Pattern Recognition 000 (2017) 1–8

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

Robust discrete code modeling for supervised hashing

Yadan Luo[a], Yang Yang[a,*], Fumin Shen[a], Zi Huang[b], Pan Zhou[c], Heng Tao Shen[a]

[a] School of Computer Science & Engineering, University of Electronic Science and Technology of China, Chengdu, China
[b] The University of Queensland, Brisbane, Australia
[c] Huazhong University of Science and Technology of China, Wuhan, China

ARTICLE INFO

Article history:
Received 31 August 2016
Revised 13 February 2017
Accepted 27 February 2017
Available online xxx

Keywords:
Supervised hashing
Robust modeling
Discrete optimization.

ABSTRACT

Recent years have witnessed the promising efficacy and efficiency of hashing (also known as binary code learning) for retrieving nearest neighbor in large-scale data collections. Particularly, with supervision knowledge (*e.g.*, semantic labels), we may further gain considerable performance boost. Nevertheless, most existing supervised hashing schemes suffer from the following limitations: (1) severe quantization error caused by continuous relaxation of binary codes; (2) disturbance of unreliable codes in subsequent hash function learning; and (3) erroneous guidance derived from imprecise and incomplete semantic labels. In this work, we propose a novel supervised hashing approach, termed as *Robust Discrete Code Modeling* (RDCM), which directly learns high-quality discrete binary codes and hash functions by effectively suppressing the influence of unreliable binary codes and potentially noisily-labeled samples. RDCM employs $\ell_{2,p}$ norm, which is capable of inducing sample-wise sparsity, to jointly perform code selection and noisy sample identification. Moreover, we preserve the discrete constraint in RDCM to eliminate the quantization error. An efficient algorithm is developed to solve the discrete optimization problem. Extensive experiments conducted on various real-life datasets show the superiority of the proposed RDCM approach as compared to several state-of-the-art hashing methods.

1. Introduction

In recent years, driven by the rapid evolution of computer techniques, including fast Internet, massive storage devices, ubiquitous smart phones, ever-increasing volume of multimedia data (*e.g.*, image, audio, video) with high dimensionality have been emerging on the Web. How to precisely and efficiently index and search these data in order to fulfill user needs has posed significant challenges on modern multimedia search engines. While most traditional data indexing techniques (*e.g.*, B^{+}-tree [1]), may easily fail when coping with high-dimensional data due to “curse of dimensionality”, numerous research endeavors have been dedicated to hashing techniques [2–7], which have shown promising effectiveness and efficiency in various real-world applications, ranging from image retrieval [8–10], security protection [11], pattern recognition [12–15] to recommendation [16]. The fundamental goal of hashing is to generate a Hamming space, where high-dimensional data are projected and represented as compact binary codes. The derived binary codes may embody several excellent characteristics: 1) low storage cost for supporting in-memory applications; (2) efficient computation of Hamming distance based on bit-wise XOR operations; and (3) preservation of properties in original space, *e.g.*, structural information [17].

In general, hashing methods can be roughly divided into two groups, *i.e.*, data-independent methods and data-dependent methods. One of classical data-independent hashing, Local-Sensitive Hashing (*LSH*) [18] was introduced based on random projection without exploring data distribution, solving the approximate or exact Near Neighbor Search in high dimensional spaces. However, data-independent methods usually need quite long code length to perform well, thus lacking efficiency.

Compared with data-independent methods, data-dependent hashing, *i.e.*, learning based hashing, can achieve desirable performance with significantly shorter length of code (typically$\leq$200). There are three main categories of data-dependent methods: supervised, semi-supervised and unsupervised hashing. Unsupervised hashing learns the binary codes while taking no accounts of additional semantic labels including [19–23], *etc.* However, through unsupervised learning process, the learnt binary codes hardly preserve adequate semantic information to apply in classification. Supervised hashing methods, as well as semi-supervised methods utilize class labels to generate semantic-preserving hash codes, like Supervised Hashing with Kernels (*KSH*) [24–26], Multiple

* Corresponding author.
E-mail addresses: lyadanluol@gmail.com (Y. Luo), dlyyang@gmail.com (Y. Yang), fumin.shen@gmail.com (F. Shen), huang@itee.uq.edu.au (Z. Huang), panzhou@hust.edu.cn (P. Zhou), shenhengtao@hotmail.com (H.T. Shen).

http://dx.doi.org/10.1016/j.patcog.2017.02.034
0031-3203/

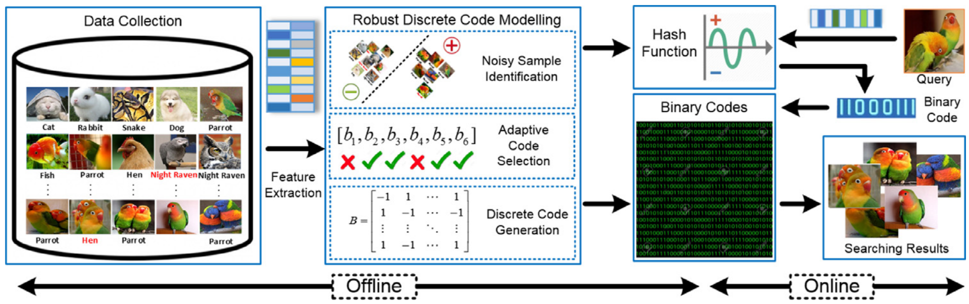

Fig. 1. An illustration of the training phase and retrieval phase involved in RDCM.

feature kernel hashing [27], Multi-task Hashing [28,29], Deep Hashing [30,31].

Typically, most of the existing hashing models are formulated as mixed-integer optimization problems, which are normally intractable to solve due to the discrete constraints on binary codes. In order to simplify the optimization and obtain a feasible solution, a commonly-used way is to first relax the codes to be continuous-valued and make the problem computationally tractable. Then, the generated codes are rounded to binary solution. Such two-stage strategy may easily lead to quantization error, which will be continually accumulated. To handle the above problem, several recent techniques are proposed to achieve discrete hashing with supervision, which can directly produce discrete codes to alleviate quantization error. For example, Shen et al. [32] proposed supervised discrete hashing, which applies discrete cyclic coordinate descent (DCC) [33] algorithm to generate binary codes bit by bit, making a breakthrough towards the binary optimization in the hashing literature. However, the way in SDH to leverage the supervisory information might be non-robust in unreliable and noisy environment.

Indeed, exploiting supervision knowledge and discrete optimization can help to improve retrieval and classification tasks to some extent, the existing methods may still suffer from the following obstacles. On the one hand, the generated codes contain varying degrees of unreliable factors, *e.g.*, redundant and/or imprecise codes, which probably jeopardize the subsequent generation of hash functions. On the other hand, existing labeled data often inevitably contain inaccurate and/or incomplete labels, which are usually caused by the problems of lack of expertise, subjectivity, *etc*. Such erroneous guidance may lead to undependable binary codes.

To address the aforementioned issues, in this paper, we present a novel supervised hashing approach, termed as *Robust Discrete Code Modeling* (RDCM), which jointly considers the generation of discrete codes, as well as suppressing the influence of code noise and label noise in a unified framework by employing $\ell_{2,p}$ norm [34–36] on objective function. Fig. 1 illustrates the overall flowchart of RDCM. We summarize the main contributions as below:

- We propose a novel supervised hashing scheme to generate high-quality hash codes and hash functions for facilitating large-scale multimedia applications. Different from most supervised methods, our method not only alleviates quantization error using discrete optimization but also controls the influence of noise in both the generated binary codes and the given labeled samples.
- We devise an effective binary code modeling approach based on $\ell_{2,p}$ norm, which can adaptively induce sample-wise sparsity, to perform automatic code selection as well as noisy samples identification.
- We preserve the discrete constraint in the proposed model to directly produce discrete codes with minimal quantization error. An efficient algorithm is designed to solve the discrete optimization problem, where a weighted discrete cyclic coordinate decent (WDCC) algorithm is proposed to derive robust binary codes.
- Extensive experiments conducted on various real-world datasets demonstrate the promising results of the RDCM approach in retrieval and classification tasks.

The rest of this paper is organized as follows. In Section 2, the proposed RDCM and the corresponding algorithm for optimization is presented. Extensive experimental results and analysis are reported in Section 3, followed by a conclusion in Section 4.

2. Robust Discrete Code Modeling

In this section, we elaborate the details of the proposed robust discrete code modeling for supervised hashing, including preliminary information, model introduction and an efficient algorithm for optimization.

2.1. Model formulation

Suppose that we have n training samples $X=[x_1,x_2,\ldots,x_n]\in\mathbb{R}^{d\times n}$, where d is the dimension of feature space. Also, we have the label information of X, denoted as $Y=[y_1,y_2,\ldots,y_n]\in\mathbb{R}^{c\times n}$. We aim to learn a set of binary codes $B=[b_1,b_2,\ldots,b_n]\in\{-1,1\}^{L\times n}$, where b_i denotes the resultant binary code of the i-th datum x_i. Recall that three obstacles prevent existing approaches from achieving practical usage. To the end of simultaneously addressing all of them and generating reliable binary codes, we propose the following formulation including 1.The first loss term bridges supervised information between binary codes, with $l_{2,p}$ norm controlling confidence and sparsity. 2. The second penalty is used for avoiding overfitting and balancing noise. 3. The last term models the fitting error of the binary codes B by the continuous embedding $g(X)$. The formulation is

$$\min_{g,W,B}\left\|(Y-W^TB)^T\right\|_{2,p}+\alpha\|W\|_{2,q}+\beta\|B-g(X)\|_F^2,\quad \text{s.t. } B\in\{-1,1\}^{L\times n}, \tag{1}$$

where $W\in\mathbb{R}^{L\times c}$ and $\|\cdot\|_F$ denotes the Frobenius norm of a matrix. $\alpha>0$ and $\beta>0$ are balancing parameters. $g:\mathbb{R}^{d\times 1}\to\mathbb{R}^{L\times 1}$ is a mapping function from a feature space $\mathcal{M}$ to a dimensional space:

$$g(x)=P^T\phi(x), \tag{2}$$

where $P^T\in\mathbb{R}^{L\times m}$ is the transformation matrix. $g(X)=[g(x_1),\ldots,g(x_n)]$. To deal with linearly inseparable data, kernel trick is used to tackle the data to improve the performance. Such a process has been theoretically and empirically shown to be an effectual approach for hash function learning [24]. $\phi:\mathbb{R}^{d\times 1}\to\mathbb{R}^{m\times 1}$ is

Please cite this article as: Y. Luo et al., Robust discrete code modeling for supervised hashing, Pattern Recognition (2017), http://dx.doi.org/10.1016/j.patcog.2017.02.034