基于 LSTM 和 CNN 的跨模态信息匹配

周茂林[1], 王复英[2], 时子威[3]

(1.无 57 班 2015011173；2.无 57 班 2015011175；3.无 57 班 2015011166)

摘　要：在本报告中，我们给出了对若干无声视频与音频特征进行相似度度量和匹配的算法，该方法主要基于当前主流的深度神经网络实现，在测试集上达到 86.67%的准确率，实现了较高的匹配精度。

关键词：跨模态匹配，卷积神经网络（CNN），长短时记忆神经网络（LSTM），残差神经网络（Resnet）

1　引言

机器学习是一类从数据中自动分析获得规律，并利用规律对未知数据进行预测的算法。它的应用已遍及人工智能的各个分支，如专家系统、自动推理、自然语言理解、模式识别、计算机视觉、智能机器人等领域。

深度学习是机器学习的分支，是一种试图使用包含复杂结构或由多重非线性变换构成的多个处理层对数据进行高层抽象的算法。在近年的计算机视觉、自然语言处理等领域深度学习展示出了优越的效果和性能。

在本篇文章里，我们将介绍基于深度学习完成跨模态匹配的过程。在第 2 部分将会给出课程任务以及实验要求，在第 3 部分给出组内分工。在第 4 部分将对我们在此次实验中使用的网络原理进行介绍。而紧接着在第 5 部分给出网络的具体实现，第 6、7 部分里给出我们网络的性能分析和待改进的不足之处。

2　课程任务和实验要求[1]

2.1　任务描述

给定若干对无声视频与音频文件，并将其名称和顺序打乱。同学需使用机器学习方法，利用提供的训练视频，提取其视觉、听觉特征， 并设计视听信息之间相似性的度量方法，评估无声视频与音频数据的匹配度，从而找回每个无声视频所对应的原始音频文件。

2.2　任务要求

要求同学利用训练数据集，运用提供的特征提取方法或其改进方法，获取视频的视觉和听觉特征；可以通过修改 models.py 设计自己的网络模型，并通过 train.py 训练网络。 建议同学使用 PyTorch 训练网络模型，对输入的无声视频和音频文件进行匹配，并返回匹配结果，最后利用 evaluate.py 评估算法的性能。

3　小组分工

在小组中，王复英同学主要负责模型的编写和参数的调试，并对设计报告进行最后的整理。周茂林同学对网络结构提出了建设性意见，参与可视化的设计，并且承担了大部分报告编写任务。时子威同学负责网络和参数的调试与优化，并且对数据文件进行整理和总结。

4　网络设计原理

本小组使用了深度学习中的两种常用的神经网络——CNN 和 LSTM，同时结合了残差神经网络、批标准化和 Dropout 等方法，对网络模型进行了设计与调试。

4.1　循环神经网络 RNN 及 LSTM

在传统 RNN 中，随着输入信息的时间距离的增大，会发生梯度消失或梯度爆炸的问题，无法实现对长期记忆的处理。

LSTM 是专为处理长时记忆而设计的，其主要改进是引进了乘法输入单元和乘法输出单元[2]。一个 LSTM Cell 中被放置了三扇门，分别叫做输入门、遗忘门和输出门。

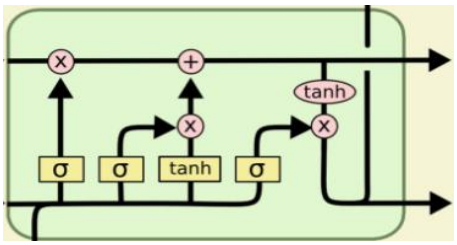

图 1 LSTM 原理示意图

当信息进入 LSTM 的网络当中，网络会判断信息是否有用。只有符合条件的信息才会留下，不符的信息则通过遗忘门被遗忘。

4.2 卷积神经网络

卷积神经网络是一种前馈神经网络，它的神经元可以响应一部分覆盖范围内的周围单元，对于大型图像处理有出色表现。

一般地，CNN 的基本结构包括两层，其一为卷积层，即对图像和滤波矩阵（一组固定的权重：因为每个神经元的多个权重固定，所以又可以看作一个恒定的滤波器 filter）做内积（逐个元素相乘再求和）的操作就是所谓的“卷积”操作，也是卷积神经网络的名字来源。

卷积操作之后一般接一个池化层，用于筛选特征从而进行下一步的操作。此外往往还需要激活函数便于梯度传播和计算。

通过 CNN 可以对特征进行更深度的提取和聚集，通过全连接可以获得更好的分类效果。

4.3 深度残差网络

传统的模型中往往会出现深层网络的性能不如浅层网络的情况，因此为了处理更深的网络我们引入了残差神经网络。

残差神经网络的基本思想是将隐藏层表示为：H（x）=F（x）+ x，其原理图示意如下：

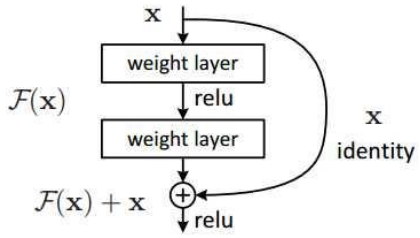

图 2 残差结构原理图

残差单元的输出由多个 LSTM（或 CNN）级联的输出和输入元素相加(保证输出和输入元素维度相同)，再经过 Relu 激活后得到[3]。将这种结构级联起来，就得到了残差网络。

5 网络模型实现

本小节先对我们采用的网络结构做一个说明，此外还介绍了几个在我们的网络训练中较为重要的关键结构。

5.1 网络结构

我们的网络使用 LSTM 网络，结合了卷积层与全连接层，并在残差结构发挥功用。整体方案如下：

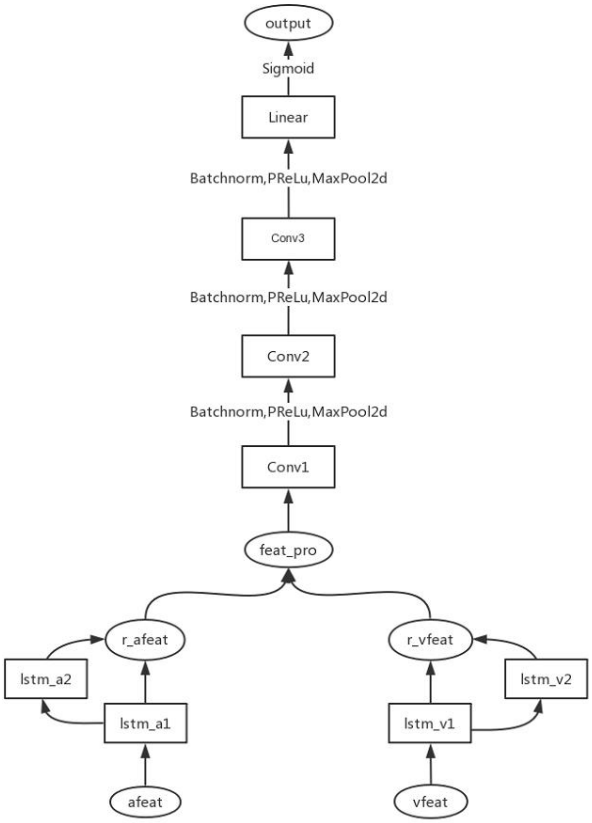

图 3 网络结构框图

首先视频特征和音频特征分别经过 LSTM 结构，这里使用残差结构获得更好的效果；然后将这两种特征在第二个维度连接起来，通过三个 CNN 结构；最后该特征经过全连接层映射和 Sigmoid 激活函数，作为网络的输出。

在训练过程中我们采用 BCELoss(二分类的交叉熵)获得损失并反向传播来优化网络参数。另外优化器我们采用 Adam 实现。

5.2 参数初始化

好的初始化往往能产生测试精度更好的网络，且训练速度也有一定提高。当与每层相关的雅可比矩阵的奇异值远离 1 时，训练可能会更困难[4]。基于这些考虑，我们使用 xavier_normal 初始化，在本模型中起到了较好的效果。

5.3 Dropout

具有大量参数的深度神经网络是非常强大的机器学习系统。然而，过度拟合在这样的网络中是一个严重的问题。Dropout 是解决这个问题的函数。

在训练期间，使用伯努利分布的样本以概率 p 随机地将输入张量的一些元素归零。 每个 forward 都随机分配零元素[5]，以降低过拟合，使网络获得更好的泛化性能。

5.4 Batch Normalization

训练深层神经网络的复杂性在于，随着前一层参数的变化，每层输入的分布在训练过程中发生变化。由于要求较低的学习速度和仔细的参数初始化，从而减慢了训练速度，并且使得非线性饱和的训练模型变得非常困难。我们把这个现象称为内部的 Ovariate Shift，并通过标准化层（BatchNorm）输入来解决问题。BatchNorm 的公式如上：

$$y = \frac{x - mean[x]}{\sqrt{Var[x] + \epsilon}} * gamma + beta$$

批量标准化使我们能够使用更高的学习率，并且不需要太多的初始化操作，并且在某些情况下，无需使用压差。应用到最先进的图像分类模型中，批量标准化达到相同的准确度，训练步骤减少了 14 倍，并且大幅度地优于原始模型[6]。

5.5 Tensorboard

为了更方便 Pytorch 程序的理解、调试与优化，我们使用在程序中加入了 Tensorboard 模块，可以记录与展示标量、图片、计算图、数据分布、直方图、嵌入向量等数据形式，获得更高的模型调试效率。

具体在我们的网络中，我们对 Loss、网络每一层的权重、梯度和输出做了可视化，通过观察数据直方图分布的变化， 实现了更直观便捷的参数调试和模型优化。

6 性能分析

6.1 视频匹配准确度

我们提交的模型(LSTM+CNN)经过训练后，在提供的测试集中，正确率最高可达 86.7%。

另外这里还给出了我们之前尝试的几种模型的最高准确率，在本问题中 LSTM 对特征的提取和处理居于很好的性能，而卷积则为网络性能的进一步提高提供了可能。

当然参数和结构设计在本问题中也起到了关键作用，具体需要经过以后进一步探索。

模型结构	准确率
CNN	70%
LSTM	63.3%
LSTM+时间加权	83.3%
LSTM+CNN	86.7%

6.2 基于 Tensorboard 的可视化分析

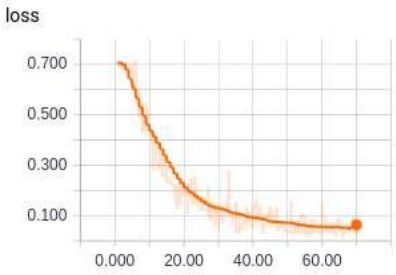

图 4 loss 随迭代的变化

通过此图可以看到在训练中，随着迭代次数的增加，loss 能够稳步的下降，这说明我们的模型训练是行之有效的。

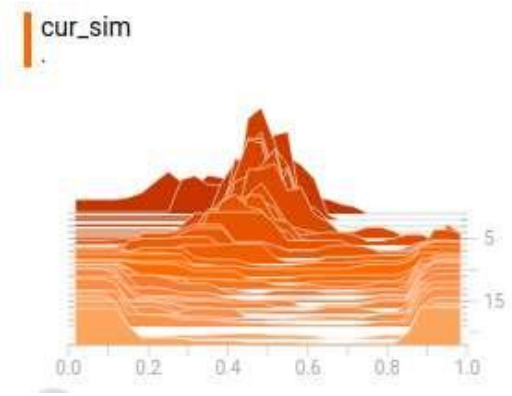

图 5 相似矩阵随迭代的变化

图 5 是一个三维图，纵坐标代表迭代次数，整个图片代表的是一个随迭代次数变化的相似矩阵直方图集合，可以看到经过多次迭代，输出结果向 0 和 1 两个中心聚集，复合我们的预期。

6.3 相似矩阵热点图

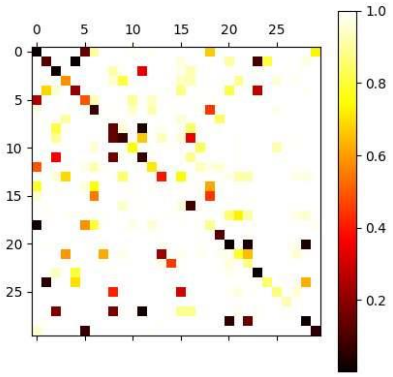

图 6 相似矩阵热点图

将测试所得相似矩阵画成热点图，基本可以看出矩阵中的一条明显的对角线，符合我们的预期。

6.4 网络参数量分析

基于训练网络的时间开销的考虑，我们估计了网络各层的参数，并对其时间效率做大致分析。

结构	参数量
lstm_v1	27~28 万
lstm_v2	3~4 万
lstm_a1	4~5 万
lstm_a2	3~4 万
conv1_1	100
conv2_1	800
conv3_1	3200
Linear	60~70

由此发现，LSTM 占用参数最多，因此考虑时间开销和性能的折中，后续改进应该在准确率可以接受的范围内削减 LSTM 的数量。

7 存在的问题和不足

7.1 整体网络结构

本次实验中，我们使用结合 LSTM 和 CNN 结合残差结构的模型，取得了不错的效果。但是因为时间和能力有限，我们没有继续将残差网络和 CNN 卷积网络结合起来，网络潜力有待进一步挖掘。

7.2 音频和视频的匹配失误

虽然正确率达到了 86.7%，但是用热点图观测在测试代码中输出相似矩阵时，可以看到一些视频和音频的对应相似度存在明显误判，也说明此网络相似度度量的一定不足。

7.3 特征提取有待改良

在多次的网络测试过程中，我们注意到一些音频和视频对很难正确匹配，如测试集的第 28 对视频与音频。该现象启示我们现有的特征可能存在不足，但是由于时间问题，我们在此次实验中没有对这一问题进行展开处理。期待后续能进一步探索。

8 实验总结与展望

8.1 实验总结

本次实验是我们本科阶段对于机器学习的一次启蒙性实验，通过此实验我们接触并使用了当前机器学习领域的一些流行的网络以及算法，并通过实际操作掌握了网络搭建和优化、参数调整等基础的机器学习方法，收获了研究能力和编程能力的提升。

8.2 未来工作展望

在此次实验中，我们发现给定的视频特征和音频特征并不理想。视频与视频之间，音频和音频之间的距离都没有拉开，这限制了我们网络的训练效果。要进一步得到更优秀的网络，还需要对视频和音频做更好的特征提取。

此外，本次实验过程中，我们发现文献的阅读和研究对我们实际的工作起着指导性的作用。除去文章内提到的方法，还有很多有意思的相似工作的算法我们并没有实现，在今后可以更多的尝试学习其他工作者的方法并将其运用到我们的工作中来。

9 致谢

在本报告的最后，感谢陈建生老师及各位助教对我们的悉心指导，还有许多帮助过我们的人，在此一并表示感谢。

参考文献

[1] 视听信息的跨模态匹配 2017-2018 学年度秋季学期《视听信息系统导论》课程设计

[2] Long-Short Term Memory, Sepp Hochreiter.

[3] Deep Residual Learning for Image Recognition, Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun

[4] Understanding the difficulty of training deep feedforward neural networks. Xavier Glorot Yoshua Bengio, DIRO,

[5] Dropout: A Simple Way to Prevent Neural Networks from Overfitting, Nitish Srivastava , Geoffrey Hinton

[6] Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift Sergey Ioffe , Christian Szegedy

附录：文件清单

文件	说明
dataset.py	原 dataset.py 文件，未更改
evaluate.py	新的测试文件
models.py	新的模型文件
train.py	新的训练文件
train.sh	运行 train.py 的命令