

基于深度神经网络的多模态特征自适应聚类方法

敬明旻

(中国石化石油工程技术研究院信息与标准化研究所 北京 100101)

(中国社会科学院数量经济与技术经济研究所 北京 100732)

摘要 深度神经网络学习单模态特征的能力较强,而学习多模态特征的难度较大。针对这种情况,提出基于深度神经网络的多模态特征自适应分类算法。在训练阶段以预定的概率从每个模态选择部分代表信息进行融合,并且建模模式间的相关性。设计基于聚类的特征剪枝策略,减小网络的计算复杂度,通过正则化防止学习程序过拟合。利用动态编码搜索对超参数进行自适应地优化处理。采用自编码器和卷积神经网络分别对该算法进行验证,实现了较好的多模态特征分类效果,并且对数据缺失具有鲁棒性。

关键词 深度神经网络 多模态特征融合 模式识别 卷积神经网络 动态编码搜索

中图分类号 TP391

文献标志码 A

DOI:10.3969/j.issn.1000-386x.2020.10.042

MULTI-MODAL FEATURE ADAPTIVE CLUSTERING METHOD BASED ON DEEP NEURAL NETWORKS

Jing Mingmin

(Research Department of Information and Standardization, Research Institute of Petroleum Engineering, Beijing 100101, China)

(Institute of Quantitative and Technical Economics, Chinese Academy of Social Sciences, Beijing 100732, China)

Abstract Deep neural network has strong ability to learn single model features, but it is difficult to learn multi-model features. In view of this, we propose a multi-model features adaptive classification algorithm based on the deep neural networks. It selected part of information from each model with a predefined probability in the training phase to fuse the selected information and modeled the correlation of models. We also designed a feature pruning strategy based on clustering to reduce the computational complexity of networks, and avoid learning procedure over-fitting by regulation. It adopted dynamic coding searching to optimize the hyper-parameters adaptively. Finally, we validated our algorithm on both autoencoders and convolutional neural networks. It achieves a good multi-model classification effect, and is robust to data loss.

Keywords Deep neural networks Multi-modal features fusion Pattern recognition Convolutional neural networks Dynamic coding searching

0 引言

深度神经网络对各种单模态数据表现出极强的特征学习能力,目前已在许多领域内均取得了令人瞩目的成果,包括视频识别^[1]、语音识别^[2]和心电图识别^[3]等。在实际应用中存在大量多模态特征的数据,例如:多媒体的音频数据和视频数据,医学领域的多个

生物信号监测数据等。这些数据不同模态的特征之间存在相关性,将多模态特征融合,能够有效地提高总体的分类效果。虽然自编码器、卷积神经网络、对抗网络等深度神经网络在单模态特征学习方面表现出较强的能力,但是将多个模态的特征融合是深度神经网络的一个难点。

文献[4]将新闻文字和视频特征融合,以文字内容为支配特征,以语音特征和视频特征为辅助特征,在

提取相同底层特征的情况下,该分类算法实现了性能提升。文献[5]根据人脸和姓名的共现关系,分析文本姓名位置对姓名重要程度的贡献,使用反向传播神经网络融合上述信息,实现了较高的新闻人脸识别率,对于噪声也具有鲁棒性。除了新闻领域,生物应用中不同生物信号之间也存在极大的相关性,因此多模态特征融合技术在生物领域也具有广阔的应用前景。文献[6]将人脸、指纹等多个生物特征融合,用以提高身份验证的安全性。近期研究人员将多特征融合引入推荐系统,也成功提高了推荐系统的性能^[7-8]。

多特征融合技术的最大优势是在仅提取底层特征的情况下,即可有效地提升分类算法的性能^[9]。深度神经网络广泛应用于模式识别领域,而深度神经网络的超参数繁多,且对不同数据所提取的特征形式也存在差异,导致将深度神经网络学习的多模态特征集融合成为一个难题。为了解决上述问题,本文提出兼容不同深度神经网络模型的多模态特征融合方案。

1 方法设计

本文方法主要由扩展层和融合层两部分构成,如图1所示。

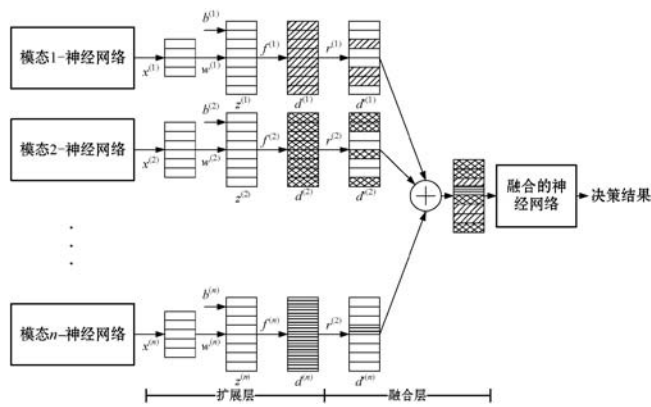


图1 本文方法总体结构

1.1 扩展层

不同网络的输出向量输入扩展层,神经网络可以是任意类型的深度学习网络。不同模态的特征格式不同,特征向量大小也可能不同,因此在扩展层首先将特征向量转化为相同长度。

假设共有 m 个模态网络,设 $\mathbf{x}^{(k)}$ 为第 k 个网络的输出向量,其中 $k \in \{1, 2, \dots, m\}$ 。第 k 个扩展层输入向量为:

$$\mathbf{z}^{(k)} = \mathbf{w}^{(k)} \cdot \mathbf{x}^{(k)} + \mathbf{b}^{(k)} \quad (1)$$

式中: $\mathbf{w}^{(k)}$ 和 $\mathbf{b}^{(k)}$ 分别为权重向量和偏差向量。运用激活函数 f_a 处理 $z_i^{(k)}$, $z_i^{(k)}$ 为 $\mathbf{z}^{(k)}$ 的第 i 个元素,获得第 k 个扩展层第 i 个节点的输出:

$$d_i^{(k)} = f_a(z_i^{(k)}) \quad (2)$$

$\mathbf{d}^{(k)} = [d_1^{(k)}, d_2^{(k)}, \dots, d_c^{(k)}]^T$, 扩展层输出向量的维度均为 c , 即 $i \in \{1, 2, \dots, c\}$ 。

1.2 融合层

融合层收到扩展层的 m 个向量,每个向量的元素数量为 c 。融合层将这些向量重组成一个长度为 c 的向量,称为融合向量。采用多项式采样技术融合多个模态的向量,设多项式采样的向量为 $\mathbf{r}_i = [r_i^{(1)}, r_i^{(2)}, \dots, r_i^{(m)}]^T$, 使其满足以下的多项式分布:

$$\mathbf{r}_i \sim Mn(1, \mathbf{p}) \quad (3)$$

式中: $\mathbf{p} = [p_1, p_2, \dots, p_m]^T$, $\sum_k p_k = 1$, 该条件保证 \mathbf{r}_i 中存在唯一元素等于 1, 而其他值等于 0。运用向量 $\mathbf{r}^{(k)} = [r_1^{(k)}, r_2^{(k)}, \dots, r_c^{(k)}]$ 形式处理 $\mathbf{d}^{(k)}$, 可得:

$$\mathbf{d}'^{(k)} = [d'_1^{(k)}, d'_2^{(k)}, \dots, d'_c^{(k)}]^T = \mathbf{r}^{(k)} \circ \mathbf{d}^{(k)} \quad (4)$$

式中:“ \circ ”表示 Hadamard 积。最终,融合层输出向量 $\mathbf{e} = [e_1, e_2, \dots, e_c]^T$ 的第 i 个元素表示为:

$$e_i = \sum_k d'_i^{(k)} \quad (5)$$

\mathbf{e} 的长度和 $\mathbf{d}^{(k)}$ 的长度相等,将输出结果输入终端网络,终端网络的输出则是分类问题的最终决策。

上述程序确保一个模态数据对应融合向量 \mathbf{e}_i 的一个元素。在图1的实例中,如果 $\mathbf{r}_4 = [1, 0, \dots, 0]^T$, 那么 $e_4 = d'_4^{(1)}$ 。对 \mathbf{e} 的每个元素独立运行模态选择程序,所以最终的融合向量集成了多模态信息。模态选择程序和 \mathbf{p} 的概率值不相关,采用 $\mathbf{p} = [1/m, 1/m, \dots, 1/m]^T$ 为各个模态分配相等的被选择机会。

1.3 模态间的相关性分析

在扩展层考虑了模态特征间的相关性。每个模态的扩展层独立产生 $\mathbf{d}^{(k)}$, 然后筛选出一部分向量输出到融合层。每次训练中多项式采样技术随机采样每个模态的特征,因此对于相同的输入数据,融合的特征向量应该相似。该机制保证模态的扩展层既学习了本模态的数据,也学习了其他模态的特征。

1.4 处理缺失的数据

在实际应用中可能发生一部分模态数据缺失的情况,例如:某个无线传感器可能断开连接,导致一段时间内该模态数据缺失。融合层通过调节概率 \mathbf{p} 能够实现对缺失数据的鲁棒性。设 $\mathbf{u} = [u_1, u_2, \dots, u_m]^T$ 为一个二值向量,表示对应模态数据是否存在:

$$u_k = \begin{cases} 1 & \text{存在 } \mathbf{x}^{(k)} \\ 0 & \text{其他} \end{cases} \quad (6)$$

将 \mathbf{p} 修改为 $\mathbf{p}' = [p'_1, p'_2, \dots, p'_m]^T$ 来调节 s_i 的多项式分布,向量的元素转化为:

$$p'_k = \frac{u_k p_k}{\sum_j u_j p_j} \quad (7)$$

式中： u_j 是对应模态 j 是否存在的二值变量； p_j 为模态 j 所占的概率。如果突然缺失第 k 个模态的数据，那么 u_k 值变为 0， p'_k 也变为 0，由此可防止 $r_i^{(k)}$ 值变为 1，避免第 k 个网络模型的数据向前传播。

1.5 正则化分析

对于第 k 个网络，如果 $r_i^{(k)} = 1$ ，则选择第 k 个扩展层向下一层传播， $r_i^{(k)} = 1$ 的概率为 p_k ，所以第 k 个展开层的运算满足二项分布：

$$r_i^{(k)} \sim B_i(1, p_k) \quad (8)$$

式中： B_i 表示二项式分布； p_k 为传到下一层的成功率。

随机变量的二项分布等价于贝努利分布，因此这个过程等价于对神经网络的随机失活 (dropout)^[10] 处理，输出向量每个值的失活概率为 $(1 - p_k)$ ，基于失活的正则化处理可防止过拟合。虽然结合层的设计初衷是选择合适的模态进行融合，但也能够防止训练阶段的学习过程出现过拟合的情况。

2 多神经网络的剪枝方法

通过剪枝冗余的特征减少最终深度神经网络模型的特征量，缩小网络规模。

2.1 卷积特征聚类 and 剪枝

DNN 包括输入层、输出层和若干的中间层。输入 l 层的通道数量、高度和宽度分别表示为 n'_l, h_l 和 v_l ， l 层的输入和输出分别表示为 Z_l 和 Z_{l+1} 。卷积层卷积 Z_l 和 n'_{l+1} 的三维滤波 $\chi \in \mathbf{R}^{n'_{l+1} \times k \times k}$ ，结果为 n'_{l+1} 的输出特征图 Z_{l+1} 。每个三维滤波由 n'_l 个二维核 $\zeta \in k \times k$ 组成，将全部特征构建成一个核矩阵 $\mathbf{W}^{(l)}$ ，设 $w_i^{(l)}$ 表示网络第 l 层的第 i 个特征， $w_i^{(l)}$ 对应 $\mathbf{W}^{(l)}$ 的第 i 列元素。

本文提出了卷积特征的聚合和剪枝算法，目标是从 n'_l 个原始特征中选出 n_r 个代表特征。将每个特征向量 w_i 初始化为一个类，再根据两个向量的相似性将相似的向量合并，采用平均互信息度量向量的相似性：

$$\overline{SIM}_C(C_a, C_b) = \frac{\sum_{\phi_i \in C_a, \phi_j \in C_b} SIM_C(\phi_i, \phi_j)}{|C_a| \times |C_b|} \quad (9)$$

式中： C_a 和 C_b 表示两个特征分类； $\phi_i = w_i / \sqrt{\|w_i\|^2}$ ；

$SIM_C(\phi_1, \phi_2) = \frac{\langle \phi_1, \phi_2 \rangle}{\|\phi_1\|, \|\phi_2\|}$ 为两个特征间的余弦相似性， $\langle \phi_1, \phi_2 \rangle$ 为两个特征向量的内积运算。

2.2 冗余滤波器剪枝

算法 1 为冗余卷积特征的剪枝算法，算法 2 是代表特征的提取算法，将滤波向量分组获得相似的权重空间。算法 1 调用了算法 2，将所有的滤波 ϕ_i 分为 n_r 个类， τ 为分类的相似性阈值，然后从每个簇随机采样一个代表滤波器。算法 2 的输出是每个类的代表特征索引列表 L_r ，算法 1 根据 L_r 处理 $\mathbf{W}^{(l)}$ 获得缩小的核矩阵 $\mathbf{W}_p^{(l)}$ ，基于 $\mathbf{W}_p^{(l)}$ 构建缩小的网络模型。

算法 1 冗余卷积特征的剪枝算法

1. FOREACH 层 IIN 训练模型 DO
2. 读取 $\mathbf{W}^{(l)}$ 的卷积特征；
3. 调节参数 τ ；
4. $(L_r, n_r) = \text{representation_feature}(\mathbf{W}^{(l)}, \tau)$ ； //提取代表滤波
5. $k = 0$ ；
6. FOR i IN L_r DO
7. $\mathbf{W}^{(l)}$ 的第 i 列拷贝到 $\mathbf{W}'^{(l)}$ 中； //初始化剪枝的模型
8. $k++$ ；
9. ENDFOR
10. 初始化 $\mathbf{W}'^{(l)}$ 的权重；
11. 微调剪枝的网络模型；

算法 2 representation_feature 函数

输入： \mathbf{W}, τ 。

输出： L_r, n_r 。

1. 遍历 \mathbf{W} 的向量，将相似性大于 τ 的向量分类。
2. 从每个分类中随机采样一个代表特征。
3. 返回代表特征的序号列表 L_r 和分类数量 n_r 。

图 2 为描述剪枝处理和不剪枝处理差异的示意图。原网络共有 5 个滤波，每个滤波对应核矩阵 $\mathbf{W}^{(l)}$ 的一列。滤波和三维输入图像 Z_l 卷积获得 5 个特征图 Z_{l+1} 。通过核矩阵 $\mathbf{W}^{(l+1)}$ 将特征图 Z_{l+1} 连接到 $l+2$ 层，设 $\mathbf{W}^{(l)}$ 的列分成 3 类 $\{1, 3, 5\}, \{2\}, \{4\}$ ，滤波 1、3、5 之间的相似性大于 τ 。调用算法 2 获得的索引列表为 $\{1, 2, 4\}$ ， $n_r = 3$ ，滤波 3 和滤波 4 标注为冗余滤波。剪枝处理在 Z_{l+1} 中将滤波 3 和滤波 4 及其他对应的特征图剪枝，同时删除下一个卷积层的特征图权重。

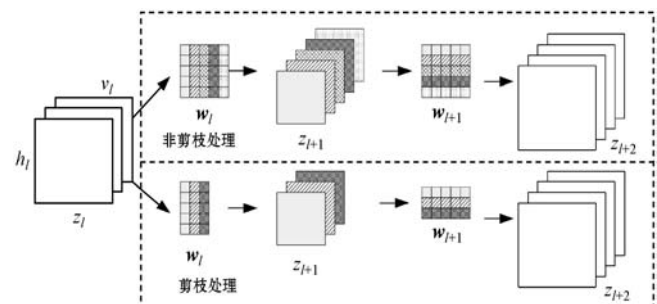


图 2 剪枝处理和不剪枝处理的差异图

3 多模态特征融合的超参数学习

神经网络包含随机失活处理和修正线性单元处理导致模型非线性且不平滑,所以一般采用群体智能和模拟退火等算法对网络超参数进行优化,但这些算法的寻优质量不佳。本文设计一种动态编码搜索算法的深度神经网络超参数高效率优化算法。

3.1 深度神经网络的超参数优化问题

将深度神经网络超参数优化视为一个黑盒误差函数 f 的全局优化问题,优化 f 的问题定义为:

$$\begin{aligned} \min_{x \in \mathbf{R}^D} f(x, \theta, Z_{\text{val}}) \quad (10) \\ \text{s. t. } \theta = \arg \min_{\theta} f(x, \theta, Z_{\text{train}}) \end{aligned}$$

式中: Z_{train} 和 Z_{val} 分别表示训练数据集和测试数据集; θ 为学习参数,通过最小化训练误差来学习 θ ; x 为一个超参数集。

神经网络结构相关的超参数包括学习率、损失函数、最小批大小、迭代次数和动量。神经元相关的超参数决定了每层的大小和非线性变换的方式,具体包括隐层单元数量、权重衰减、激活稀疏性、非线性、权重初始化、随机种子和预处理输入数据方式等。

3.2 动态编码搜索算法 (Dynamic Encoding Searching, DES)

DES^[11] 能够快速地搜索成本函数的全局最优值,分为局部搜索和全局搜索两个阶段。局部搜索负责开发局部区域,包括二分查找(Binary Search, BS)和单向搜索(Unidirectional Search, US)两个步骤。全局搜索负责搜索全局的最优区域,采用多起点方法实现。

(1) 局部搜索阶段。DES 局部搜索阶段包括 BS 和 US 两个步骤,BS 在任意二进制数的最低有效位(Least Significant Bit, LSB)右侧增加 0 或 1,使对应的实数值分别减小或增大。

定理 1 设一个 n 比特长的二进制数为 $s_p = a_n a_{n-1} \cdots a_1$, 其中 $a_i \in \{0, 1\}$, $i = 1, 2, \dots, n$, 将 s_p 称为父数。如果 s_p 的 LSB 增加 0, 生成的新数值称为左子数 s_c^l ; 如果 s_p 的 LSB 增加 1, 生成的新数值称为右子数 s_c^r 。将二进制数解码为实数的函数为:

$$f_d(b_m b_{m-1} \cdots b_1) = \frac{1}{2^m - 1} \sum_{j=1}^m b_j 2^{j-1} \quad (11)$$

式中: $b_i \in \{0, 1\}$, $i = 1, 2, \dots, m$ 。解码函数存在以下关系:

$$f_d(s_c^l) \leq f_d(s_p) \leq f_d(s_c^r) \quad (12)$$

如果 s_p 全为 0, 那么左侧相等; 如果 s_p 全为 1, 那么右

侧相等。超参数优化程序采用以下的无偏二进制解码函数:

$$f_d(b_m b_{m-1} \cdots b_1) = \frac{1}{2^{m+1}} \sum_{j=1}^m b_j 2^j + 1 \quad (13)$$

算法 3 是 DES 的局部搜索算法, 输入为一个 $n \times k$ 的二进制矩阵。首先加载上一次全局搜索的最优二进制矩阵 $\mathbf{B}_{n \times k}^*$, 将 $\mathbf{B}_{n \times k}^*$ 解码为实数向量 $\theta_{n \times 1}^*$ 。然后选择 x_i 为第一个搜索方向, BS 函数复制 $\mathbf{B}_{n \times k}^*$ 的第一行 $r_{1 \times k}$, 对 $r_{1 \times k}$ 的 LSB 分别加 0 和 1 获得两个子数 $s_{1 \times (k+1)}^{(0)}$ 和 $s_{1 \times (k+1)}^{(1)}$ 。迭代计算每个向量的成本值, 选出其中的最优方向 d_{opt} 和最优子数 $s_{1 \times (k+1)}^*$ 。BS 函数选择下一行继续处理, 最终最优矩阵变为一个伪矩阵。

BS 函数处理 x_1 之后, 沿着最优方向 d_{opt} 重复运行 US 函数, 直至解的质量不再提高。如果 d_{opt} 为 0, 则产生相邻数值 $s' = s^* - 1$; 如果 d_{opt} 为 1, 则 $s' = s^* + 1$ 。US 函数程序获得最优的行 s^* , 将 s^* 保存于最优矩阵 \mathbf{D}^* 的第一行。然后 BS 开始 x_2 方向的局部搜索, 最终完成对所有行的搜索, 最优矩阵 \mathbf{D}^* 变为一个矩形矩阵。

算法 3 局部搜索算法

1. $\mathbf{B}^* \rightarrow \theta_{n \times 1}^*$; // 解码初始化矩阵
2. **FOREACH** $i = 1$ **TO** n **DO**
3. BS(\mathbf{B}^*); // 二分查找
4. US(\mathbf{B}^*); // 单向编码搜索
5. **ENDFOR**
- /* BS(\mathbf{B}^*) 函数 */
6. $r_{1 \times k} = \mathbf{B}^*(i, 1:k)$; // 读取当前最优矩阵的第 i 行
7. **FOREACH** $j = 0$ **TO** 1 **DO**
8. $\theta^j \leftarrow \theta^*$; // 读取当且最优的参数向量
9. $s_{1 \times (k+1)}^j = [r_j]$; // 第 i 行向量加一个二进制比特 j ;
10. $\theta_{(i)}^j = f_d(s^j)$; // 更新解码的参数;
11. $J^j = f(\theta^j)$; // 评估新参数向量的成本;
12. **ENDFOR**
13. $J^* = \min(J^0, J^1)$; // 选择更小的成本函数;
14. **IF** $J^* == J^0$ **THEN**
15. $d_{\text{opt}} = 0, s^* = s^0, \theta^* = \theta^0$;
16. **ELSE**
17. $d_{\text{opt}} = 1, s^* = s^1, \theta^* = \theta^1$;
18. **ENDIF**
- /* US(\mathbf{B}^*) 函数 */
19. **WHILE** 不满足迭代结束调节 **DO**
20. $\theta^l \leftarrow \theta^*$;
21. **IF** $d_{\text{opt}} == 0$ **THEN**
22. $s' = s^* - 1$;
23. **ELSE**
24. $s' = s^* + 1$;
25. **ENDIF**
26. 采用第 i 个参数更新参数向量;

```

27.  $\theta'(i) = f_d(s')$ ;
28.  $J = f(\theta')$ ; //评估新参数向量的成本;
29. IF  $J < J^*$  THEN
30.      $s^* \leftarrow s', J^* \leftarrow J$ ;
31. ELSE
32.     RETURN;
33. ENDIF
34. ENDWHILE
    
```

(2) 基于多起点的全局搜索阶段。搜索空间分为连续逐渐缩小的网格,根据成本函数的形状自动决定搜索路径。因为不同初始化矩阵的搜索路径可能在一个子矩阵出现重复,导致重复计算,为此创建一个查找表,为二进制矩阵在表中产生一条记录,通过比较记录来避免重复搜索。DES 通过检查 LENTH_T 和 COST_T 两个参数跳出局部最小。如果当前最优矩阵的行长度等于 LENTH_T,且成本大于 COST_T,则停止搜索,随机选择一个新起点重新搜索。本文将 LENTH_T 设为矩阵最小的行长度, COST_T 设为局部最优和局部次优值的中位数。

(3) 基于 DES 的超参数优化方法。图 3 为超参数优化方法的流程图。第一步,预处理步骤确定并初始化神经网络的超参数,例如:批大小、epochs 数量、学习率等。该步骤也需要确定深度神经网络的类型,包括卷积神经网络、递归神经网络、对抗神经网络、自编码器等。第二步,使用初始化值训练超参数,更新 DES 的成本函数。第三步,采用更新的成本函数再次运行 DES 获得新的超参数,采用新的超参数重新初始化和训练神经网络,经过 DES 的若干次迭代处理,获得最小化网络成本的最优超参数。

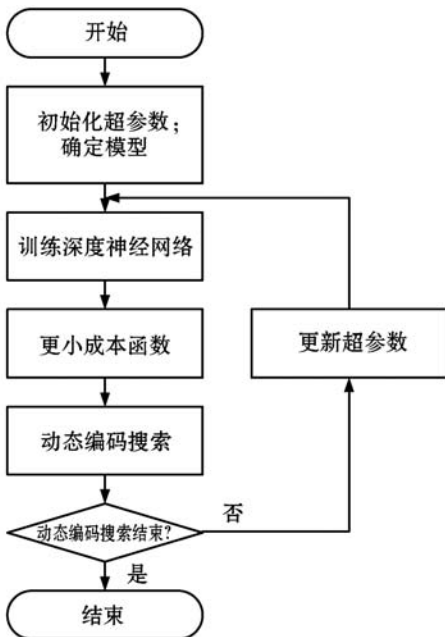


图 3 DES 优化超参数的流程框图

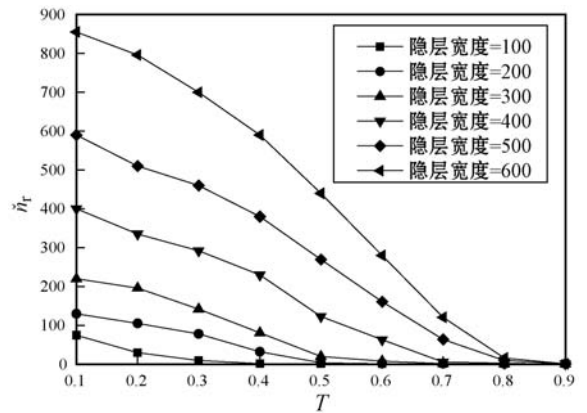
4 仿真实验

4.1 实验数据集和实验环境

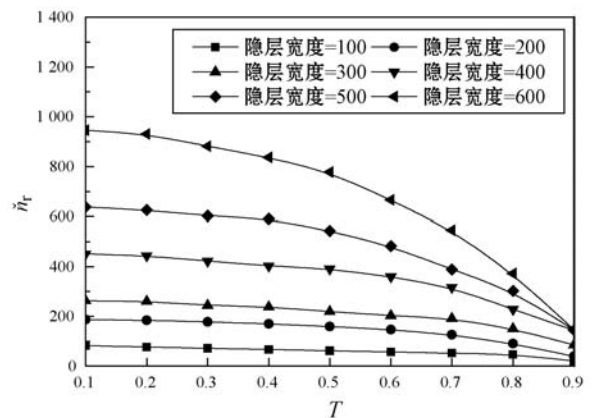
实验环境为 CPU Intel Core i7-6700,主频为 3.4 GHz,内存为 64 GB,操作系统为 64 位 Ubuntu 14.04。采用经典的 MNIST 数据集单独测试本方案每个机制的有效性,数据集包含 60 000 幅训练图像和 10 000 幅测试图像,图像大小均为 28 × 28 像素。

4.2 特征剪枝实验

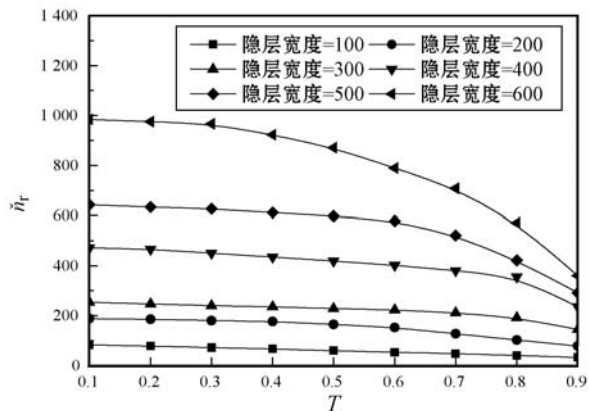
为了评估本文特征剪枝方案的性能,采用多层感知机直接观察本文方案删除冗余特征的效果,直接将本文的特征剪枝模块和多层感知机集成。采用 Adam 实验方法^[12]统计感知机的特征变化,在每次多层感知机模型训练完成之后,统计冗余特征的变化曲线,将网络所有层的平均冗余特征数量表示为 $\bar{n}_r = n'_l - n_l$, n'_l 表示 n_l 层的滤波总数量, n_l 表示剪枝完保留的特征数量。图 4 是 1 个隐层、2 个隐层和 4 个隐层的冗余特征统计结果,图中显示隐层的宽度和深度都会引起 \bar{n}_r 升高,隐层数量越多,平均冗余特征的数量越高,特征的相关性也越高。



(a) 1 个隐层的实验



(b) 2 个隐层的实验



(c) 4 个隐层的实验

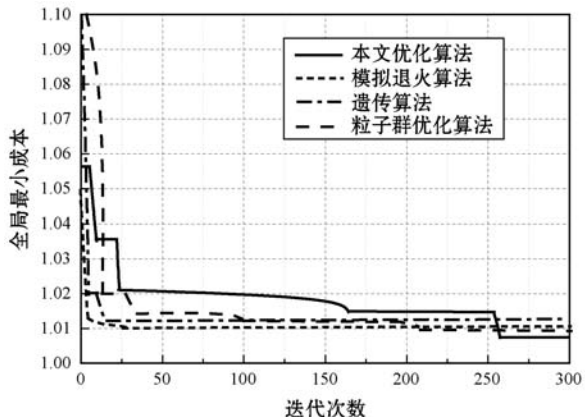
图 4 多层感知机的特征剪裁实验结果

4.3 DES 的超参数优化实验

为了评估本文超参数优化机制 DES 的效果,将其与模拟退火^[13]、遗传算法^[14]、粒子群优化算法^[15]等经典优化方法比较。采用图 5(a)所示的卷积神经网络模型,将分类精度的倒数作为成本函数。图 5(b)是 4 个优化算法的收敛曲线,模拟退火算法在第 8 次迭代的成本为 1.011 43,遗传算法在第 23 次迭代的最小成本为 1.012 65,粒子群算法在第 233 次迭代的最小成本为 1.011 071,本文算法在第 263 次搜索的最小成本值为 1.008 98。本文算法实现最低的最小成本,卷积层神经元数量为 64,全连接层的神经元数量为 512,学习率为 10^{-3} , $p=0.5$ 。



(a) 卷积神经网络的结构

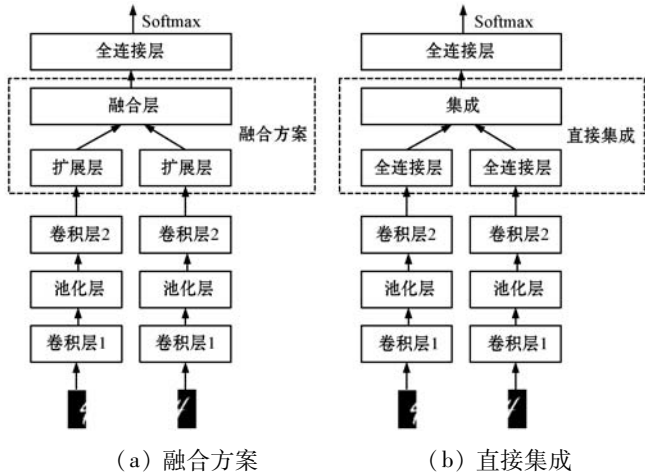


(b) 超参数优化程序的收敛曲线

图 5 DES 的超参数优化实验

4.4 两个模态的融合实验

将 MNIST 的每个图像剪裁为左右两半,图 6 是两个模态的卷积神经网络结构,采用 4.3 节的超参数,图中输入是数字“4”剪裁为左右两半的实例。设置两个卷积层,直接集成法直接聚合全连接层的结果,传递至 Softmax 层。本文融合方法对两个全连接层的结果进行融合,再传递至 Softmax 层。



(a) 融合方案

(b) 直接集成

图 6 两个模态的卷积神经网络结构

使用 60 000 个双模态样本训练网络模型,测试了 10 000 个测试样本的分类错误率,分别测试了完整图像、左半边图像缺失和右半边图像缺失的情况,缺失的半边图像替换为全零的数据。表 1 为实验结果,可看出本文融合方法在完整图像的情况下实现微小的提高,在缺失半边图像的情况下则实现了大幅度的提高。

表 1 MNIST 数据集的平均分类错误率 %

方案	全图	缺失右半图	缺失左半图
直接集成	0.95	15.17	16.06
本文融合方案	0.93	3.35	4.31

4.5 对比实验分析

(1) 实验数据集。OPPORTUNITY^[16]是一个多个模态的动作识别数据集,数据集由 5 个自由动作序列和 1 个受限动作序列组成,所有数据的采样频率为 30 Hz。本文构建运动模式识别和手势识别两组实验,运动模式分为 4 个类,即站立、行走、坐和躺,手势分为 17 个类,即开洗碗机、关洗碗机、开吹风机 1、开吹风机 2、开吹风机 3、关吹风机 1、关吹风机 2、关吹风机 3、开冰箱、关冰箱、开门 1、开门 2、关门 1、关门 2、移动杯子、清洁桌子、其他。

人物 1 和人物 2 的数据集作为训练数据集,人物 3 的数据集作为测试数据集。将所有的采样数据归一化到 $[-1, 1]$ 。滑动窗口设为 500 ms,测试数据的步长设为 250 ms,训练数据的步长设为 33.3 ms。

(2) 性能评价指标。采用 F_1 评价分类的性能, 定义为:

$$F_1 = \sum_i 2 \times \frac{n_i}{n} \times \frac{pre_i \times rec_i}{pre_i + rec_i} \quad (14)$$

式中: n_i 为第 i 个类的数据量; n 为总数据量; pre_i 为第 i 个类的准确率; rec_i 为第 i 个类的召回率。(n_i/n) 项能够平衡偏态分布的分类。

(3) 对比算法。目前深度学习多模态融合技术依然较少, 选择 3 个优质的多模态技术与本文方法作比较。MPC^[17] 是一种专门为卷积神经网络设计的多模态分类算法, 该算法主要对池化层进行线性融合, 但无法扩展到其他神经网络模型。MDLAR^[18] 是一种专门为动作识别设计的多模态受限玻尔兹曼机模型, 该模型对动作识别的性能较好。MDL-CW^[19] 是一种基于权值的跨模态深度神经网络融合算法, 通过多个阶段的预训练迭代调节每个模态的权值。其中两个方案都提供了卷积神经网络的实现方案, 因此将本文方法和卷积神经网络集成。

(4) 网络模型。本文多模态卷积神经网络的网络结构如图 7 所示。

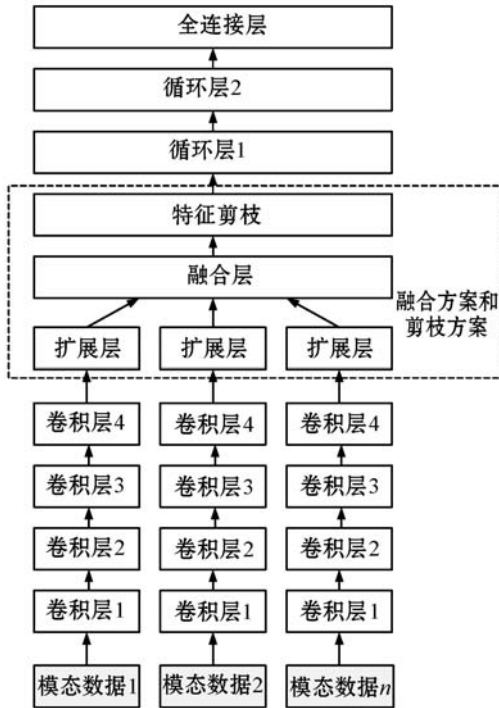
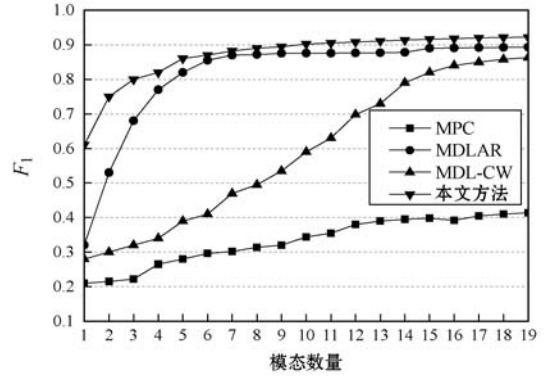


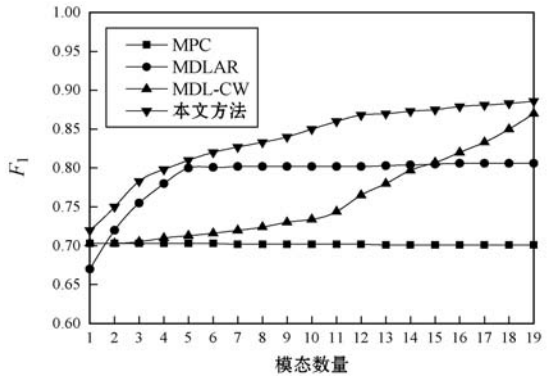
图 7 本文方法和卷积神经网络的集成结构

(5) 实验结果和分析。图 8 为 4 个多模态神经网络的实验结果。MPC 主要对池化层进行线性融合, 对多模态之间的非线性拟合能力不足, 因此其总体分类性能较低。MDL-CW 是一种基于权值的跨模态深度神经网络融合算法, 通过多个阶段的预训练迭代调节每个模态的权值, 基于权值融合的方法在特征量少的情况下难以实现理想的学习效果, 而在特征量大的情况

下容易出现过拟合, 其总体分类性能也较低。MDLAR 和本文方法的性能明显好于另外两个算法, MDLAR 通过对多模态玻尔兹曼机的非线性融合, 对多模态特征的学习能力较强, 并且该模型对动作识别问题制定了专门的超参数学习机制, 实现理想的性能。本文方法通过多模态特征融合和特征剪枝处理, 不仅有效地实现了非线性特征学习, 而且也避免过拟合问题, 最终实现了最佳的分类性能。



(a) 运动模式分类实验



(b) 手势分类实验

图 8 多模态分类算法的结果比较

5 结 语

为了解决深度神经网络的多模态特征融合和分类问题, 本文设计基于深度神经网络的多模态特征自适应聚类方法。该方法能够兼容不同的深度学习网络, 分别与多层感知机和卷积神经网络进行集成实验, 实现稳定的性能。本文方法考虑跨模式的特征相关性, 学习程序通过正则化处理防止过拟合, 对于数据缺失也具有鲁棒性。

因为本文方法需要将多个深度神经网络融合, 因此超参数的学习是一个难题, 本文设计的动态编码搜索策略能够提高神经网络的优化效果, 但对于大量的模态融合问题, 还存在计算时间过长的问題。未来将关注于对多个神经网络的超参数进行联合优化, 以期提高模型的学习效率。

参 考 文 献

- [1] Savchenko A V, Belova N S, Savchenko L V. Fuzzy analysis and deep convolution neural networks in still-to-video recognition[J]. *Optical Memory & Neural Networks*, 2018, 27 (1):23 - 31.
- [2] Darojah Z, Ningrum E S. The extended kalman filter algorithm for improving neural network performance in voice recognition classification[C]//*International Seminar on Intelligent Technology & Its Applications*, 2017.
- [3] 颜菲,胡玉平. 叠加去噪自动编码器结合深度神经网络的心电图信号分类方法[J]. *计算机应用与软件*, 2019, 36 (4):178 - 185.
- [4] 王鹏,蔡锐,杨士强.“文本为主”的多模态特征融合的新闻视频分类算法[J]. *清华大学学报(自然科学版)*, 2005, 45(4):475 - 478.
- [5] 征察,吉立新,李邵梅,等. 基于多模态信息融合的新闻图像人脸标注[J]. *计算机应用*, 2017, 37(10):3006 - 3011.
- [6] Bodla N, Zheng J, Xu H, et al. Deep heterogeneous feature fusion for template-based face recognition[C]//*Applications of Computer Vision*, 2017.
- [7] 原福永,冯凯东,李晨,等. 融合用户经历的多策略自适应推荐模型[J]. *小型微型计算机系统*, 2019, 40(7):1410 - 1415.
- [8] 郭宁宁,王宝亮,侯永宏,等. 融合社交网络特征的协同过滤推荐算法[J]. *计算机科学与探索*, 2018, 12(2):208 - 217.
- [9] 张露,王华彬,陶亮,等. 基于分类距离分数的自适应多模态生物特征融合[J]. *计算机研究与发展*, 2018, 55(1):151 - 162.
- [10] 周安众,罗可. 一种卷积神经网络的稀疏性 Dropout 正则化方法[J]. *小型微型计算机系统*, 2018, 39(8):1674 - 1679.
- [11] Yun J P, Jeon Y J, Choi D C, et al. Real-time defect detection of steel wire rods using wavelet filters optimized by univariate dynamic encoding algorithm for searches[J]. *Journal of the Optical Society of America A*, 2012, 29(5):797 - 807.
- [12] Kingma D P, Ba J. Adam: a method for stochastic optimization[EB]. arXiv:1412.6980, 2014.
- [13] Roshani A, Giglio D. Simulated annealing algorithms for the multi-manned assembly line balancing problem: minimising cycle time [J]. *International Journal of Production Research*, 2017, 55(10):21.
- [14] Sheikh R H, Raghuvanshi M M, Jaiswal A N. Genetic algorithm based clustering: a survey[C]//*International Conference on Emerging Trends in Engineering & Technology*, 2008.
- [15] Masdari M, Salehi F, Jalali M, et al. A survey of PSO-Based scheduling algorithms in cloud computing[J]. *Journal of Network & Systems Management*, 2017, 25(1):122 - 158.
- [16] Gioanni L, Dartigues-Pallex C, Lavirotte S, et al. Opportunistic human activity recognition: a study on opportunity dataset[C]//*International Conference on Mobile & Ubiquitous Systems: Computing*, 2016.
- [17] Algashaam F M, Nguyen K, Alkanhal M, et al. Multispectral periocular classification with multimodal compact multi-linear pooling[J]. *IEEE Access*, 2017, 5:14572 - 14578.
- [18] Radu V, Lane N D, Bhattacharya S, et al. Towards multimodal deep learning for activity recognition on mobile devices[C]//*2016 ACM International Joint Conference on Pervasive & Ubiquitous Computing: Adjunct*, 2016.
- [19] Rastegar S, Baghshah M S, Rabiee H R, et al. MDL-CW: a multimodal deep learning framework with cross weights[C]//*2016 IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 2016.
- ~~~~~
- (上接第 248 页)
- [4] Smith S M, Brady J M. SUSAN—a new approach to low level image processing[J]. *International Journal of Computer Vision*, 1997, 23(1):45 - 78.
- [5] Rosten E, Porter R, Drummond T. Faster and better: a machine learning approach to corner detection[J]. *IEEE Trans Pattern Anal Mach Intell*, 2008, 32(1):105 - 119.
- [6] Rublee E, Rabaud V, Konolige K, et al. ORB: an efficient alternative to SIFT or SURF[C]//*2011 International Conference on Computer Vision*, 2011.
- [7] Leutenegger S, Chli M, Siegwart R Y. BRISK: binary robust invariant scalable keypoints[C]//*2011 International Conference on Computer Vision*, 2011.
- [8] 高翔,万成浩,李润生. 一种基于 Harris-Laplace 算法的亚像素角点检测方法[J]. *测绘科学技术学报*, 2017, 34(5):475 - 480.
- [9] Wu M. Research on optimization of image fast feature point matching algorithm[J]. *EURASIP Journal on Image and Video Processing*, 2018(1):1 - 27.
- [10] 李伟生,李泽亚. 一种改进的多尺度自适应角点检测方法[J]. *计算机应用与软件*, 2016, 33(1):185 - 189.
- [11] 周保余,赵宏伟,肖杨,等. 基于局部熵的图像特征描述方法[J]. *吉林大学学报(工学版)*, 2017, 47(2):601 - 608.
- [12] 蔡青,刘慧英,孙景峰,等. 基于信息熵的自适应尺度活动轮廓图像分割模型[J]. *西北工业大学学报*, 2017, 35(2):286 - 291.
- [13] 高飞,王聪,矫东航. 基于分块信息熵和特征尺度的图像配准算法[J]. *北京理工大学学报*, 2016, 36(11):1194 - 1199.