Journal of Computer Applications
计算机应用, 2021, 41(4): 1106 - 1112
ISSN 1001-9081
CODEN JYIIDU
2021-04-10
http://www.joca.cn

文章编号:1001-9081(2021)04-1106-07
DOI:10.11772/j.issn.1001-9081.2020071047

在线哈希算法研究综述

郭一村*,陈华辉
(宁波大学 信息科学与工程学院,浙江 宁波 315000)
(*通信作者电子邮箱493187400@qq.com)

摘　要:在当前大规模数据检索任务中,学习型哈希方法能够学习紧凑的二进制编码,在节省存储空间的同时能快速地计算海明空间内的相似度,因此近似最近邻检索常使用哈希的方式来完善快速最近邻检索机制。对于目前大多数哈希方法都采用离线学习模型进行批处理训练,在大规模流数据的环境下无法适应可能出现的数据变化而使得检索效率降低的问题,提出在线哈希方法并学习适应性的哈希函数,从而在输入数据的过程中连续学习,并且能实时地应用于相似性检索。首先,阐释了学习型哈希的基本原理和实现在线哈希的内在要求;接着,从在线条件下流数据的读取模式、学习模式以及模型更新模式等角度介绍在线哈希不同的学习方式;而后,将在线学习算法分为六类:基于主-被动算法、基于矩阵分解技术、基于无监督聚类、基于相似性监督、基于互信息度量和基于码本监督,并且分析这些算法的优缺点及特点;最后,总结和讨论了在线哈希的发展方向。

关键词:在线学习;学习型哈希;无监督学习;监督学习;最近邻检索
中图分类号:TP391　　**文献标志码**:A

Survey on online hashing algorithm

GUO Yicun*, CHEN Huahui
(*Faculty of Electrical Engineering and Computer Science, Ningbo University, Ningbo Zhejiang* 315000, *China*)

Abstract: In the current large-scale data retrieval tasks, learning to hash methods can learn compact binary codes, which saves storage space and can quickly calculate the similarity in Hamming space. Therefore, for approximate nearest neighbor search, hashing methods are often used to improve the mechanism of fast nearest neighbor search. In most current hashing methods, the offline learning models are used for batch training, which cannot adapt to possible data changes appeared in the environment of large-scale streaming data, resulting in reduction of retrieval efficiency. Therefore, the adaptive hash functions were proposed and learnt in online hashing methods, which realize the continuous learning in the process of inputting data and make the methods can be applied to similarity retrieval in real-time. Firstly, the basic principles of learning to hash and the inherent requirements to realize online hashing were explained. Secondly, the different learning methods of online hashing were introduced from the perspectives such as the reading method, learning mode, and model update method of streaming data under online conditions. Thirdly, the online learning algorithms were further divided into six categories, that is, categories based on passive-aggressive algorithms, matrix factorization technology, unsupervised clustering, similarity supervision, mutual information measurement, codebook supervision respectively. And the advantages, disadvantages and characteristics of these algorithms were analyzed. Finally, the development directions of online hashing were summarized and discussed.

Key words: online learning; learning to hash; unsupervised learning; supervised learning; nearest neighbor search

0　引言

随着大数据时代网络数据不断增加,大规模的数据集对传统的机器学习方式提出了重大挑战。在各种检索方式中,最近邻(Nearest Neighbor, NN)检索[1-3]在多种学习算法如基于标签的图像注释、语义分割、视频分割、文本检索[4]、内容检索[5]、物体识别等领域内得到了广泛应用。最近邻检索的主要任务是对于给定一个查询点检索一个语义最近邻数据集。传统基于空间划分的算法[6]虽然能得到比较精确的结果,但是在高维数据集上的学习和检索的时间效率上都不高,因此对于高维度数据的最近邻查询往往使用乘积量化的策略,映射到低维子空间进行近似最近邻(Approximate NN, ANN)[7]检索。学习型哈希[8-9]通过将数据表示为紧凑的二进制码形式,很方便地使用异或运算快速计算数据间相似度,将原样本空间相似的两个数据点映射到海明空间里接近的两个点。学习型哈希不仅能大大减少数据的存储空间和运算开销,还能降低数据维度,从而显著提高大数据学习系统的效率。

在线学习型哈希算法的关系如图1所示。本文首先介绍了学习型哈希算法的原理;然后介绍了在线哈希的难点以及在线哈希学习所采取的不同方式,随后讨论在线哈希的各种算法的发展状况并总结,对在线哈希未来发展方向进行了

收稿日期:2020-07-21;**修回日期**:2020-10-19;**录用日期**:2020-10-20。　**基金项目**:国家自然科学基金资助项目(61572266)。
作者简介:郭一村(1995—),男,河南平顶山人,硕士研究生,主要研究方向:数据挖掘、深度学习;　陈华辉(1964—),男,浙江宁波人,教授,博士,CCF会员,主要研究方向:数据库、大数据处理。

展望。

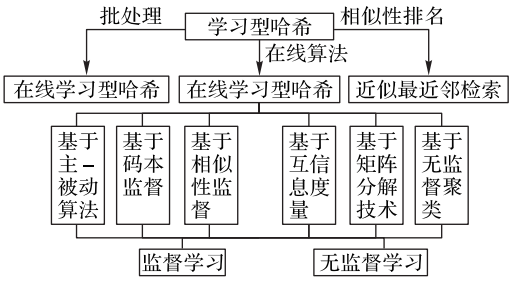


图1　在线学习型哈希算法关系图

Fig. 1　Relation chart of online learning to hash algorithms

1　学习型哈希

学习型哈希由数据、哈希函数、目标方程三个基本要素构成。海明距离用来衡量哈希值之间的关联程度,在海明空间内反映数据的相似性,因此哈希学习的过程就是建立高维度空间到较低维度海明空间的映射关系,并设计合理的目标方程量化损失减少两个空间分布的差异。也就是说相似的数据在海明空间内的距离足够接近,在最近邻检索数据时尽可能地找到相似数据;与之对应的,不相似的数据在海明空间内的距离足够疏远,不同类别数据更容易被区分开。

假设输入数据为 n 个 d 维的向量 $\boldsymbol{X} \in \mathbf{R}^{d \times n}$,而学习型哈希模型的目标是要生成对应数量的二进制哈希码 $\boldsymbol{y} = \{y_1, y_2, \cdots, y_k\}$,位数为 k。每一位哈希码都由一个哈希函数进行映射,得到一位哈希码,依次就可以计算出一个数据样本 $\boldsymbol{x} \in \mathbf{R}^d$ 的所有哈希码:

$$\boldsymbol{y} = \left\{h_1(\boldsymbol{x}), h_2(\boldsymbol{x}), \cdots, h_k(\boldsymbol{x})\right\} \tag{1}$$

样本数据被哈希函数映射为一批哈希码的过程就是向量经过一组线性运算后再进行二值化:

$$\boldsymbol{Y} = H(\boldsymbol{X}) = \left\{h_1(\boldsymbol{X}), h_2(\boldsymbol{X}), \cdots, h_k(\boldsymbol{X})\right\} \tag{2}$$

$$h_k(\boldsymbol{x}) = \mathrm{sgn}\left(f\left(\boldsymbol{w}_k{}^{\mathrm{T}}\boldsymbol{x} + b_k\right)\right) \tag{3}$$

哈希模型可以大致分为数据独立和数据依赖技术。数据独立的技术往往设置若干个固定的哈希函数对数据进行映射。在早期位置敏感哈希(Locality Sensitive Hashing, LSH)算法[10]应用中,每一个哈希值当作是一个容器:哈希桶(Hash Bin)。哈希桶被用来构建哈希表,查询操作相当于列表搜索。原始样本数据经过哈希函数运算得到一个哈希值,这个哈希值对应与之相似的样本。数据之间的相似度并不取决于数据本身,如余弦相似度[11],除此之外没有其他更多的信息。这种数据独立的方式导致随着学习的数据增加,会有越来越多的数据哈希值产生碰撞,那些相似的数据共享同一个哈希桶,增加了检索所消耗的时间。为缓解碰撞则需要增大哈希码的长度,或者使用多个哈希桶,然而这又添加额外的存储,并且学习到的模型只适合特定的数据分布,泛化性比较弱。但是在大规模数据集上应用时,学习过程的计算成本至关重要,这种速度缓慢的将大批量数据集中学习的方式很难适应随着数据增长而变化的数据集以更新哈希学习模型。

为了克服这些问题,近年来的研究重点开始转向数据依赖的哈希技术。数据依赖型哈希通过分析数据结构特征及分布信息自动学习哈希函数,通常分为无监督方法和有监督方法以及半监督方法。无监督哈希方法[12-14]根据数据原始分布学习哈希函数,无须任何监督信息。与之相比,有监督哈希方法因引入了监督信息显著地提高了检索相似度而越来越被受到关注。有监督的学习型哈希方法利用数据标签来获得语义相似度对生成的哈希码进行有效监督[15-17]。查询时按照海明空间距离反映的相似性进行排序,选取一定数量的相似样本。虽然在整个数据集中进行检索成本偏高,但是二进制码的距离计算十分简便,并且保持了更多原始空间上的相似性。另外如半监督哈希[18]方式使用有标签和无标签的数据学习哈希函数,解决标签获取困难的问题,同时避免出现模型过拟合。随着深度学习快速发展,且深度学习模型往往具备强大的表征能力,于是近年来一些研究将深度学习与哈希学习两者结合强化模型对数据复杂特征的表示[19]。得益于这些监督或半监督的方法,模型能够在多媒体数据上学习到共同的哈希函数,可以跨模态对数据进行检索[20-21]。

2　在线学习型哈希

在线哈希学习是一种特殊的学习型哈希方式,关键在于训练前后对数据的依赖性。离线的哈希学习假设所有数据都是已知的,基于全局优化的目标,数据被重复挑选用来纠正学习初期所产生的偏差。这就带来了在线哈希学习中最主要的矛盾:即随着模型更新带来的"遗忘"问题,因此在线哈希学习的目的是寻找一种变化与保持的平衡策略。

2.1　在线哈希方式

尽管已有的学习型哈希算法已有很好的性能,但是面对大量流数据时,仍然存在很多缺陷:1)当数据集发生变化或扩展时,为适应新的数据分布,必须将全部数据纳入计算以重新学习模型所有相关参数,这显然是十分低效的。在实际的应用中,数据往往以数据流的方式输入,而模型很难作出频繁的响应。2)对于许多大规模的数据集,数据以分布式的形式存储在磁盘中。每次训练新数据时,需要将所有先前数据调入内存处理,不仅对于现有的内存容量是无法接受的,同时也给中央处理器的调度增加了很大压力。3)训练后的数据仍然长期保存以应对多次训练,耗费大量存储空间。

针对以上问题,在线学习型哈希进行了相关研究,即哈希模型需要满足几个重要的条件:1)在训练原有数据的基础上,能够在数据流中学习哈希函数并且不依赖先前存储的数据;2)学习到的哈希函数产生的哈希码分布仍然符合相似性分布,使相似(不相似)数据的哈希码保持一致的相似(不相似)性,这和传统的哈希学习方式要求一致;3)学习速度加快,以响应现实中较频繁的最近邻检索。

在线学习型哈希仍然遵循学习型哈希的基本原则,但许多传统的学习方式并不适合引入在线学习环境。现有的在线哈希算法采用了多种在线学习方式,可以从不同角度对在线哈希方式进行区分。

2.1.1　单次学习与多次学习

单次学习可大幅降低学习成本,数据只被用来训练一次,所以不必长期存储使得模型可以应对更广阔的数据。大部分现有方法如主被动算法、聚类算法等都适用单次学习。一种折中的方法是保留少部分数据作为样本库多次学习,缓解模型更新的偏差。

2.1.2　监督和无监督

和离线哈希学习类似,在线哈希学习也可被分为无监督在线哈希和监督在线哈希。无监督哈希学习分析样本数据之间的关系,分析相似程度,如降维提取特征值和使用自组织映射网络;监督学习的方式往往利用标签信息带来更高的检索精度,解决语义鸿沟问题。每一个样本数据都有对应的标签提供这个数据的类别信息,由此可以计算出数据的语义相似

度，比如较早的基于主-被动算法，和后来的基于适应性哈希函数。或者可以采取矢量量化的方式，将标签直接生成码本向量，直接对哈希码进行直接监督。而码本向量需要针对不同位数的哈希码做相应的转换，因此要去除码本向量之间的相关性以及减少降维运算时产生的误差来保证监督的可靠性。

2.1.3 数据点、数据对和数据块

在线哈希在模型训练时可以按照三种不同的数据划分层次：数据点、数据对和数据块（或数据列表）进行参数更新。数据点的形式不需要进行相似度的量化，当一个数据点样本输入，模型可以通过标签生成目标哈希码进行直接监督，实际上相当于聚类或分类问题。哈希码作为高维空间向量对数据类别进行划分（如图2所示），局限于标签所指示的类别数量，不适合类别复杂且多样的数据。数据对和数据块的区别在于以一对一还是一对多的方式保持相似性。数据块在输入模型时需要计算相似矩阵来指示数据间的相似关系，然而在数据变化较大时很难计算全局的均值做归一化处理，给模型造成频繁计算问题。

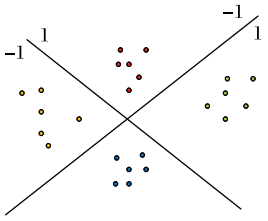


图2 二进制哈希码作为分类向量

Fig. 2 Binary hash code used as classification vector

2.2 在线哈希算法

本节将回顾在近年来关于在线哈希学习方面表现出检索效率较高的各种算法，并比较它们的不同。

2.2.1 基于主-被动算法

受到主-被动算法[22]的启发，Huang等[23]首次提出了哈希函数在线学习方式，将主-被动算法适用于每对新样本数据的哈希函数。在第t批次数据中，给出新的一对数据$\left(\boldsymbol{x}_i^t,\boldsymbol{x}_j^t\right)$和它们的相似性标签$s_{ij}\in\{-1,+1\}$，模型相应地更新哈希函数，使其能够正确计算新数据的哈希码，同时与旧的哈希函数足够接近。于是目标方程用来约束参数的变化，同时用一个非负变量ξ松弛化约束：

$$\boldsymbol{W}^t=\arg\min_{\boldsymbol{w}}\frac{1}{2}\left\|\boldsymbol{W}-\boldsymbol{W}^{t-1}\right\|_{\mathrm{F}}^2+C\xi \tag{4}$$

$$\text{s.t.}\quad L\left(\boldsymbol{W};\left\langle\left(\boldsymbol{x}_i^t,\boldsymbol{x}_j^t\right),s_{ij}\right\rangle\right)\leqslant\xi\ \text{且}\ \xi\geqslant 0$$

单纯使用主-被动算法面临两个明显的问题：1）算法每次以数据对的形式进行优化，使得模型频繁更新限制了优化效率；2）如果到来的数据越来越多变，则此算法可能会面临收敛困难的问题。

针对第二个问题可以采用多模型的优化方式[24]，分别对两种情况下的模型选择采取不同策略。为了控制模型参数更新频率，引入了阈值来量化损失，如果超过这个阈值，则认为哈希码得到了相应的匹配，模型参数不变；反之则要更新参数。在两个数据样本相似的情况下，海明距离大于阈值α，不相似时海明距离小于阈值βr，产生大于0的损失$R\left(\boldsymbol{H}^t,s^t\right)$：

$$R\left(\boldsymbol{H}^t,s^t\right)=\begin{cases}\max\left\{0,D_{\boldsymbol{h}}\left(\boldsymbol{h}_i{}^t,\boldsymbol{h}_j{}^t\right)-\alpha\right\}, & s^t=1\\ \max\left\{0,\beta r-D_{\boldsymbol{h}}\left(\boldsymbol{h}_i{}^t,\boldsymbol{h}_j{}^t\right)\right\}, & s^t=-1\end{cases} \tag{5}$$

文献[25]提出根据当前数据分布采用动态的损失阈值，使优化目标松弛为一个置信区间，同时约束损失函数变化，增强模型的稳定性。

Weng等[26]在哈希函数的框架基础上又增加了一个映射函数。或者说将模型分为了两部分，首先由哈希函数迭代量化（Iterative Quantization，ITQ）[12]映射为哈希码，再经过映射函数进行调整生成一个新的哈希码，来适应持续到来的新数据：

$$\boldsymbol{g}=\operatorname{sgn}\left(\boldsymbol{P}^{\mathrm{T}}\boldsymbol{h}\right) \tag{6}$$

为了获得更好的监督学习效果，使用独热（One-Hot）编码向量的标签$\boldsymbol{y}_i$，类似的投影生成理想的哈希码用作监督：

$$\boldsymbol{g}_i=\operatorname{sgn}\left(\boldsymbol{L}^{\mathrm{T}}\boldsymbol{y}_i\right) \tag{7}$$

单独地更新每个映射矩阵中的向量$\boldsymbol{p}_k$，按位与理想哈希码计算损失：

$$l\left(\boldsymbol{g}_i,\boldsymbol{h}_i\right)=\begin{cases}0, & \boldsymbol{g}_i\left(\boldsymbol{p}^{\mathrm{T}}\boldsymbol{h}_i\right)\geqslant 1\\ 1-\boldsymbol{g}_i\left(\boldsymbol{p}^{\mathrm{T}}\boldsymbol{h}_i\right), & \boldsymbol{g}_i\left(\boldsymbol{p}^{\mathrm{T}}\boldsymbol{h}_i\right)<1\end{cases} \tag{8}$$

同时约束参数更新时的变化：

$$\boldsymbol{p}_i=\arg\min\frac{1}{2}\left\|\boldsymbol{p}-\boldsymbol{p}_{i-1}\right\|^2+C\xi \tag{9}$$

$$\text{s.t.}\quad l_i\leqslant\xi\ \text{且}\ \xi\geqslant 0$$

映射函数旨在优化二进制哈希函数，纠正固定哈希函数带来的偏差，使哈希码适应新的数据分布；同时基于主成分分析（Principal Component Analysis，PCA）降维的哈希函数本身带来的误差并不能消除，限制了优化上限。

2.2.2 基于矩阵分解技术

Leng等[27]提出了一个在流数据中学习哈希函数的思想：用一个尺寸更小的数据集模块，保存数据主要特征，之后在线学习哈希函数，计算哈希函数的过程就会有一个比较低的计算复杂度和存储空间。

以往的实验结果表明，哈希码长度越长，对原有数据相似性拟合度越好。加入平衡约束和不相关约束能在有限的长度内提高哈希码的表达能力[28]：每一位哈希码应当有50%的概率为+1或-1；不同的位之间相互独立，即$\sum_i\boldsymbol{h}_k\left(\boldsymbol{x}_i\right)=0(k=1,2,\cdots,n)$。

在线概要哈希将上述约束松弛为最大化哈希码的方差，即最大化协方差矩阵的迹，防止模型优化变成非确定性多项式（Non-Deterministic Polynomial，NP）困难问题。主要任务为求解方程得到最优解$\boldsymbol{W}$，即解$(\boldsymbol{X}-\mu)$协方差矩阵的前r个最大的特征值对应的特征向量。然而直接使用最优化的矩阵$\boldsymbol{W}$作为哈希投影将会带来不平衡的问题，因此需要在训练之前使数据零均值化。

$$\max_{\boldsymbol{W}\in\mathbf{R}^{d\times r}}\operatorname{tr}\left(\boldsymbol{W}^{\mathrm{T}}(\boldsymbol{X}-\mu)(\boldsymbol{X}-\mu)^{\mathrm{T}}\boldsymbol{W}\right) \tag{10}$$

$$\text{s.t.}\quad \boldsymbol{W}^{\mathrm{T}}\boldsymbol{W}=\boldsymbol{I}_r$$

使用基于矩阵分解的数据块进行哈希学习和PCA降维过程类似，实质上是在线求解特征值或奇异值的过程。另外在线概要哈希针对流数据提出一个零均值块算法弥补了数据的均值变化问题。

文献[29]通过采用子采样随机的阿达玛变换的方式加快了矩阵分解的进程，加快了学习速度。其后Weng等[30]加入了

样本相似度作为监督信息,提高了检索精确度。

2.2.3　基于无监督聚类

文献[31]利用自适应的K均值聚类进行无监督哈希函数学习。但是K均值聚类哈希算法本质上是基于批次的学习模型,具有很高的时间和存储的复杂性。Chen等[32]将传统的自组织映射(Self-Organizing Mapping, SOM)网络扩展到高维空间,形成网格状超立方体。聚类中心作为超立方体的顶点,顶点位置信息引导生成二进制哈希码(如图3所示)。

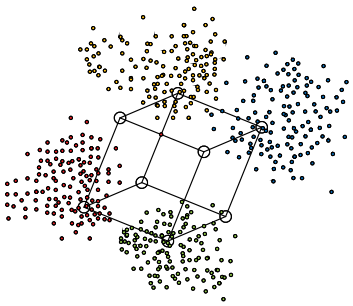


图3　三维空间的自组织映射网络

Fig. 3　SOM network of 3D space

使用PCA将数据降维到与超立方体同一维度,减小量化误差的同时保持了海明空间与欧几里得空间之间的亲和度。但另外一方面,SOM一次只能针对特定维度的数据进行学习,若数据样本维度过高,映射到超立方体神经网络的编码则可能会超过哈希码表示的限度,于是模型不得不再次降维而产生二次近似误差。

2.2.4　基于相似性监督

Cakir等[33]指出批处理学习在逐渐增大的数据集上的计算代价和内存需求难以在合理时间内满足,因此提出了一个基于随机梯度下降的在线学习算法,在流数据中迭代的更新哈希函数,在线适应新的数据分布,并且比传统学习速度更快。

当一对新数据点$\{\boldsymbol{x}_i,\boldsymbol{x}_j\}$到来时,利用相似性信息$s_{ij}$进行随机梯度下降找到适应数据变化的哈希函数。损失函数用来保持海明亲和度:

$$l\left(f\left(\boldsymbol{x}_i\right),f\left(\boldsymbol{x}_j\right);\ \boldsymbol{W}\right)=\left(f\left(\boldsymbol{x}_i\right)^{\mathrm{T}}f\left(\boldsymbol{x}_j\right)-bS_{ij}\right)^2 \tag{11}$$

通过随机梯度下降,每次迭代选择一对数据点进行在线更新:

$$\min_{\boldsymbol{W}\in\mathbf{R}^{m\times b}}J(\boldsymbol{W})=\sum_{ij}l\left(f\left(\boldsymbol{x}_i\right),f\left(\boldsymbol{x}_j\right);\ \boldsymbol{W}\right)=\left\|\boldsymbol{F}^{\mathrm{T}}\boldsymbol{F}-b\boldsymbol{S}\right\|_{\mathrm{F}}^2 \tag{12}$$

只针对新数据设计适应性的哈希函数忽略了在线连续学习的情况下,不仅随着旧数据样本越来越多模型承受着“退化”的风险,同时相似数据与不相似数据的不均衡也往往会导致相似数据之间并没有得到充分学习而降低了检索时的精确度。Lin等[34]主要关注了在流数据的哈希模型学习中新数据与现有数据对应的相似性分布,以及在线学习中数据的不平衡问题(相似数据与不相似数据的数量不均等),采用了一种新颖的平衡相似性,使得在线学习中使用离散优化成为可能。

在线学习环境中,学习数据不断增加,将旧数据重新读取学习是十分困难的,这也违背了在线哈希学习的框架。为解决模型在旧数据上的偏离问题,首先需要对新旧数据样本分开来讨论,生成的哈希码也被划为两部分:$\boldsymbol{B}_s^t$和$\boldsymbol{B}_e^t$。然后在新旧数据之间根据相似度标签进行监督学习:

$$\min_{\boldsymbol{B}_s^t,\boldsymbol{B}_e^t}\left\|\boldsymbol{B}_s^{t\,\mathrm{T}}\boldsymbol{B}_e^t-k\boldsymbol{S}^t\right\|_{\mathrm{F}}^2 \tag{13}$$

$$\text{s.t.}\quad \boldsymbol{B}_s^t\in\{-1,1\}^{k\times n_t},\boldsymbol{B}_e^t\in\{-1,1\}^{k\times n_t}$$

容易得出,式(13)应分为两种情况计算,即相似S_1^t和不相似S_2^t。显然因为学习到越来越多的数据,S_1^t也远小于S_2^t,则拆分后的两项会出现不平衡:

$$\sum_{i,j,S_{ij}^t=1}\left(\boldsymbol{b}_{si}^{t\,\mathrm{T}}\boldsymbol{b}_{ej}^t-k\right)^2+\sum_{i,j,S_{ij}^t=-1}\left(\boldsymbol{b}_{si}^{t\,\mathrm{T}}\boldsymbol{b}_{ej}^t+k\right)^2 \tag{14}$$

$$\text{s.t.}\quad \boldsymbol{b}_{si}^t\in\{-1,1\}^k,\boldsymbol{b}_{ej}^t\in\{-1,1\}^k$$

如图4所示,新数据在数据集中的相似数据远少于不相似的。事实上模型训练的目的是利用相似数据能够得到较紧密的哈希码,因为在检索时所需要的是相似数据样本而非不相似的,学习时间被大量地消耗在了不相似数据的学习上而致使学习效率降低。最直接的做法就是调整两式的超参数使之均衡,然而这在实验上模型很难被优化,且随着数据流的增加超参数也要实时调整。具有平衡相似度的在线离散哈希提出了平衡相似度的概念,用两个特征值来平衡相似与不相似时的相似度:

$$\tilde{S}_{ij}^t=\begin{cases}\eta_sS_{ij}^t, & S_{ij}^t=1\\ \eta_dS_{ij}^t, & S_{ij}^t=-1\end{cases} \tag{15}$$

添加平衡相似度后得到能够平衡学习两种数据分布的目标方程:

$$\sum_{i,j,S_{ij}^t=1}\left(\boldsymbol{b}_{si}^{t\,\mathrm{T}}\boldsymbol{b}_{ej}^t-k\tilde{S}_{ij}^t\right)^2+\sum_{i,j,S_{ij}^t=-1}\left(\boldsymbol{b}_{si}^{t\,\mathrm{T}}\boldsymbol{b}_{ej}^t+k\tilde{S}_{ij}^t\right)^2 \tag{16}$$

$$\text{s.t.}\quad \boldsymbol{b}_{si}^t\in\{-1,1\}^k,\boldsymbol{b}_{ej}^t\in\{-1,1\}^k$$

平衡相似度同时调节了训练过程中相似和不相似数据与新旧数据两类失衡的问题,防止模型出现退化和遗忘。因此在学习过程中不得不保留一部分旧数据,消耗部分存储空间维持模型性能,在数据集较大时,旧数据如何表示复杂的原始分布仍然存在挑战。

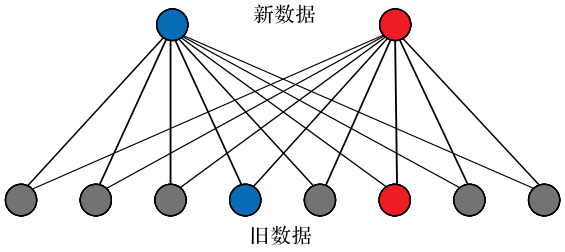


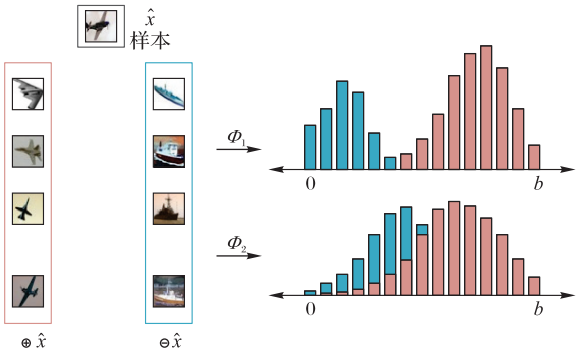
图4　新数据和旧数据构成非对称图

Fig. 4　Asymmetric graph constructed by new data and old data

2.2.5　基于互信息度量

Cakir等[35]致力于分离不相似的数据在海明空间上的分布,提出了一种新的衡量哈希码质量的方式,对比一直使用的基于海明距离对数据相似性的度量,互信息基于二进制码的分布量化哈希函数的质量显得更加直观有效。在被哈希函数映射进哈希桶的模型中,考察哈希码编码分布往往会各不相同:如图5所示,统计某个样本相似数据的哈希码分布情况可以得到一个近似的高斯分布,其他不相似的哈希码又会得到另一个不同的分布。两种分布重叠的区域可能使得相似和不相似数据的哈希码产生重复而导致误差。在理想情况下,这两种分布尽量疏离,相似的数据紧密分布,那么重叠区域的面积则较小,两种数据的编码的重合程度也会较小,海明距离就自然较远。基于上述对于哈希码分布的认识,便得出互信息

的概念，基于互信息的取值反映模型的性能。



(a) 相似相本 (b) 不相似相本 (c) 被不同哈希函数映射后哈希码分布

图5 哈希码分布指示映射函数质量

Fig. 5 Hash code distribution indicating quality of mapping function

得到一个样本 $\hat{x}$ 以及相似的样本 $\oplus\hat{x}$ 和不相似的样本 $\ominus\hat{x}$，经过哈希函数 $\boldsymbol{\Phi}$ 映射出两种不同的哈希码分布 $P\left(D_{\hat{x},\boldsymbol{\Phi}}|C_{\hat{x}}=1\right)$ 和 $P\left(D_{\hat{x},\boldsymbol{\Phi}}|C_{\hat{x}}=0\right)$，记作 P_D^+ 和 P_D^-，$C_{\hat{x}}$ 用来指示样本是否为相似。计算这些分布的条件熵可得到互信息的值：

$$I\left(D_{\hat{x},\boldsymbol{\Phi}};\ C_{\hat{x}}\right)=H\left(C_{\hat{x}}\right)-H\left(C_{\hat{x}}|D_{\hat{x},\boldsymbol{\Phi}}\right)=H\left(D_{\hat{x},\boldsymbol{\Phi}}\right)-H\left(D_{\hat{x},\boldsymbol{\Phi}}|C_{\hat{x}}\right) \tag{17}$$

显然互信息 $I(D;\ C)$ 取值越大时，分布的不确定性就越低，体现出哈希函数 $\boldsymbol{\Phi}$ 能够将两种分布映射得更加离散，减少了哈希码重叠的可能性。在理想的状态下，互信息足够大，哈希码几乎不发生重叠，P_D^+ 与 P_D^- 为独立分布。利用互信息可以对模型质量进行整体的检验：

$$Q(\boldsymbol{\Phi})=\int_X I\left(D_{\hat{x},\boldsymbol{\Phi}};\ C_{\hat{x}}\right)p(\hat{x})\mathrm{d}\hat{x} \tag{18}$$

由于在实际情况下不可能加载所有样本，因此在流数据中采样一部分作为样本库：

$$Q_R(\boldsymbol{\Phi})=\frac{1}{|R|}\sum_{x'\in R}I\left(D_{x',\boldsymbol{\Phi}};\ C_{x'}\right) \tag{19}$$

Q_R 可以当作是一个触发器，新数据和样本库中的数据同时被哈希函数映射，如果函数保证了原有的互信息，哈希函数才可以更新，这样就控制了不同数据间的映射分布在添加新数据后也是离散的，维持了模型性能。另一方面，样本库中的数据也不能无限增长，随着学习到的哈希码越来越多，样本库表示性的下降也是无法避免的。再者算法使用了哈希桶的方式，虽然在进行互信息的优化后不相似的样本哈希码得到较好分离，却又加重相似样本的哈希码的重叠程度，增大了检索时的复杂度。

2.2.6 基于码本监督

在先前的各种学习方式中，数据都是以批次或数据对的形式进行学习，而无法立刻学习单个数据；又考虑到新到来的数据可能会具有原来未包含的标签，而产生错误分类。于是受通信领域的信号传输模型的启发，Cakir 等[36]引入了错误纠正编码(Error Correcting Output Codes, ECOC)来代表每一个新的标签。

哈希函数可以被用来训练为空间里的分类器，生成的二进制哈希码则是指示分类的超平面向量，由错误纠正编码来表示。码本(Codebook)$\boldsymbol{C}$ 是由 1 和 −1 两种元素组成的矩阵，其中每一列向量称为码字(Codeword)分别代表了一个虚拟类别，同时正交的行向量就代表了类别所处的虚拟区域。假设新标签的数量是未知的，当带有新标签的数据到来时，将会在码本中为其分配一个新的错误纠正编码来进行监督学习，而不需要对标签数据的任何先验信息。另一方面，带有旧标签的数据则依据先前已分配的错误纠正编码来学习，那么所有具有相同标签的数据则会由相同的错误纠正编码紧凑地聚集在同一个类别里而拥有近似的哈希码。

上述随机梯度下降的在线监督哈希虽然为哈希学习提供了监督信息，但未明确监督的质量。在构造错误纠正编码的码本时用随机的方式使编码向量离散，这并不能完全保证消除其相关性。Lin 等[37]提出编码矩阵应当满足以下要求：最大化每行之间的海明距离，从而具有较强的纠错能力；最大化每行之间的海明距离，确保每个分类器之间保持显著的差异性。

阿达玛矩阵满足以上要求。阿达玛矩阵是一个 n 阶正交矩阵，行向量和列向量都各自正交，其元素为+1 或 −1。高阶的阿达玛矩阵可由低阶矩阵推导生成。当带有新标签的数据样本输入时，将会从阿达玛矩阵中随机且非重复选择列向量构造用来虚拟的表示这个标签。若标签已存在，则给出已分配给相同标签样本的虚拟标签。最终把这些向量进行聚合以构成编码矩阵(如图6所示)。

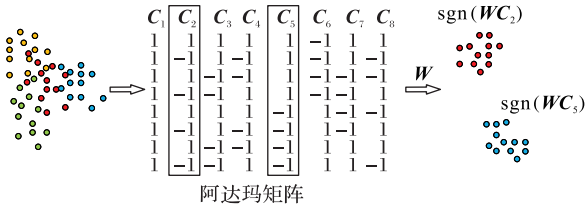


图6 阿达玛矩阵作为码本

Fig. 6 Hadamard matrix used as codebook

值得注意的是，基于阿达玛矩阵的错误纠正编码的码本可以离线生成而且采用哈希桶的位置敏感哈希也同样是数据独立的方法，在查找时可以用近似线性的复杂度进行查找，同时也缓解了在线学习时所带来的不稳定性。在此之后，文献[38]采用了核的方式映射原始数据，并进一步地在多标签数据输入的情况下进行了研究。

2.2.7 小结

综上所述，无监督的在线哈希方式如基于无监督聚类和早期基于矩阵分解的方式无法利用标签信息而检索能力较差，目前大部分在线哈希算法如基于主−被动算法、相似性监督、码本监督、互信息度量等都采用监督学习的方式提高检索精确度，总结如表1所列。基于主−被动算法的方式限制了更新模型时对旧数据可能出现的偏差，但模型不能学习到参数更新的方向。系统不能分辨哪些特征是前所未有的哪些是已经存在的，在保留原有映射函数的同时针对性地优化部分映射，及损失函数中对参数的约束往往使得模型难以优化。基于相似性监督的方法同时优化相似数据和不相似数据之间的距离损失，然而在数据流中无法保证各类数据是独立同分布的，尤其是相似的数据获取困难。要解决这种不平衡问题则必须耗费一些存储空间来存储旧数据，例如平衡相似度在线哈希。互信息哈希也同样面临此类问题，尤其是一些网络通信数据，会定期删除一些历史流量，在处理时效性较短的数据时面临挑战；基于码本监督的算法和互信息在线哈希把哈希码学习转化为分类问题，通过优化分类能够较好地保持相似哈希码之间的紧密度，在明确固定类别的数据上表现较好。码本监督允许数据以单个数据点的更新方式进行分类，更能适应数据流环境下学习哈希码来应对实时检索。难点在于固定长度的码本向量的编码过程是离线的，如阿达玛矩阵引导

的在线哈希,在转化为不同长度的哈希码时产生的误差导致保证哈希码之间离散度的问题,并且如果标签类别改变则要重新生成码本,导致额外的计算开销。此外码本数量也较为固定,不适合处理数据类别有较多增加或减少的在线学习任务,数据的可扩展性较差;部分基于矩阵分解的在线哈希虽然采用了监督学习但未考虑到求解过程中原有数据所内含的语义信息,导致数据尺寸缩小的同时没有学习到有效特征来拟合优化目标。

表 1　在线哈希算法总结

Tab. 1　Summary of online hashing algorithms

方法	主要算法	更新方式	学习模式	读取次数
OH (Online Hashing)[23]等	主-被动算法	数据对	监督学习	单次
HCOH (Hadamard Codebook based Online Hashing)[37]等	码本监督	数据点	监督学习	单次
AH(Adaptive Hashing)[33]	相似性监督	数据对	监督学习	单次
BSODH (Balanced Similarity for Online Discrete Hashing)[34]	相似性监督	数据块	监督学习	多次
MIHash (Online Hashing with Mutual Information)[35]	互信息度量	数据块	监督学习	多次
OSSH (Online Supervised Sketching Hashing)[30]	矩阵分解	数据块	监督学习	单次
OSH (Online Sketching Hashing)[27]等	矩阵分解	数据块	无监督学习	单次
SOH (Online Self-Organizing Hashing)[32]	无监督聚类	数据点	无监督学习	单次

3　未来发展方向

进入到互联网时代线上数据每时每刻呈爆炸式增长,处理这些大规模流数据的任务显得至关重要。目前哈希学习方式引入了不少在线算法,但具体到现实应用仍然有一些相关问题值得被探索:1)流数据的一大特点在于其产生的实时性,而数据个体本身可能是高维且复杂的,比如使用哈希码处理图片分类任务[39],需要先对数据进行预处理压缩或提取特征,如提取尺度不变特征转换(Scale-Invariant Feature Transform)特征[40-41]、梯度直方图(Histogram of Gradient)特征使哈希码获取深层的语义信息。由于深度神经网络的庞大参数量给整个在线更新过程带来很大计算压力而无法完成端到端学习,即遇到数据流中非独立同分布的特征变化学习能力可能会大打折扣。2)数据流的实时性也体现在即时的反馈结果,如电商平台根据用户的商品浏览信息提供有针对性的商品推荐,在改善用户体验的同时也扩大了平台市场潜在的交易量。这就需要模型在用户作出操作行为后快速计算保证结果的时效性。因此算法的复杂度不能太高且收敛速度不能太慢,而现有的在线哈希方式则较少关注这两项指标。3)最近一些方法逐渐开始关注模型更新后对旧数据的检索能力,设置弥补措施防止模型在学习过程中倾向遗忘。但长期存储旧数据的成本较高,往往会删除一些早期数据,因此模型也应考虑学习过程中的时序性:每个时间步的更新优化都会影响到后续时间步的先验概率。不仅如此,数据流有时会出现新类型的数据,这些数据是以往学习过程中没有出现的,模型会面临“概念演化”(Concept Evolution)的问题。例如在金融大数据运营[42]中出现异常交易或非法交易信息系统能及时对这些结构化的数据进行识别,而非误判为原有的合法信息,向系统发出警示信号防止资产流失,类似的也可用于其他的异常检测。因此模型应当能够学习到新数据的增量特征并且保持原有数据的深层次内联关系来提高在整个时间线前后的泛化能力。

就上述观点而言,仍有许多新的技术和方向(如深度学习、强化学习)可以与在线哈希学习进行结合来进一步提高模型的时效性与泛化能力。

4　结语

本文总结了近年来表现较好的几种在线哈希学习方法,这些方法通过权衡模型更新与保持有效检索,使得在大规模数据集上进行在线哈希学习成为可能,相较于原有的离线学习带来了存储空间更低、学习成本更小以及在新数据样本上具有更好适应性等优势。当前大数据的迅猛发展,要求哈希模型能够在数据流中快速学习以应对检索,因此在线学习型哈希在面对复杂且多变的未知数据,进一步提高学习效率,增强模型的实时性和准确性上有着非常广阔的发展前景。

参考文献 (References)

[1] SHAKHNAROVICH G, DARRELL T, INDYK P. Nearest-Neighbor Methods in Learning and Vision: Theory and Practice [M]. Cambridge: MIT Press, 2006: 221-222.

[2] ATHITSOS V, POTAMIAS M, PAPAPETROU P, et al. Nearest neighbor retrieval using distance-based hashing[C]// Proceedings of the IEEE 24th International Conference on Data Engineering. Piscataway: IEEE, 2008: 327-336.

[3] YANG Y, ZHA Z, GAO Y, et al. Exploiting Web images for semantic video indexing via robust sample-specific loss[J]. IEEE Transactions on Multimedia, 2014, 16(6): 1677-1689.

[4] 郦荣. 基于人工智能技术的富媒体信息管控研究[J]. 电信工程技术与标准化, 2017, 30(8): 1-6. (LI R. Artificial intelligence based rich media information monitor schemes [J]. Telecom Engineering Technics and Standardization, 2017, 30(8): 1-6.)

[5] 孙君顶,原芳. 基于内容的图像检索技术[J]. 计算机系统应用, 2011, 20(8): 240-244. (SUN J D, YUAN F. Content-based image retrieval[J]. Computer Systems and Applications, 2011, 20(8): 240-244.)

[6] GUTTMAN A. R-trees: a dynamic index structure for spatial searching [C]// Proceedings of the 1984 ACM SIGMOD International Conference on Management of Data. New York: ACM, 1984: 47-57.

[7] ARYA S, MOUNT D M, NETANYAHU N S, et al. An optimal algorithm for approximate nearest neighbor searching fixed dimensions[J]. Journal of the ACM, 1998, 45(6): 891-923.

[8] WANG J, ZHANG T, SONG J, et al. A survey on learning to hash [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2018, 40(4): 769-790.

[9] WANG J, KUMAR S, CHANG S F. Sequential projection learning for hashing with compact codes [C]// Proceedings of the 27th International Conference on Machine Learning. Madison, WI: Omnipress, 2010: 1127-1134.

[10] GIONIS A, INDYK P, MOTWANI R. Similarity search in high dimensions via hashing[C]// Proceedings of the 25th International Conference on Very Large Data Bases. San Francisco: Morgan Kaufmann Publishers Inc., 1999: 518-529.

[11] KAFAI M, ESHGHI K, BHANU B. Discrete cosine transform locality-sensitive hashes for face retrieval[J]. IEEE Transactions

on Multimedia, 2014, 16(4): 1090-1103.
[12] GONG Y, LAZEBNIK S. Iterative quantization: a procrustean approach to learning binary codes [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2012, 35(12): 2916-2929.
[13] LIU W, WANG J, KUMAR S, et al. Hashing with graphs [C]// Proceedings of the 28th International Conference on Machine Learning. Madison, WI: Omnipress, 2011: 1-8.
[14] KONG W, LI W. Double-bit quantization for hashing [C]// Proceedings of the 26th AAAI Conference on Arti fi cial Intelligence. Palo Alto, CA: AAAI Press, 2012: 634-640.
[15] NOROUZI M, FLEET D J. Minimal loss hashing for compact binary codes [C]// Proceedings of the 28th International Conference on International Conference on Machine Learning. Madison, WI: Omnipress, 2011: 353-360.
[16] LIU W, WANG J, JI R, et al. Supervised hashing with kernels [C]// Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2012: 2074-2581.
[17] WAGSTAFF K, CARDIE C, ROGERS S, et al. Constrained K-means clustering with background knowledge [C]// Proceedings of the 18th International Conference on Machine Learning. San Francisco: Morgan Kaufmann Publishers Inc., 2001: 577-584.
[18] WANG J, KUMAR S, CHANG S F. Semi-supervised hashing for scalable image retrieval [C]// Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2010: 3424-3431.
[19] LAI H, PAN Y, LIU Y, et al. Simultaneous feature learning and hash coding with deep neural networks [C]// Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2015: 3270-3278.
[20] DING K, FAN B, HUO C, et al. Cross-modal hashing via rank-order preserving [J]. IEEE Transactions on Multimedia, 2017, 19(3): 571-585.
[21] ZHANG S, LI J, JIANG M, et al. Scalable discrete supervised multimedia hash learning with clustering [J]. IEEE Transactions on Circuits and Systems for Video Technology, 2018, 28(10): 2716-2729.
[22] CRAMMER K, DEKEL O, KESHET J, et al. Online passive-aggressive algorithms [J]. Journal of Machine Learning Research, 2006, 7: 551-585.
[23] HUANG L, YANG Q, ZHENG W. Online hashing [C]// Proceedings of the 23rd International Joint Conference on Artificial Intelligence. Palo Alto, CA: AAAI Press, 2013: 1422-1428.
[24] HUANG L, YANG Q, ZHENG W. Online hashing [J]. IEEE Transactions on Neural Networks and Learning Systems, 2018, 29(6): 2309-2322.
[25] 钱江波,胡伟,陈华辉,等. 基于学习型哈希的在线近邻查找算法[J]. 控制与决策, 2019, 34(12): 2567-2575. (QIAN J B, HU W, CHEN H H, et al. Online learning to Hash for nearest neighbor search [J]. Control and Decision, 2019, 34(12): 2567-2575.)
[26] WENG Z, ZHU Y. Online hashing with efficient updating of binary codes [C]// Proceedings of the 34th AAAI Conference on Artificial Intelligence. Palo Alto, CA: AAAI Press, 2020: 12354-12361.
[27] LENG C, WU J, CHENG J, et al. Online sketching hashing [C]// Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2015: 2503-2511.
[28] WEISS Y, TORRALBA A, FERGUS R. Spectral hashing [C]// Proceedings of the 21st International Conference on Neural Information Processing Systems. Red Hook, NY: Curran Associates Inc., 2008: 1753-1760.
[29] CHEN X, KING I, LYU M R. FROSH: FasteR online sketching hashing [C]// Proceedings of the 33rd Conference on Uncertainty in Artificial Intelligence. Sydney: AUAI Press, 2017: 1-10.
[30] WENG Z, ZHU Y. Online supervised sketching hashing for large-scale image retrieval [J]. IEEE Access, 2019, 7: 88369-88379.
[31] HE K, WEN F, SUN J. K-means hashing: an affinity-preserving quantization method for learning binary compact codes [C]// Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition. Piscataway: IEEE, 2013: 2938-2945.
[32] CHEN J, LI Y, LU H. Online self-organizing hashing [C]// Proceedings of the 2016 IEEE International Conference on Multimedia and Expo. Piscataway: IEEE, 2016: 1-6.
[33] CAKIR F, SCLAROFF S. Adaptive hashing for fast similarity search [C]// Proceedings of the 2015 IEEE International Conference on Computer Vision. Piscataway: IEEE, 2015: 1044-1052.
[34] LIN M, JI R, LIU H, et al. Towards optimal discrete online hashing with balanced similarity [C]// Proceedings of the 33rd AAAI Conference on Artificial Intelligence. Palo Alto, CA: AAAI Press, 2019: 8722-8729.
[35] CAKIR F, HE K, BARGAL S A, et al. MIHash: online hashing with mutual information [C]// Proceedings of the 2017 IEEE International Conference on Computer Vision. Piscataway: IEEE, 2017: 437-445.
[36] CAKIR F, BARGAL S A, SCLAROFF S. Online supervised hashing [J]. Computer Vision and Image Understanding, 2017, 156: 162-173.
[37] LIN M, JI R, LIU H, et al. Supervised online hashing via Hadamard codebook learning [C]// Proceedings of the 26th ACM International Conference on Multimedia. New York: ACM, 2018: 1635-1643.
[38] LIN M, JI R, LIU H, et al. Hadamard matrix guided online hashing [J]. International Journal of Computer Vision, 2020, 128(8/9): 2279-2306.
[39] LIN K, YANG H F, HSIAO J H, et al. Deep learning of binary hash codes for fast image retrieval [C]// Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition Workshops. Piscataway: IEEE, 2015: 27-35.
[40] LOWE D G. Distinctive image features from scale-invariant key points [J]. International Journal of Computer Vision, 2004, 60(2): 91-110.
[41] MIKOLAJCZYK K, SCHMID C. A performance evaluation of local descriptors [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2005, 27(10): 1615-1630.
[42] 郭佳睿,魏进武,张云勇. 大数据助力运营商提升规模化运营核心力策略[J]. 电信科学, 2018, 34(1): 120-125. (GUO J R, WEI J W, ZHANG Y Y. Strategies for enhancing core capability of large-scale operation for national telecom operators assisted by big data [J]. Telecommunications Science, 2018, 34(1): 120-125.)

This work is partially supported by the National Natural Science Foundation of China (61572266).

GUO Yicun, born in 1995, M. S. candidate. His research interests include data mining, deep learning.

CHEN Huahui, born in 1964, Ph. D., professor. His research interests include database, big data processing.