均值平均准确率(Map)

我们使用均值平均准确率(Map)作为全局性能指标。Map 是查询的平均精确度(AP)的平均值，其表达式如式(16)所示：

$$MAP = \frac{\sum_{i=1}^{Q} AP(q_i)}{Q} \tag{16}$$

其中，$Q$ 是已定的查询数据集，$AP(q_i)$ 是单个查询数据 $q_i$ 的平均精确度。

$AP$ 表示精确度的平均值以及召回率的变化，即精准率-召回率曲线的面积。给定单个查询数据 $q_i$，其 $AP$ 的计算式如式(17)所示：

$$AP(q_i) = \frac{\sum_{i=1}^{Q} \delta(q_i,i) P(q_i,i)}{L_q} \tag{17}$$

其中，$L_q$ 表示查询数据 $q_i$ 在训练数据集中的真实近邻数目(如果一对样本中有一个及其以上的相同标记，则称其为真实近邻)；$n$ 表示整个训练数据集中的样本数目；如果第 $i$ 个检索不属于真实近邻，那么 $\delta(q_i,i)=0$，如果第 $i$ 个检索属于真实近邻，那么 $\delta(q_i,i)=1$；$P(q_i,i)$ 表示检索样本中的前 $i$ 个精确度。

(2)精准率-召回率(P-R)

P-R 曲线是自然语言处理、信息检索和机器学习领域常用的评价标准，涉及精确率(Precision)、召回率(Recall)和 F-measure。本文使用 P-R 曲线来衡量不同数据集的检索效果，其定义如表 1 所列。

表 1 文档分类

Table 1 Classification of document

| | 相关正例 | 不相关正例 |
|---|---|---|
| 检索到 | 检索的相关数据 TP | 检索的不相关数据 FP |
| 未检索到 | 未检索的相关数据 FN | 未检索的不相关数据 TN |

精确率是指被检索到的相关数据占所有被检索到的数据的比例，如式(18)所示：

$$P = \frac{TP}{TP+FP} \tag{18}$$

召回率是指所有被检索到的相关数据占所有相关数据的比例，如式(19)所示：

$$R = \frac{TP}{TP+FN} \tag{19}$$

F-measure 结合了精确率和召回率，如式(20)所示：

$$F = \frac{2PR}{P+R} \tag{20}$$

### 4.2 Wiki 数据集上的实验结果与分析

随机选取 Wiki 数据集中 75%的样本作为训练数据集，将剩下的 25%的样本作为测试数据集。将目标函数式(14)的哈希码长度 $k$ 分别设置为 16 位、32 位、64 位和 128 位，然后根据式(16)和式(17)计算 $Map$ 值，并取 10 次实验的 $Map$ 的平均值。在 Wiki 数据集上实验的 $Map$ 值如表 2 所列。

表 2　Wiki 数据集上不同码长的 *Map* 值

Table 2　*Map* value on Wiki dataset with different code

| 任务 | 对比方法 | 哈希码长度 $k$ | | | |
|---|---|---|---|---|---|
| | | 16 位 | 32 位 | 64 位 | 128 位 |
| 图像检索文本 | SCM | 0.2017 | 0.2163 | 0.2243 | 0.2239 |
| | SePH | 0.1935 | 0.2017 | 0.2213 | 0.2314 |
| | CCA-SCH | **0.2382** | **0.2497** | **0.2501** | **0.2617** |
| 文本检索图像 | SCM | 0.2162 | 0.2332 | 0.2364 | 0.2401 |
| | SePH | 0.1943 | 0.2074 | 0.2158 | 0.2260 |
| | CCA-SCH | **0.2276** | **0.2363** | **0.2511** | **0.2638** |

由表 2 可知:1)针对 Wiki 数据集,相对于其他算法,CCA-SCH 算法在所有长度的哈希码上取得较大的 *Map* 值,即检索的效果最佳。具体而言,在图像检索文本任务上,CCA-SCH 算法在不同的哈希码长度上比最好的对比算法的 *Map* 值最高提高了约 10%;在文本检索图像任务上,CCA-SCH 算法在不同的哈希码长度上比最好的对比算法的 *Map* 值最高提高了约 13%。这是因为 CCA-SCH 算法加入了协同矩阵分解,获得了更好的共享语义子空间,而且该算法保持了模态内的语义一致性和模态间的语义一致性,因此检索效果更佳。2)"文本检索图像"的 *Map* 值优于"图像检索文本"的 *Map* 值,即"文本检索图像"的效果优于"图像检索文本"的效果。这种情况表明,CCA-SCH 算法对文本的语义描述比对图像的语义描述更精确、具体。3)一些方法(如 SCM)的 *Map* 值会随着哈希码长度的增加而减小。这说明出现这种情况的方法(如 SCM)对哈希码长度有一定的限制因素,学习目标函数的质量可能会随着哈希码长度的增加而降低。

为了更直观、准确地对比各方法的性能,分别计算当哈希码长度为 32 位和 64 位时各种方法在 Wiki 数据集上的精准率-召回率,得出的 P-R 曲线图如图 1 和图 2 所示。

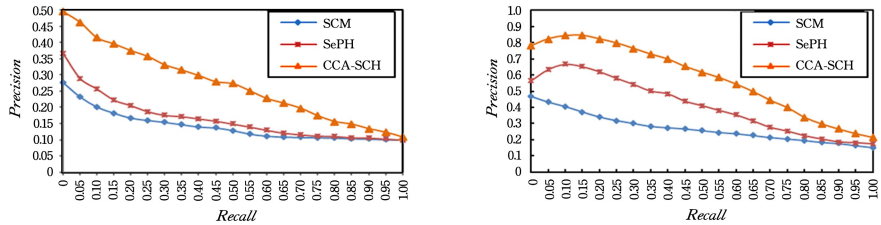


图 1　$k=32$ 时 Wiki 数据集的 P-R 图

Fig. 1　Precision-Recall curves on Wiki dataset with $k=32$

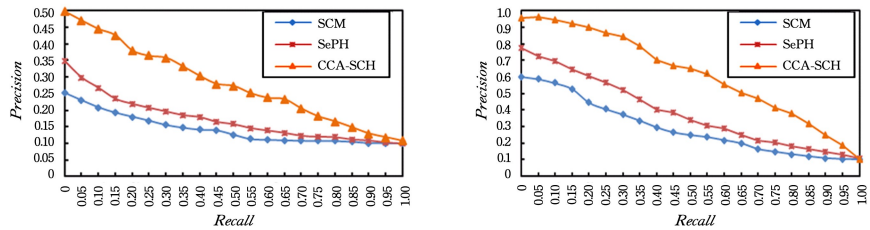


图 2　$k=64$ 时 Wiki 数据集的 P-R 图

Fig. 2　Precision-Recall curves on Wiki dataset with $k=64$

由图 1 和图 2 可知,CCA-SCH 算法优于基准算法。SCM 算法只考虑了异构多模态数据对之间的联系,没有考虑高层语义信息;SePH 算法只考虑了高层语义信息,没有考虑异构多模态数据对之间的联系,属于无监督检索;SCM 算法、SePH 算法没有充分考虑模态间和模态内的高层语义一致性;本文所提出的 CCA-SCH 算法既考虑了异构多模态数据对的语义联系,又保持了模态间和模态内的相似性,因此具有更好的检索性能。

### 4.3　NUS-WIDE 数据集上的实验结果与分析

随机选取 NUS-WIDE 数据集中的 2000 个样本作为训练数据集,将剩下的 1000 个样本作为测试数据集。将目标函数式(14)的哈希码长度 $k$ 分别设置为 16 位、32 位、64 位和 128 位,然后根据式(16)和式(17)计算 *Map* 值,并取 10 次实验的 *Map* 的平均值。实验的 *Map* 值如表 3 所列。

由表 3 可知,针对 NUS-WIDE 数据集,相对于其他的算法,CCA-SCH 算法在所有长度的哈希码上均取得较大的 *Map* 值,即检索的效果最佳。具体而言,在图像检索文本任务上,CCA-SCH 算法在不同的哈希码长度上比最好的对比算法的 *Map* 值最高提高了约 18%;在文本检索图像任务上,CCA-SCH 算法在不同的哈希码长度上比最好的对比算法的 *Map* 值最高提高了约 20%。

表 3　Wiki 数据集上不同码长的 *Map* 值

Table 3　*Map* value on Wiki dataset with different code

| 任务 | 对比方法 | 哈希码长度 $k$ | | | |
|---|---|---|---|---|---|
| | | 16 位 | 32 位 | 64 位 | 128 位 |
| 图像检索文本 | SCM | 0.3183 | 0.3276 | 0.3330 | 0.3476 |
| | SePH | 0.3293 | 0.3301 | 0.3349 | 0.3322 |
| | CCA-SCH | **0.5739** | **0.5802** | **0.5879** | **0.5917** |
| 文本检索图像 | SCM | 0.4850 | 0.4872 | 0.4886 | 0.4932 |
| | SePH | 0.5923 | 0.6044 | 0.6158 | 0.6205 |
| | CCA-SCH | **0.6219** | **0.6507** | **0.6672** | **0.6793** |

图 3 和图 4 是当哈希码长度为 32 位和 64 位时,基于 NUS-WIDE 数据集的 P-R 曲线图。由图 3 和图 4 可知,针对不同的数据集,CCA-SCH 算法的 P-R 曲线图均优于基准算法。本文所提出的 CCA-SCH 算法保持了模态内和模态间的

语义一致性，因此该算法仍然比基准算法的检索精准度高，在异构多模态数据检索中的泛化性能更好。

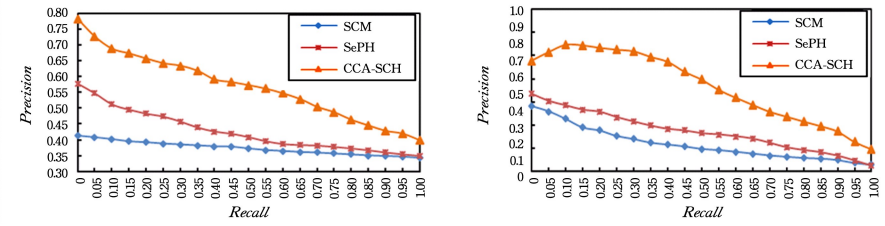


图 3 $k=32$ 时 NUS-WIDE 数据集的 P-R 图

Fig. 3 Precision-Recall curves on NUS-WIDE dataset with $k=32$

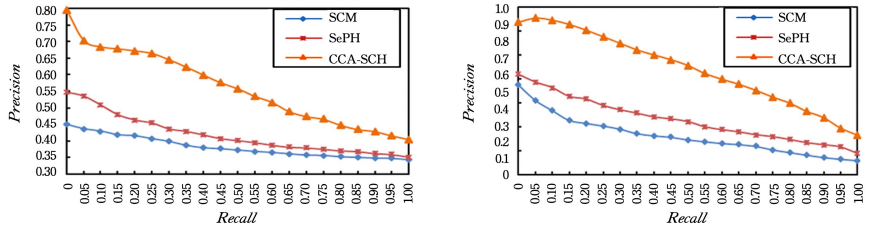


图 4 $k=64$ 时 NUS-WIDE 数据集的 P-R 图

Fig. 4 Precision-Recall curves on NUS-WIDE dataset with $k=64$

**结束语** 针对异构多模态数据检索的问题，基于哈希的异构多模态数据检索不仅可以缩减数据存储空间，还可以提高检索速度。本文提出了基于 CCA 语义一致性的哈希检索算法，该算法建立了文本数据和图像数据底层特征的低维共享语义子空间，结合文本语义、图像语义和图像与文本语义相关性生成目标函数，最后将该算法扩展到异构多模态数据检索上。实验结果表明，CCA-SCH 算法能够有效地提升检索精度。下一步工作是引入视频和音频等数据，将本文算法应用于更多模态间的相互检索中。

## 参 考 文 献


[1] MA Q,GU Y,ZHANG T C,et al. A Heterogeneous Multi-Source Multi-Mode Sensory Data Acquisition Method Based on Data Quality[J]. Chinese Journal of Computers,2013,36(10):2120-2131.

[2] MAO X J,YANG Y B. Semantic Hashing with Image Subspace Learning[J]. Journal of Software,2014,25(8):1781-1793.

[3] CAO Y D,LIU Y Y,SUN F M,et al. LSH with low space complexity for image retrieval[J]. Computer Engineering & Science,2015,37(2):379-383.

[4] ZHANG L. Research on Locality Sensitive Hashing Based Approximate Nearest Neighbor(s) Searching Algrothm[D]. Nanjing: Nanjing University of Posts and Telecommunications, 2015.

[5] LIU H,WANG R,SHAN S,et al. Deep Supervised Hashing for Fast Image Retrieval[C]// IEEE Conference on Computer Vision and Pattern Recognition. IEEE Computer Society, 2016: 2064-2072.

[6] WEN Q F,WANG J M,ZHU H,et al. Distributed Learning to Hash for Approximate Nearest Neighbor Search[J]. Chinese Journal of Computers,2017,40(1):192-206.

[7] TANG J,WANG K,SHAO L. Supervised Matrix Factorization Hashing for Cross-Modal Retrieval[J]. IEEE Transactions on Image Processing,2016,25(7):3157-3166.

[8] ZHANG L,ZHAO Y,ZHU Z F. Advances in Semantically Shared Subspace Learning for Cross-Media Data[J]. Chinese Journal of Computers,2017,40(6):168-195.

[9] WANG K,TANG J,WANG N,et al. Semantic Boosting Cross-Modal Hashing for efficient multimedia retrieval[J]. Information Sciences,2016,330(C):199-210.

[10] WANG D,GAO X,WANG X,et al. Multimodal Discriminative Binary Embedding for Large-Scale Cross-Modal Retrieval[J]. IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society,2016,25(10):4540-4554.

[11] LI X,GAO L,XU X,et al. Kernel based Latent Semantic Sparse Hashing for Large-scale Retrieval from Heterogeneous Data Sources[J]. Neurocomputing,2017,253:89-96.

[12] DATAR M,IMMORLICA N,INDYK P,et al. Locality-sensitive hashing scheme based on p-stable distributions[C]// Twentieth Symposium on Computational Geometry. ACM,2004:253.

[13] DING G,GUO Y,ZHOU J. Collective Matrix Factorization Hashing for Multimodal Data[C]// Computer Vision and Pattern Recognition. IEEE,2014:2083-2090.

[14] ZHU Y Y. Research on Semantic Consistency and Matrix Factorization based Cross-modal Hashing Retrieval[D]. Hefei: Anhui University,2017.

[15] HOTELLING H. Relations Between Two Sets of Variates[J]. Biometrika,1936,28(3/4):321-377.

[16] ZHANG D,LI W J. Large-scale supervised multimodal hashing with semantic correlation maximization[C]// Twenty-Eighth AAAI Conference on Artificial Intelligence. AAAI Press,2014: 2177-2183.

[17] LIN Z,DING G,HU M,et al. Semantics-preserving hashing for cross-view retrieval[C]// 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2015: 3864-3872.