Graph Convolutional Network Hashing for Cross-Modal Retrieval

Ruiqing Xu, Chao Li, Junchi Yan, Cheng Deng and Xianglong Liu

摘要

基于深度网络的跨模态检索最近取得了重大进展。然而，弥合模态之间的差距以进一步提高检索的准确性仍然是一个关键的瓶颈。在本文中，我们提出了一种图卷积哈希（Graph Convolutional Hashing, GCH）方法，它通过亲和图学习模态统一的二进制代码。该论文提出一个端到端的模型，其深度架构主要由三个部分组成：一个语义编码器模块、两个特征编码网络和一个图卷积网络（Graph Convolutional Network, GCN）。我们设计了一个语义编码器作为教师模块来指导特征编码过程，以此充分利用数据中的语义信息。此外，该模型还引入了一个 GCN 来探索数据点之间固有的相似性结构，这将有助于产生更具鉴别力的哈希码。在三个大型数据集上进行的大量实验表明，所提出的 GCH 优于目前大多数的跨模态哈希算法。

Keywords: 跨模态检索，哈希学习，图卷积

1 引言

随着互联网的普及，各类高维数据爆发式增长，如何在如此大量的数据中进行高效的跨模态检索成为一个较为热门的研究话题。跨模态检索的目的，是通过使用一种模态（如文本）的查询来搜索另一种模态（如图像）中语义相似的数据点。而通过对数据进行编码，使检索从实值空间转化到汉明空间的哈希学习方法在近邻搜索中受到了一定的关注，因此一些学者开始将跨模态检索与哈希学习进行结合。然而，由不同模态数据的异质性所造成的模态差距仍然是一个夯待解决的问题。

大多数早期提出的跨模式哈希方法[1-4]都是采用浅层结构和手工获取的特征。然而，这些方法有一个主要的缺点，即手工获取的特征提取过程与哈希码学习过程无关，这样子减低了学习出来的哈希码的准确性。

而近期许多学者提出了基于深度网络的跨模态哈希方法[5-9]，这类方法通常使用端到端的训练方式，即用卷积神经网络提取特征，同时学习哈希码。这样子模型可以得到更准确的哈希码。然而这些方法大多直接采用相似性矩阵作为语义约束来生成哈希码，或为图像数据和文本数据设计不同的网络，如使用了双流网络的深度跨模式哈希 (Deep Cross-Modal Hashing, DCMH)[10]。这样设计的模型往往无法充分利用语义相关性来指导哈希码的学习过程。此外，数据点之间的语义结构相似性常常在模型设计过程中被忽视，但其对生成具有良好的语义保持能力的哈希码可能非常有帮助。因此，如何将不同数据点之间的语义相似性和结构相似性纳入哈希学习过程中，是一个具有研究价值的问题。

数据点相互独立是现有机器学习算法的一个核心假设。然而，这一假设对于图数据来说并不成立，在图数据中，每个数据点都通过一些复杂的链接信息与其他数据点相联系，而这些信息可以捕捉到数据点之间的相互依赖关系。同样的直觉也存在于跨模式检索中，因为两种模式中的每个数据对都与其相邻的数据对有联系，采用这种依赖关系对准确的检索有好处。对图结构数据的深度卷积操作，如图

卷积网络（Graph Convolutional Networks，GCN），由于其利用节点间关系的精细能力[11-12]，已经吸引了越来越多的关注。最早关于 GCN 的研究之一是 Bruna 等人在 2013 年提出的[13]，从那时起，许多变形及改进相继被提出，并在节点分类等应用中展现了其优异的性能[14-15]。GCN 的基本思想是通过图的邻接矩阵及邻接点的特征来更新该点的特征，因此 GCN 可以通过邻接关系得到数据点的语义结构。作者认为，将这个优点用于跨模态哈希中有利于学习结构相似保持的哈希码并进一步提升模型的检索性能。

本文所复现的工作提出了一种用于跨模态检索的图卷积哈希（Graph Convolutional Hashing, GCH）[16]，它包括三个主要部分：一个语义编码器、两个特征编码网络和一个基于图卷积网络的融合模块。该方法利用 GCN 获取不同的数据点之间的结构相似性，同时采用语义编码器来指导特征编码过程，这样可以在特征学习的过程中同时保留语义和结构的相似性，从而生成更具辨别力的哈希码。本文工作的贡献如下:

作者提出了一种新颖的基于图卷积网络的跨模态哈希方法，以减小数据间的模态差距，提高模型跨模态检索的能力。

为了充分有效地探索语义信息，作者训练一个语义编码器用来挖掘数据间的语义相关性，它作为”教师模块”，指导特征编码网络学习具有辨别力和语义丰富的特征。然后，GCN 被用来进一步增强具有语义结构的丰富特征，获得一个用于进一步更新编码特征的信标特征。

作者在三个大型数据集上的实验表明，GCH 明显优于目前最先进的跨模式哈希方法，包括传统的和基于深度学习的方法。

2 相关工作

跨模态哈希方法可以分为两种不同的类别：无监督的方法和有监督的方法。其中，集体矩阵因子化哈希方法 (Collective Matrix Factorization Hashing, CMFH)[2]是一种无监督的方法，其通过对来自不同视角的联合矩阵进行因式分解来生成多个模态的统一哈希码。而 CMSSH[1]通过特征分解和提升来保留类内相似性，是一种有监督的跨模态哈希方法。

基于深度网络的模型[8,10,17-18]在近年来受到广泛关注。与那些利用手工特征的方法相比，基于深度的方法可以更好地获得更多的鉴别性特征，这使得这些方法在跨模态检索任务上的性能进一步上升。在最近提出的跨模态汉明哈希（Cross-Modal Hamming Hashing, CMHH）[18]中，作者通过在贝叶斯学习框架中联合优化一个新的外指数焦点损失和指数量化损失生成了有利的哈希代码，并实现了精确检索。此外，与本文复现的方法类似，深度跨模态哈希（Deep Cross-Modal Hashing, DCMH）[10]和自监督对抗哈希（Self-Supervised Adversarial Hashing, SSAH）[8]都是通过保留标签中的相似性关联来学习哈希码，以此充分利用语义信息。这两种方法取得了令人满意的结果，然而，DCMH 简单得使用相似性矩阵来保持相似数据点的语义相关性，而没有过多地关注跨模式数据的潜在结构。另一方面，SSAH 虽然注意到数据点潜在的语义结构，却非常耗时。因此，如何有效地连接不同的模态，并在有监督的信息下探索模态的相关性以产生有利的哈希码，是提高跨模态哈希的搜索精度的关键。

与现有的基于深度网络的跨模态哈希方法不同，GCH 采用 GCN 挖掘数据的结构信息，并利用语义编码器从不同模态中提取语义信息，将其转移到编码特征中，从而很好地保留了语义和结构的相似

性，最终得到较有效的散列码以及更好的检索性能。

3 本文方法

在这项工作中，我们专注于图像和文本模态之间的跨模态检索，这是日常生活中最常用的两种模态。图 1显示了 GCH 模型的流程图。它由三个主要部分组成：一个语义编码器，两个的特征编码网络(一个用于图像数据的编码，一个用于文本数据的编码)，以及一个基于图卷积网络的融合模块，这些都将在下文中进行具体的介绍。

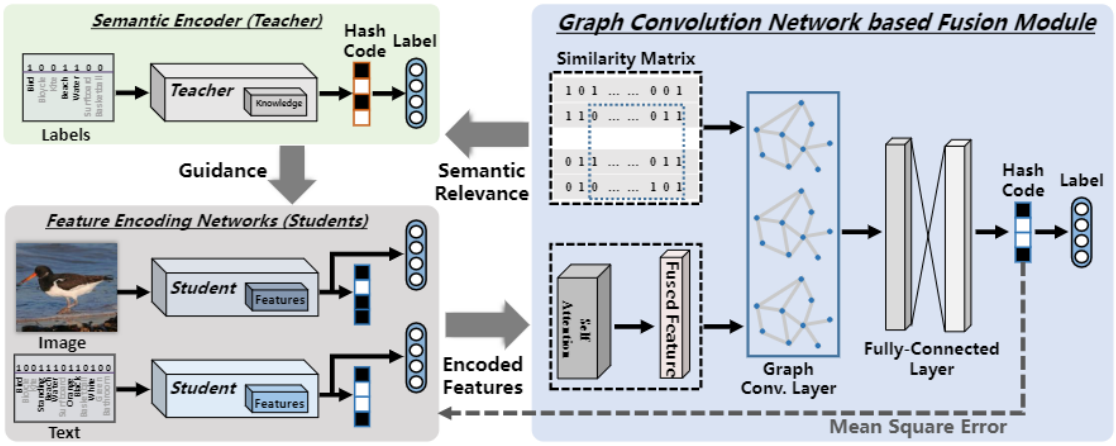

图 1: GCH 流程图

3.1 问题描述

给定一个有 n 个数据点的跨模态数据集 $O = \{o_i\}_{i=1}^n$，其中 $o_i = (x_i, y_i, l_i)$，x_i 和 y_i 分别代表第 i 个数据点的原始图像和文本，$l_i = [l_{i1}, ..., l_{ic}]$ 是 o_i 的标签，其中 c 是类别数。具体来说，我们利用多标签相似性矩阵 S 来表示两个数据点 m, n 的相似性：如果它们至少属于一个相同的类别，$S_{mn} = 1$，否则 $S_{mn} = 0$。跨模态哈希的目标就是为各个模态学出统一的 K 位哈希码 $B \in \{-1, 1\}^K$，并同时保留数据对之间的原始相似性。此外，为了衡量两个哈希码 h_i 和 h_j 之间的相似性，我们可以用它们的内积 $< h_i, h_j >$ 来计算它们的汉明距离。为了获得任何一种模式的哈希码，我们可以简单地对编码的特征进行非线性变换，如下式所示：

$$H^* = h(f^*). \tag{1}$$

其中，$* = \{x, y, l\}$，$h(\cdot)$ 为非线性变化，H^* 为我们得到的哈希码。

3.2 语义编码器

为了发现标签中丰富的语义信息，并将这些信息转移到编码特征中，在“教师-学生”策略的启发下，作者构建了一个新颖的语义编码器作为教师模块来充分挖掘标签中的语义信息，并利用学习到的信息来指导特征编码过程。该语义编码器可以表述如下：

$$g^l = G^l(l, \theta^l). \tag{2}$$

其中，θ^l 是网络参数。具体来说，该语义编码器是一个端到端的全连接深度神经网络。作者希望语义编码器 G^l 能够很好地保留语义特征和相应的哈希码之间的相似性，为此，他们将语义编码器的目标函数设计为如下形式：

$$\min_{\theta^l} \mathcal{L}^l = -\alpha \sum_{i,j=1}^{n} (S_{ij} \Gamma_{ij}^l - \log(1 + e^{\Gamma_{ij}^l})) + \beta \|\hat{L}^l - L\|_F^2. \tag{3}$$

其中，$\Gamma_{ij}^{l} = \frac{1}{2}(H_i^l)(H_j^l)^\top$，$\alpha$ 和 β 是超参数。H_{*i}^{l} 是由特征 g^l 转化而来的由模型预测出的哈希码，$\hat{L}^l$ 是同样由特征转化得来的预测标签。在公式(3)中，第一项是负对数似然函数，用来保持特征之间的相似性，第二项是原始标签 L 和预测标签 $\hat{L}^l$ 之间的分类损失。语义编码器的输出对指导特征编码网络学习丰富的语义特征很有帮助，这有利于给两种模态的数据学习出优质的哈希码。

3.3 特征编码器

为了建立不同模态之间的相关性，并进一步学习可靠的哈希码，作者构建了两个特征编码网络，并在语义编码器的指导下将跨模态数据编码为共同的表示。对于第 i 个数据点 o_i，模型用卷积神经网络对图像特征的编码函数 $E^x(x,{}^x)$ 进行建模，以提取高水平的图像特征 f^x，并用四个全连接层构建文本特征编码网络 $E^y(y,{}^y)$。其中 ^{x} 和 ^{y} 是网络参数。特征编码网络可以写成：

$$f^* = E^*(*,\theta^*), * = \{x, y\}. \tag{4}$$

此外，我们希望在特征编码过程中保留从标签中提炼的信息，即语义相关性。因此，两个特征编码过程都必须在语义编码器的指导下完成端到端的训练。为了引入语义编码器的监督，与公式(3)类似，特征编码器的目标函数采用以下形式：

$$\min_{\theta^*} \mathcal{L}^* = \alpha\mathcal{J}_1 + \beta\mathcal{J}_2 + \gamma\mathcal{J}_3 = -\alpha\sum_{i,j=1}^{n}(S_{ij}\Gamma_{ij}^{l*} - \log(1+e^{\Gamma_{ij}^{l*}})) + \beta\|H^b - H^*\|_F^2 + \gamma\|\hat{L}^* - L\|_F^2. \tag{5}$$

与公式(3)中定义的类似，$\Gamma_{ij}^{l*} = \frac{1}{2}(H_i^l)(H_j^*)^\top$，$\alpha$、$\beta$ 和 γ 是超参数。H^* 表示两种模态得到的预测哈希码，而 $\hat{L}^*$ 是特征编码器得到预测标签。H_b 是由基于 GCN 的混合模块生成的信标哈希码，它有利于特征编码器学习出包含更多语义信息的特征，该哈希码将在后面的章节中讨论。值得注意的是，在公式(5)中，我们利用 Γ_{ij}^{l*} 中的 H^l，使得特征编码器能在语义编码器的指导下对每种模态数据的特征进行编码。这样一来，从语义编码器中获取的语义相关性在两种模态的编码特征中得到了很好的保留。与公式(3)类似，该目标函数通过减少 $\hat{L}^*$ 和原始标签 L 之间的差异来保持分类信息，即 $\mathcal{J}_3$。

3.4 基于图卷积的融合模块

为了进一步保持数据间的结构相似性，作者在模型中用到了图卷积。而在将特征输入图卷积网络前，我们首先需要在不损失太多语义关系的情况下融合前面的编码特征。受[19]的启发，作者选择了自注意力机制作为语义保留的融合方法。具体来说，两个模态的特征使用相对模态的特征进行重新加权，如式(6)所示：

$$f_r = \frac{1}{2}(f_{s-a}^x + f_{s-a}^y), f_{s-a}^x = f^x \times \widetilde{W}, f_{s-a}^y = f^y \times \widetilde{W}. \tag{6}$$

其中，$\widetilde{W} = norm(f^x \times f^{y\top})$ 作为归一化权重矩阵，f^x 和 f^y 是来自不同模态的特征编码器得到的原始特征，而 f_{s-a}^x 和 f_{s-a}^y 是经过处理的特征，f_r 是融合后的特征。至于操作符号，$\times$ 表示矩阵内积，norm 指矩阵归一化。

在这个融合模块的基础上制定出的图卷积网络如下：给定 N_b 对用于训练的数据点 $\{x_i, y_i, l_i\}_{i=1}^{N_b}$，将相应的数据点送入各自的特征编码网络后，我们将得到两个特征矩阵 $f_x \in \mathbb{R}^{N_b \times d}$ 和 $f_y \in \mathbb{R}^{N_b \times d}$，并利用如式(6)的自注意力机制进行融合。在得到语义丰富的融合特征 f_r 后，作者希望进一步保留数据间的结构相似性，以减小模态差距。为此，该模型采用了多层 GCN，在参数迭代更新的过程中，拥有大量潜在结构关系的特征将相互作用，这样可以得到统一了两种模态的哈希码，最终提高跨模态检索

精度。

表示了 N_b 个数据对间相邻关系的邻接矩阵 $A \in \mathbb{R}^{N_b \times N_b}$ 与融合后的特征 f_r 一起被送入多层 GCN 中进行训练。根据[20]的建议，多层 GCN 的逐层传播规则采取以下形式：

$$H^{(l+1)} = \sigma(\tilde{D}^{-\frac{1}{2}}\tilde{A}\tilde{D}^{-\frac{1}{2}}H^{(l)}W^{(l)}). \tag{7}$$

其中，$\tilde{A} = A + I_N$ 是无向图 G 的归一化邻接矩阵，而 $A(i,j) = l_i \times l_j$，这里的 l_i 是第 i 个数据点的真实标签。I_N 是单位矩阵，表示每个节点都与自身相连，$\tilde{D}_{ii} = \sum_j \tilde{A}_{ij}$ 是 $\tilde{D}$ 的度矩阵，$\sigma(\cdot)$ 表示激活函数，如 ReLU。$H^{(l)}$ 和 $H^{(l+1)}$ 分别表示第 l 层的输入和输出特征矩阵，同时也代表了前一层和本层学到的特征。$W^{(l)}$ 的作用类似于传统 CNN 中第 l 层的卷积滤波器，其参数将在训练过程中被更新。

在训练期间更新 GCN 网络中的参数有利于动态更新节点的特征。从公式(7)可知，对于一个节点 V_i，图卷积通过加权求和的方式将 V_i 的邻接节点的特征融合入该节点中，同时根据 $\tilde{A}$ 为 V_i 分配新的特征。这个过程表明该图卷积鼓励邻接节点的特征更加接近。通过这种方式，两种模式的融合特征被结构上的相似性所拉进。因此，该融合模块的目标函数被定义为：

$$\min_{\theta^G} \mathcal{L}^{GCN} = -\alpha \sum_{i,j=1}^{n} (S_{ij}\Gamma_{ij}^G - \log(1 + e^{\Gamma_{ij}^G})) + \beta \|\hat{L}^G - L\|_F^2. \tag{8}$$

其中 $\Gamma_{ij}^G = \frac{1}{2}(H_i^b)(H_j^b)^\top$。将 GCN 的输出作为共同特征空间中的“信标”，使得同类的编码特征相互接近，因此称为信标特征，在公式(5)和公式(8)中表示为 H_b。通过这种方式，结构上的相似性在特征编码过程中中得到了很好的保留。公式(8)中的其余参数与公式(3)中参数的定义类似，在这里不再进行赘述。

3.5 训练策略

结合上述四个目标函数，GCH 最终的目标函数可以表述为：

$$\mathcal{L}^{all} = \mathcal{L}^x + \mathcal{L}^y + \mathcal{L}^l + \mathcal{L}^{GCN}. \tag{9}$$

值得注意的是，正如本文前面所讨论的，作者认为该四个损失具有同等的重要性。目标公式(9)是通过迭代优化学习的。具体来说，算法的优化顺序是 $\mathcal{L}^l \Rightarrow \mathcal{L}^* \Rightarrow \mathcal{L}_G \Rightarrow \mathcal{L}^*$，其中 $* = \{x, y\}$。该网络通过随机梯度下降 (SGD) 和反向传播 (BP) 算法来学习，这在现有的深度方法中被广泛使用。1展示了整个模型训练的细节。

在整个网络以端到端方式得到良好的训练后，通过将原始特征输入特征编码网络，可以直接获得未见过的数据点的哈希码，如公式(10)所示：

$$b_q^{x,y} = sign(f^*(b_q; \theta^*)). \tag{10}$$

其中，$* = \{x, y\}$。

Procedure 1 Graph Convolutional Network Hashing (GCH)。

Input: 图像数据 X，文本数据 Y，标签 L。
Output: 网络参数 $\theta^{x,y,l,G}$。
初始化：网络参数 $\theta^{x,y,l,G}$，超参数 α, β, γ，学习率 μ，批次大小 $N_b^{w,y,l} = 128$，最大迭代次数 T_{max}，$iter = 0$。
while $iter = 1 : T_{max}$ **do**
  用 BP 算法更新 θ^l；
  在语义编码器的指导下更新 $\theta^{x,y}$；
  在语义编码器的指导下更新 θ^G，得到信标特征；
  使用信标特征再次更新 $\theta^{x,y}$。
end

4 复现细节

4.1 创新点

我对该论文进行了两个方面的小尝试，第一个尝试是修改语义编码器对特征编码器的指导方式，具体的体现在修改特征编码器的目标函数；第二个尝试是启发于 []，在数据进行哈希编码前就进行语义对齐，以减少不同模态数据间的差距。

4.1.1 尝试一

从公式(5)中可以看出，作者希望语义编码器得到的哈希码 H^l 与特征编码器得到的哈希码 H^* 能够尽量相似，以此达到在特征编码过程中保留标签中的语义相似性的目的。我们希望特征编码器可以用其他方式对该模块产生影响，因此对特征编码器的损失函数进行修改：

$$\begin{aligned}\min_{\theta^*} \mathcal{L}^* &= \alpha\mathcal{J}_1 + \beta\mathcal{J}_2 + \gamma\mathcal{J}_3 + \lambda\mathcal{J}_4 \\ &= -\alpha\sum_{i,j=1}^{n}(S_{ij}\Gamma_{ij}^* - \log(1+e^{\Gamma_{ij}^*})) + \beta\|H^b - H^*\|_F^2 + \\ &\quad \gamma\|\hat{L}^* - L\|_F^2 + \lambda\sum_{i,j=1}^{n}\|H_i^l - H_j^*\|_2^2 S_{ij}.\end{aligned} \tag{11}$$

其中，$\Gamma_{ij}^* = \frac{1}{2}(H_i^*)(H_j^*)^\top$。第一项是负对数似然损失，用于保持同类数据的相似性，之后我们新增第四项，利用此项引入语义编码器的输出，以此指导特征学习过程。

4.1.2 尝试二

深度统一的跨模态哈希 (Deep Unified Cross-Modality Hashing, DUCMH)[21]在将文本和图像数据输入哈希层前先进行了语义对齐，并输入同一个哈希层中学习统一的哈希码。该论文的思想如图所示：

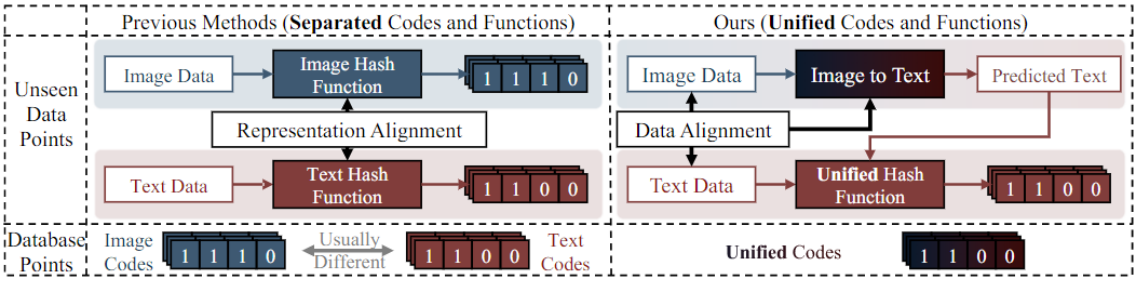

图 2: DUCMH 尝试提前进行语义对齐

受 DUCMH 的启发，我对 GCH 的语义编码器进行修改，将学习出来的图像特征 f^x 与文本特征 f^y 进行对齐，之后输入同一个哈希层中。在语义编码器的损失函数公式(5)中加入以下损失：

$$\mathcal{L}^{i2t} = \|f^x - f^y\|_F^2. \tag{12}$$

该损失用于将学习出的文本特征和图像特征进行对齐，之后将两个网络的特征输入同一个哈希层中进行学习。

4.2 实验环境搭建

运行此代码所需的实验环境如下：

1）Python 2.7；

2）Tensorflow 1.2.0；

3）其他（numpy, scipy, 等）。

4.3 与已有开源代码对比

该工作的代码基于 GCH 论文放在 github 上的源代码（网址为 https://github.com/DeXie0808/GCH）。在此代码的基础上，我们对其进行了 4.1 中所描述的改进。

1）为了实现尝试 1，我们在 GH_itpair.py 文件中修改了损失函数的计算方式，将语义编码器的损失函数改为了如式(11)的形式，其余部分没有进行修改；

2）为了实现尝试 2，我们在 GH_itpair.py 文件中的语义编码器部分增加了式(12)的损失函数，并修改了 tnet.py 文件中图像与文本网络的结构，使其输入同一个哈希层中进行训练。（具体来说是对 img_net_itpair 和 txt_net_itpair 这两个函数进行了修改。）

4.4 代码使用说明

准备好实验环境后运行 main_itpair.py 文件，这时运行的是最原始的 GCH 代码。若需要运行尝试 1 的代码，则需要将 main_itpair.py 文件中的

from GH_itpair.py import GH

改为

from GH_itpair1.py import GH

若想要运行尝试 2 的代码，则改为

from GH_itpair2.py import GH

若需要修改学习的哈希码码长，则修改 setting.py 中的 bit 参数。

5 实验结果分析

本部分将会展示在 MIRFLICKR-25K 数据集上，原始的 GCH 方法及两种改进方法在不同哈希码长下的 MAP 结果。

表 1: MIRFLICKR-25K 数据集上学习 16 位哈希码的实验结果

	GCH	Try1	Try2
I→T	0.700	0.693	0.742
T→I	0.729	0.738	0.726
T→T	0.663	0.665	0.679
I→I	0.783	0.777	0.779

其中，I 表示图像，T 表示文本，而 I→T 则表示使用图像数据查询文本数据；T→I 表示使用文本数据查询图像数据；T→T 表示使用文本数据查询文本数据；I→I 表示使用图像数据查询图像数据。Try1 表示 4.1.1，Try2 表示 4.1.2。

表 2: MIRFLICKR-25K 数据集上学习 32 位哈希码的实验结果

	GCH	Try1	Try2
I→T	0.708	0.726	0.753
T→I	0.772	0.779	0.724
T→T	0.706	0.709	0.687
I→I	0.766	0.788	0.784

1和 2分别是在学习 16 位哈希码及学习 32 位哈希码时的实验结果。总体上看，学习 32 位码长时得到的实验结果普遍比学习 16 位时好，这是可以预想到的，因为当码长变长时，可以包含的信息也会变多，因此准确率会上升。

而第一个尝试在学习 16 位哈希码时在 T→I 和 T→T 的任务上有一定的提升，在学习 32 位哈希码时则完成每一项任务的能力都超过了原始的 GCH，可见修改为这种标签指导方式所学习出来的哈希码具有更优异的性能。

而第二个尝试在学习 16 位哈希码和学习 32 位哈希码时都只有某两项任务的性能有所提升，在 16 位时能更好的进行 I→T 与 T→T 的任务，而在 32 位时能更好的进行 I→T 和 I→I 任务。可见其实第二种尝试在 I→T 任务上的提升十分明显的，甚至超越了尝试一所带来的提升。对此我们查看了部分训练过程，发现在训练集中其他几项任务差不多都达到 0.98 的精度，但到测试集上结果反而变差了，因此可能是过拟合的问题。

6 总结与展望

GCH 的模型是一个较为简单有效的跨模态哈希检索方法，其一个重要的创新点是其引入了图卷积以保持数据的结构相似性。而本工作除了复现该论文，还对其进行了一定的修改提升，一定程度上提高了它的性能。

但这两个尝试都是较为简单的尝试，我认为这两个改进或许可以融合在一起互相补足其缺憾的地方，并且在这次实验中我们没有对图卷积进行任何的修改，但图卷积其实还有许多的变形可以进行尝试，因此这也是未来可能可发展的一个方向。

参考文献

[1] BRONSTEIN M M, BRONSTEIN A M, MICHEL F, et al. Data fusion through cross-modality metric learning using similarity-sensitive hashing[C]//2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. 2010: 3594-3601. DOI: 10.1109/CVPR.2010.5539928.

[2] DING G, GUO Y, ZHOU J. Collective Matrix Factorization Hashing for Multimodal Data[C]// Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2014.

[3] SHEN F, SHEN C, LIU W, et al. Supervised Discrete Hashing[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2015.

[4] DENG C, TANG X, YAN J, et al. Discriminative Dictionary Learning With Common Label Alignment for Cross-Modal Retrieval[J]. IEEE Transactions on Multimedia, 2016, 18(2): 208-218. DOI: 10.1109/TMM.2015.2508146.

[5] YANG E, DENG C, LIU W, et al. Pairwise Relationship Guided Deep Hashing for Cross-Modal Retrieval[J/OL]. Proceedings of the AAAI Conference on Artificial Intelligence, 2017, 31(1). https://ojs.aaai.org/index.php/AAAI/article/view/10719. DOI: 10.1609/aaai.v31i1.10719.

[6] DENG C, CHEN Z, LIU X, et al. Triplet-Based Deep Hashing Network for Cross-Modal Retrieval[J]. IEEE Transactions on Image Processing, 2018, 27(8): 3893-3903. DOI: 10.1109/TIP.2018.2821921.

[7] YANG E, DENG C, LI C, et al. Shared Predictive Cross-Modal Deep Quantization[J]. IEEE Transactions on Neural Networks and Learning Systems, 2018, 29(11): 5292-5303. DOI: 10.1109/TNNLS.2018.2793863.

[8] LI C, DENG C, LI N, et al. Self-Supervised Adversarial Hashing Networks for Cross-Modal Retrieval[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2018.

[9] LI C, DENG C, WANG L, et al. Coupled CycleGAN: Unsupervised Hashing Network for Cross-Modal Retrieval[J/OL]. Proceedings of the AAAI Conference on Artificial Intelligence, 2019, 33(01): 176-183. https://ojs.aaai.org/index.php/AAAI/article/view/3783. DOI: 10.1609/aaai.v33i01.3301176.

[10] JIANG Q Y, LI W J. Deep Cross-Modal Hashing[C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). 2017.

[11] HUANG F, CHEN S. Learning Dynamic Conditional Gaussian Graphical Models[J]. IEEE Transactions on Knowledge and Data Engineering, 2018, 30(4): 703-716. DOI: 10.1109/TKDE.2017.2777462.

[12] YANG X, DENG C, ZHENG F, et al. Deep Spectral Clustering Using Dual Autoencoder Network[C]// Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2019.

[13] BRUNA J, ZAREMBA W, SZLAM A, et al. Spectral Networks and Locally Connected Networks on Graphs[EB/OL]. arXiv. 2013. https://arxiv.org/abs/1312.6203.

[14] DUVENAUD D K, MACLAURIN D, IPARRAGUIRRE J, et al. Convolutional Networks on Graphs for Learning Molecular Fingerprints[C/OL]//CORTES C, LAWRENCE N, LEE D, et al. Advances in Neural Information Processing Systems: vol. 28. Curran Associates, Inc., 2015. https://proceedings.neurips.cc/paper/2015/file/f9be311e65d81a9ad8150a60844bb94c-Paper.pdf.

[15] KIPF T N, WELLING M. Semi-supervised classification with graph convolutional networks[J]. arXiv preprint arXiv:1609.02907, 2016.

[16] XU R, LI C, YAN J, et al. Graph Convolutional Network Hashing for Cross-Modal Retrieval.[C]// Ijcai: vol. 2019. 2019: 982-988.

[17] CAO Y, LONG M, WANG J, et al. Deep Visual-Semantic Hashing for Cross-Modal Retrieval[C/OL]

//KDD '16. San Francisco, California, USA: Association for Computing Machinery, 2016: 1445-1454. https://doi.org/10.1145/2939672.2939812. DOI: 10.1145/2939672.2939812.

[18] CAO Y, LIU B, LONG M, et al. Cross-Modal Hamming Hashing[C]//Proceedings of the European Conference on Computer Vision (ECCV). 2018.

[19] VASWANI A, SHAZEER N, PARMAR N, et al. Attention is All you Need[C/OL]//GUYON I, LUXBURG U V, BENGIO S, et al. Advances in Neural Information Processing Systems: vol. 30. Curran Associates, Inc., 2017. https://proceedings.neurips.cc/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf.

[20] KIPF T N, WELLING M. Semi-Supervised Classification with Graph Convolutional Networks [EB/OL]. arXiv. 2016. https://arxiv.org/abs/1609.02907.

[21] WANG Y, XUE B, CHENG Q, et al. Deep Unified Cross-Modality Hashing by Pairwise Data Alignment. [C]//IJCAI. 2021: 1129-1135.