



# 基于多注意力机制的深度神经网络故障诊断算法

王翔, 任佳

(浙江理工大学机械与自动控制学院, 杭州 310018)

**摘要:** 针对现有的故障诊断算法难以深入挖掘复杂过程数据内在信息的问题, 引入深度神经网络增强故障诊断模型的非线性表征能力, 并在此基础上引入三种注意力机制对特征之间的非线性关系进行建模, 提出了一种基于多注意力机制的深度神经网络故障诊断算法。该算法首先引入特征位置嵌入方法生成特征位置向量, 并将其同特征向量一并作为深层网络的输入; 然后通过注意力机制计算相应的注意力特征, 完成故障类型诊断; 最后将该算法应用到田纳西-伊斯曼过程(Tennessee-Eastman process, TEP)故障诊断中进行性能验证, 并与常规的数据驱动方法进行对比。实验结果表明, 该算法的平均 F1 分数比常规的数据驱动方法高 10%~15%。

**关键词:** 故障诊断; 注意力机制; 自注意力; 深度神经网络; 田纳西-伊斯曼过程

中图分类号: TH7

文献标志码: A

文章编号: 1673-3851(2020)03-0224-08

## Deep neural network fault diagnosis algorithm based on multi-attention mechanism

WANG Xiang, REN Jia

(Faculty of Mechanical Engineering & Automation, Zhejiang  
Sci-Tech University, Hangzhou 310018, China)

**Abstract:** It is difficult for existing fault diagnosis algorithms to deeply mine intrinsic information of complex process data. For this problem, a deep neural network is introduced to enhance nonlinear representation capacity of the fault diagnosis model. On this basis, three attention mechanisms are introduced to model the non-linear relationship among the features. Besides, a deep neural network fault diagnosis algorithm based on multi-attention mechanism is proposed. First of all, the feature position vectors are generated by feature position embedding method, and the feature position vectors are used as the input of deep network together with the feature vectors. Then the attention mechanism is used to calculate the attention features to diagnose the types of faults. Finally, the algorithm is used for Tennessee Eastman process (TEP) fault diagnosis to test the performance, accompanied with the comparison with the conventional data-driven methods. The results show that the average F1 fraction of the proposed algorithm is 10%~15% higher than that of conventional data-driven methods.

**Key words:** fault diagnosis; attention mechanism; self-attention; deep neural network; Tennessee-Eastman process

收稿日期: 2019-08-28 网络出版日期: 2019-12-02

基金项目: 浙江省自然科学基金项目(LY17F030024); 浙江省公益技术研究项目(LGG20F030007)

作者简介: 王翔(1995-), 男, 浙江温州人, 硕士研究生, 主要从事深度学习方面的研究。

通信作者: 任佳, E-mail: jren@zstu.edu.cn

## 0 引言

随着科技的不断进步和发展,化工行业的规模在不断扩大,所产生的经济价值也日益增加。然而化工生产一旦发生事故,往往会造成较为严重的财产损失甚至人身伤亡。因此,开发安全高效的智能故障检测和诊断(Fault detection and diagnosis, FDD)系统便显得尤为重要<sup>[1]</sup>。处理化工过程故障通常需要以下三个步骤:首先,当故障出现时,故障检测系统应及时检测到故障。其次,在检测到故障后,系统应隔离和分类故障。最后,应部署相应的解决方案,使得化工过程恢复正常。本文主要研究第二个步骤,即故障诊断。故障诊断是一个多分类问题,在给定输入数据的情况下,故障诊断系统根据故障模式预测出特定的故障类型。根据建模原理,故障诊断算法可以分为数据驱动和模型驱动两大类。基于模型驱动的故障诊断算法需要半定量或定性的模型来实现精确的化工过程建模,但由于化工过程的工业系统高度集成且日益复杂化,基于模型驱动的故障诊断算法难以应用于当前的化工过程中。与此同时,随着化工过程数据储备的日益丰富和计算机处理速度的提高,基于数据驱动的故障诊断算法受到越来越多的关注。目前,主流的基于数据驱动的故障诊断算法有:贝叶斯网络分类器(Bayesian network classifier, BNC)<sup>[2]</sup>、Fisher判别分析(Fisher discriminant analysis, FDA)<sup>[3]</sup>、主成分分析(Principal component analysis, PCA)<sup>[4]</sup>、支持向量机(Support vector machine, SVM)<sup>[5]</sup>和人工神经网络(Artificial neural network, ANN)<sup>[6]</sup>。虽然上述数据驱动算法在化工过程的故障诊断中取得了较好的效果,但由于其仅具有浅层结构,故难以有效处理具有强非线性关系的化工过程数据。

深度学习是一种深层的机器学习模型,利用多层非线性映射的堆叠,挖掘数据与变量间的相互关系,从而实现更高层次的、更加抽象的数据表达<sup>[7-8]</sup>。通过将深度神经网络引入故障诊断领域,能够较好地克服主流故障诊断算法学习能力不足的问题,更充分地利用化工过程的数据信息。但随着模型的复杂度增加,深度神经网络易受非敏感特征的影响,导致故障诊断精度下降。

注意力机制(Attention mechanism, AM)通过构建注意力矩阵,使深度神经网络在训练过程中关注重点特征,从而能较好地避免非敏感特征的影响。注意力机制最早被应用于机器视觉领域,其作用是

使图像中需要重点关注的区域获得更多的注意力<sup>[9-10]</sup>。2016年Bahdanau等<sup>[11]</sup>将注意力机制用于机器翻译任务中,进一步验证了注意力机制可以有效地刻画特征间的关系,从而推动了结合注意力机制的深层神经网络的研究。随后,Vaswani等<sup>[12]</sup>将自注意力机制引入句子建模任务中,使用二维矩阵表征句子信息,进而获得语义信息更丰富的特征表示。

基于以上文献分析,使用注意力机制可以有选择地进行表征,从而有效地克服深度神经网络易受非敏感特征影响的问题,并且更充分地利用特征及特征间的信息。由于注意力机制拥有上述优良的特性,因此其被广泛应用于机器视觉和自然语言处理等领域。然而,化工过程数据多为数值型数据,且变量间存在复杂的非线性、耦合关系<sup>[13]</sup>,故无法简单地将注意力机制迁移到化工故障诊断领域中。本文针对上述问题引入特征位置嵌入方法生成特征位置向量,并结合三种注意力机制,提出一种基于多注意力机制的深度神经网络(Multi-attention deep neural network, MA-DNN)故障诊断算法。三种注意力机制应用过程为:a)引入位置向量自注意力机制,将每个特征的位置向量和所有特征的位置向量进行计算,从而有效地表示特征间的关系;b)引入特征向量自注意力机制,将值信息加入位置向量中得到新的特征向量,再计算每个特征向量和所有特征向量间的关系,从而有效地获得当前输入特征间的关系;c)引入交互注意机制,计算输入和位置向量间的关系,有效地表示当前输入下每个特征的重要程度。通过三种注意力机制充分表示特征间的关系,深度神经网络在训练过程中能关注与当前输入相关的特征信息,从而增强网络的故障诊断能力。本文将所提出的算法应用于田纳西-伊斯曼过程(Tennessee-Eastman process, TEP)中,并与主流的基于数据驱动的故障诊断算法进行比较,对本文所提算法的故障诊断性能进行验证。

## 1 MA-DNN 算法实现

化工过程的机理复杂,其数据特征间存在强非线性关系,并且常包含非敏感特征。基于上述问题,本文引入三种注意力机制充分利用特征及特征间的信息,并结合深度神经网络挖掘深层次的化工过程信息,提出MA-DNN模型。

### 1.1 MA-DNN 模型结构

MA-DNN整体结构如图1所示。MA-DNN模型共包含两个输入层、三个注意力机制层、一个融合

层以及一个 Softmax 层。模型通过特征位置嵌入方法生成特征位置向量,使用三种注意力机制准确表征出在当前输入下每个特征的重要程度和特征间

的关系。在使用融合层合并注意力特征向量后,利用全连接神经网络对合并后的向量进行映射,并通过 Softmax 函数输出最终的分类结果。

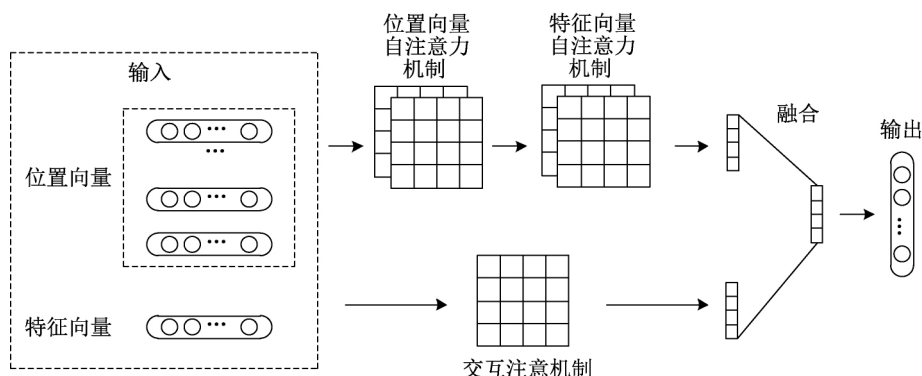


图1 MA-DNN 整体结构示意图

注意力机制可以描述为将查询矩阵(Query,  $Q$ )、键矩阵(Key,  $K$ )以及值矩阵(Value,  $V$ )映射为一个输出矩阵的过程,其计算过程如式(1)所示:

$$Attention(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d}}\right)V \quad (1)$$

其中: $d$  为键矩阵中每个列向量的维度,起调节作用,防止内积过大。

通过引入注意力机制,深度神经网络在训练过程中可以关注重点特征,减少非敏感特征的影响。然而,不同于图像与文本数据,单一的注意力机制难以充分利用数值型的化工过程数据的特征及特征间的信息。故本文在原有注意力机制的基础上,引入三种注意力机制建模范征间的关系。下面分别对三种注意力机制进行具体阐述。

## 1.2 基于位置向量的自注意力机制

为了获得更丰富的特征信息,本文使用一种特征位置嵌入方法,针对每一个特征随机生成一个特征位置向量,并通过反向传播算法更新特征位置向量,其构建步骤如图2所示。

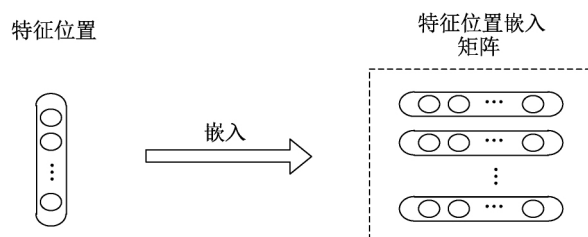


图2 特征位置嵌入构建步骤示意

通过引入特征位置向量实现对特征本身意义的表征,但若仅将其作为额外信息引入网络并不能很好地表征特征间的关系。为了充分利用特征位置向量的信息,本文使用1.1节所提的注意力机制计算特征位置向量间的关系,同时为了从更多表示子空间内学习特征间的关系,引入如图3所示的多头自注意力机制。通过将原输入向量映射为多个不同的 $Q$ 、 $K$ 和 $V$ ,多次计算注意力机制,最后将不同的 $Q$ 、 $K$ 和 $V$ 计算出的输出进行拼接,从而得到表达能力更多样的新特征向量。

为了加速网络的收敛,同时保证整个计算过程的稳定,本文采用层归一化对深度神经网络每一层

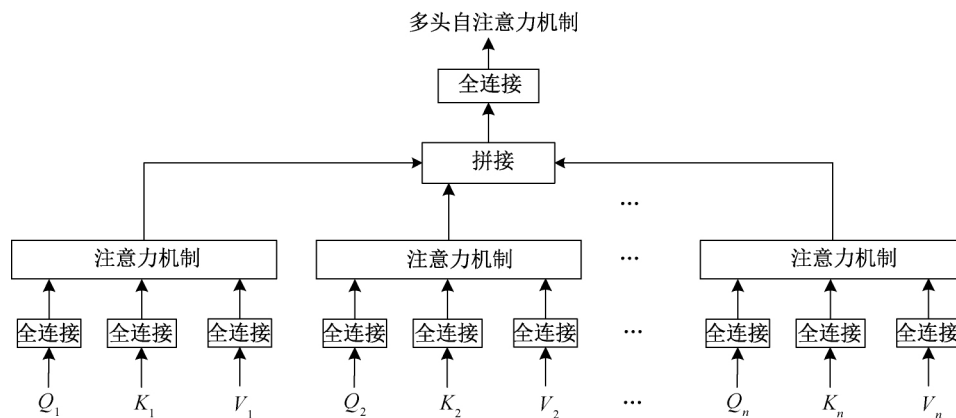


图3 多头自注意力机制示意

的所有神经元的输入都进行标准化,其中均值和方差采用式(2)–(3)计算:

$$\mu^l = \frac{1}{H} \sum_{i=1}^H x_i^l \quad (2)$$

$$\sigma^l = \sqrt{\frac{1}{H} \sum_{i=1}^H (x_i^l - \mu^l)^2} \quad (3)$$

其中: $x_i^l$ 表示第 $l$ 层的第 $i$ 个神经元的输入, $H$ 表示该隐层的神经元数。

最后,为了防止随着网络的加深而产生退化问题,本文引入残差结构<sup>[14]</sup>将原始输入信息叠加到经过注意力网络之后的输出向量中,其计算过程如式(4)所示:

$$\mathbf{X}^{l+1} = \text{LayerNorm}(\mathbf{X}^l + \text{MultiHead}(\mathbf{Q}^l, \mathbf{K}^l, \mathbf{V}^l)) \quad (4)$$

其中: $\mathbf{X}^l$ 表示第 $l$ 层的输入, $\mathbf{Q}^l$ 、 $\mathbf{K}^l$ 和 $\mathbf{V}^l$ 表示第 $l$ 层的注意力机制所需的查询矩阵、键矩阵和值矩阵。

### 1.3 基于特征向量的自注意力机制

尽管通过基于位置向量的自注意力机制可以获得特征本身意义的相关性,但此时特征间的关系并未考虑特征当前的输入值,故并不能很好地表征当前输入下每个特征间的关系。

为了更好地表征特征间的关系,本文将每个特征位置所对应的值叠加到该位置向量上,从而获得既具有本身信息又含有特征当前输入值信息的特征向量。通过如图4所示的基于特征向量的自注意力机制方法,将其他特征信息引入当前位置的特征向量表示中,使用得每个位置的特征向量表示不再独立,从而更好地将特征间的非线性关系融于特征向量中。

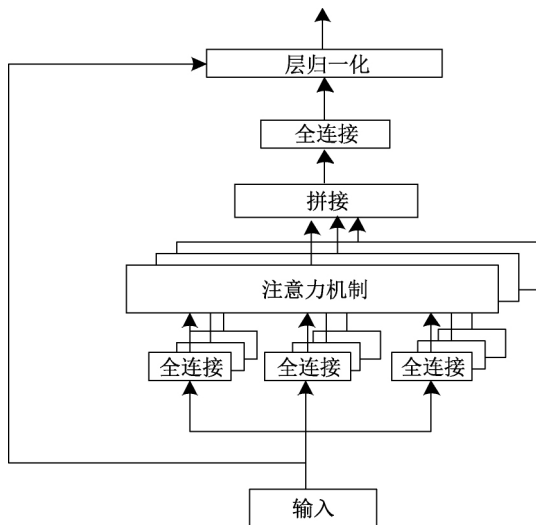


图4 基于特征向量的自注意力机制

同时,为了提取与学习到更多的隐藏信息,本文使用如式(5)–(6)所示的前馈神经网络对经过自注

意力机制之后的特征矩阵进一步提取特征:

$$\text{FFN}(\mathbf{X}^l) = \max(0, \mathbf{X}^l \mathbf{W}_1^l + \mathbf{b}_1^l) \mathbf{W}_2^l + \mathbf{b}_2^l \quad (5)$$

$$\mathbf{X}^{l+1} = \text{LayerNorm}(\mathbf{X}^l + \text{FFN}(\mathbf{X}^l)) \quad (6)$$

其中: $\mathbf{W}_1^l$ 和 $\mathbf{W}_2^l$ 是两个全连接层权重矩阵, $\mathbf{b}_1^l$ 和 $\mathbf{b}_2^l$ 是两个全连接层的偏置。该前馈神经网络主要由两层全连接组成,两层直接使用ReLU作为激活函数,并使用残差结构防止退化问题。最后,通过层归一化的方法加速网络的收敛、稳定计算过程。

### 1.4 交互注意力机制

基于位置向量和特征向量的自注意力机制较好地刻画了特征间的关系,但并未建立当前输入值和特征本身的关系。为了表征这一关系,本文引入如图5所示的交互注意力机制方法。通过当前输入向量与每个特征位置向量的交互,计算在当前输入下各特征的重要性,从而合理分配计算资源,减小甚至忽略非关键特征的影响。

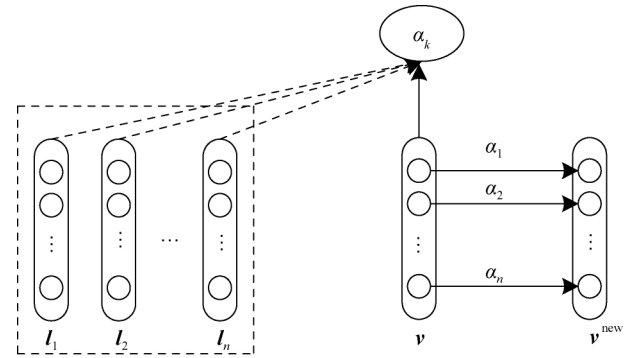


图5 交互注意力机制示意

图5中, $l_1, l_2, \dots, l_n$ 为相应的位置向量, $n$ 为输入的特征维数, $v$ 是当前输入的特征向量, $v^{\text{new}}$ 是经过交互注意力机制之后最终生成的新特征向量。 $\alpha_k$ 为位置向量对当前输入的注意力权重,其计算公式为:

$$\alpha_k = \frac{\exp(f_k)}{\sum_{i=1}^n \exp(f_i)} \quad (7)$$

$$f_i = \mathbf{A}_i \tanh(\mathbf{W}v + \mathbf{I}_i l_i) \quad (8)$$

其中: $l_i$ 表示第 $i$ 个特征所对应的特征位置向量, $\mathbf{W}$ 、 $\mathbf{I}_i$ 和 $\mathbf{A}_i$ 是相应的全连接层权重矩阵,通过网络根据损失函数自学习得到, $f_i$ 是未归一前的第 $i$ 个特征位置向量对应的注意力权重。

最后,将通过交互注意力机制得到的注意力权重与当前输入的特征向量对应位相乘,获得全新的特征向量 $v^{\text{new}}$ 。通过计算位置向量对当前输入的特征向量的注意力权重分布,网络能聚焦于重点信息,

从而获得能够突出重点信息的优化特征向量。

### 1.5 基于 MA-DNN 的故障诊断

MA-DNN 引入深度神经网络增强模型处理非线性数据的能力,并且结合三种注意力机制有选择地进行表征,减少非敏感特征对模型的影响。具体实现流程如下所示。

- a) 利用特征位置嵌入方法将当前输入样本的每一维特征都映射为一个特征位置向量;
- b) 对特征位置向量进行层归一化处理;
- c) 通过多头自注意力机制,计算当前输入样本的特征位置向量之间的关系,从而生成新的特征位置向量;
- d) 拼接原始的特征位置向量和 c) 中新生成的特征位置向量,生成新的特征位置向量;
- e) 新生成的特征位置向量都叠加上当前输入样本该特征的值,从而当前输入样本的每一维特征都生成了一个由该特征的特征位置向量加上对应特征值的特征向量;
- f) 对特征向量进行层归一化处理;
- g) 通过多头自注意力机制,计算当前输入样本的特征向量之间的关系,从而生成新的特征向量;
- h) 使用式(5)对 g) 中新生成的特征向量进行特征提取;
- i) 拼接原始的特征向量和 h) 中新生成的特征向量,生成新的特征向量;
- j) 利用式(7)~(8)计算交互注意力权重,并与

当前输入的特征对应项相乘,最后所得结果与 i) 中新生成的特征向量拼接形成新的特征向量;

k) 利用全连接神经网络对合并后的向量进行映射,并通过 Softmax 函数输出最终的分类结果;

l) 计算交叉熵;

m) 使用 Adam 算法更新模型参数;

n) 重复上述过程,直至达到最大迭代次数。

综上,本文提出一种基于 MA-DNN 的故障诊断方法,其整体流程如图 6 所示。整个模型分为离线建模训练和在线诊断两个阶段。在离线建模阶段,利用所收集的历史数据的均值和方差对数据进行标准化处理并对每一个特征都随机生成一个特征位置向量,随后将经过标准化之后的特征和特征位置向量一同作为 MA-DNN 的输入。在训练阶段,采用交叉熵作为损失函数,使用 Adam 算法<sup>[7]</sup>训练优化模型,该优化算法可以自适应地调整各个参数的学习率,从而帮助网络更快收敛。模型训练完毕之后,通过验证集对模型的故障诊断性能进行评估,若模型性能达到实际部署的要求,便将其投入实际的化工生产过程中进行在线故障诊断,否则需要调整网络参数和架构重新训练。进行在线故障诊断时,从实际化工过程中采集待诊断样本并进行标准化处理,并且使用训练阶段生成的特征位置向量矩阵一同作为输入,通过 MA-DNN 模型诊断故障的类型。此外,定时抽取部分数据作为测评数据,用以评测模型是否需要更新迭代。

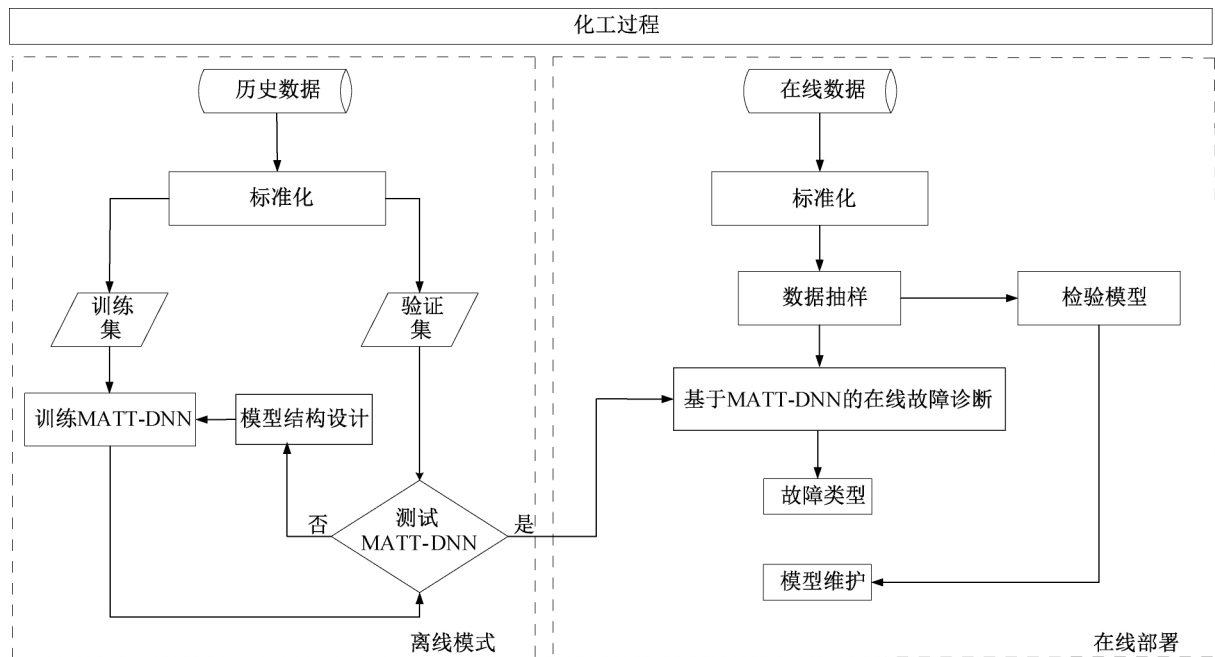


图 6 基于 MA-DNN 的故障诊断流程示意

## 2 实验仿真

### 2.1 TE 过程

TEP 是基于伊斯曼化学品公司真实的工业化过程所建立的仿真系统<sup>[15]</sup>。作为工业化领域的公开数据,TEP 生成的数据具有非线性、强耦合的特性,因而被广泛应用于各类故障检测和诊断算法的性能测试中。

TEP 主要有五个操作单元: 化学反应器、冷凝器、汽/液分离器、循环压缩机和汽提塔。过程中包含 A、B、C 和 E 四种原料以及 D、F、G 和 H 四种产物,更详尽的信息可参考文献<sup>[15]</sup>。为了

模拟真实的工业化工情况,TEP 采用图 7 所示的控制结构进行仿真,其 MATLAB 仿真程序可在 <http://depts.washington.edu/control/LARRY/TE/download.html> 中获取。TEP 中包含两类变量——操作变量和测量变量,其中测量变量 41 个、操作变量 12 个(其中一个操作变量未进行控制),每 3 min 采集一次数据。本文所使用的 TEP 数据总共有 21 种故障类型,每种故障的数据集都包括 480 组训练数据和 960 组测试数据,其中测试数据中前 160 组为正常数据<sup>[16]</sup>。此外,该数据集还包含了 500 组正常条件下的数据作为训练集的一部分。

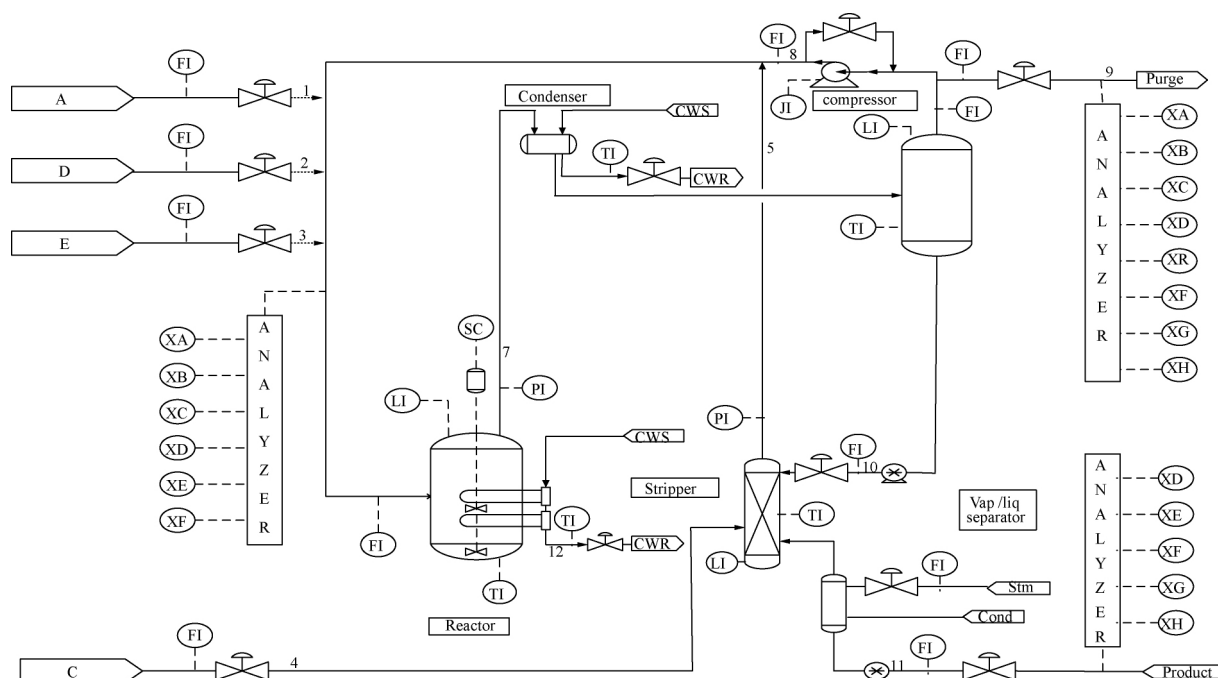


图 7 TEP 仿真结构框图

### 2.2 基于 TE 过程的故障诊断结果对比及分析

实验平台为配置 Intel i5-6500 3.20 GHz, 16 GB DDR4 RAM, 三星 750 EVO SSD, NVIDIA GTX 1060 的 PC 机。所有代码均基于 Python 语言在 Sklearn 和 Tensorflow 框架下完成。利用 CUDA 和 cudnn 提供的并行加速能力实现快速的训练和诊断任务。

为了验证本文提出的 MA-DNN 算法的性能,分别将逻辑回归(Logistics regression, LR)、SVM、K 最近邻(K-nearest neighbor, KNN)、朴素贝叶斯分类器(Naive Bayesian classifier, NB)、决策树(Decision tree, DT)、ANN 以及本文所提出的 MA-DNN 应用到 TEP 中进行测试,采用 F1 分数(F1 score)作为综合评价指标,其计算公式如下:

$$F_1 = 2 \times \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}} \quad (9)$$

其中: *precision* 是分类精确率, *recall* 是召回率。

MA-DNN 所使用的特征位置向量维度为 52 维,初始学习率为 0.001,优化算法采用 Adam 算法。本实验在每个模型的训练阶段都按 9:1 的比例分割数据得到验证集,利用验证集挑选最佳的模型参数,最后对每个模型都进行 10 次训练与测试,取 10 次平均 F1 分数作为最终结果,如表 1 所示。

从表 1 中可以看出,本文所提出的基于 MA-DNN 的故障诊断算法的平均 F1 分数达到 66%,与其他基于数据驱动的故障诊断算法相比优势明显,尤其在对故障 16 和 19 的故障诊断中,其所取得的 F1 分数相较于其他故障诊断算法提高超过 10%。同时,在总计 21 种不同类型的故障中,本文所提出的 MA-DNN 算法在其中 14 种故障上皆取得了最佳的 F1 分数,远优于其他算法。上述实验结果进

表1 7种方法在 TEP 故障数据集上的 F1 分数 %

故障类型	MA-DNN	LR	SVM	KNN	NB	DT	ANN
故障类型 1	100	97	98	97	97	96	97
故障类型 2	97	90	97	96	98	97	97
故障类型 3	24	15	16	12	14	17	20
故障类型 4	81	71	82	14	84	87	78
故障类型 5	96	62	93	19	10	53	64
故障类型 6	99	69	71	99	93	99	46
故障类型 7	100	100	100	68	95	99	100
故障类型 8	57	28	36	36	59	37	57
故障类型 9	14	12	17	10	9	15	12
故障类型 10	46	40	13	10	26	28	30
故障类型 11	63	16	20	15	63	52	60
故障类型 12	57	28	58	41	56	41	52
故障类型 13	43	34	40	37	22	23	40
故障类型 14	96	4	79	74	87	89	92
故障类型 15	22	13	20	10	21	19	20
故障类型 16	60	42	18	7	18	45	40
故障类型 17	86	73	79	63	72	76	82
故障类型 18	84	36	77	86	76	72	40
故障类型 19	79	24	55	45	65	54	60
故障类型 20	54	64	40	17	44	42	50
故障类型 21	18	6	3	5	38	43	9
平均	66	41	53	41	55	56	55

一步表明:与其他基于数据驱动的故障诊断相比,MA-DNN 通过引入深度神经网络加强了模型对特征间非线性关系的挖掘。同时,为防止深度神经网络对非敏感特征过度学习,MA-DNN 利用三种注意力机制突出重点信息,优化特征向量,从而进一步增强其故障诊断效果。

为了进一步探究本文所提出的三种注意力机制对 MA-DNN 故障诊断效果的贡献,在保证其他条件一致的情况下,消去其中一种注意力机制,从而研究该种注意力机制对结果的影响。实验设置均与上文设置相同,实验结果如表 2 所示。具体而言,本文设计了如下的由部分网络结构构成的模型:

a) ALL:同时使用三种注意力机制;

b) WTA(Without interactive attention):仅移除交互注意机制;

c) WLA(Without location attention):仅移除基于位置向量的自注意力机制;

d) WFA(Without feature attention):仅移除基于特征向量的自注意力机制。

表2 不同注意力机制故障诊断效果

模型	精确率/%	召回率/%	F1 分数/%
ALL	67	66	66
WTA	65	64	64
WLA	65	64	64
WFA	62	60	60

观察表 2 可知,三种注意力机制皆能提高模型的故障诊断能力,其中基于特征向量的自注意力机制对模型的故障诊断效果影响最大。通过基于特征向量的自注意力机制,模型可以更好地刻画当前输入下每个特征间的关系并将其融于当前的特征向量中,从而丰富模型所获取到的特征信息。对于故障诊断任务而言,基于特征向量的自注意力机制所获得的特征信息是最为丰富全面的,故当缺失这部分信息时,模型的故障诊断能力急剧下降。基于位置向量的自注意力机制和交互注意机制对模型的贡献大致相当,两者为模型提供了不同表示子空间中的特征信息,从而帮助模型更好地进行故障诊断任务。

### 3 结 论

本文提出一种基于多注意力机制的深度神经网络化工过程故障诊断算法,引入三种不同的注意力机制充分表征不同表示子空间内特征间的关系,并将其融于特征向量中,使得每个特征保留原有信息的基础上,也能获取该特征与其他特征间的联系与相互影响,从而使得深度神经网络能以多种形式学习化工过程数据的特征信息,取得更好的故障诊断效果。此外,注意力机制能借由注意力矩阵计算出相应的注意力特征,进而模型在训练阶段能关注重点特征,减少冗余和非敏感特征对模型的影响。该故障诊断算法具有优势明显的故障诊断效果,是较为前沿的工业故障诊断方案。

### 参考文献:

- [1] 齐咏生,张海利,高学金,等. 基于 KECA 的化工过程故障监测新方法[J]. 化工学报, 2016, 67(3): 1063-1069.
- [2] Cai B P, Huang L, Xie M. Bayesian networks in fault diagnosis [J]. IEEE Transactions on Industrial Informatics, 2017, 13(5): 2227-2240.
- [3] Zhu Z B, Song Z H. Fault diagnosis based on imbalance modified kernel Fisher discriminant analysis [J]. Chemical Engineering Research & Design, 2010, 88(8): 936-951.
- [4] Rato T, Reis M, Schmitt E, et al. A systematic comparison of PCA-based Statistical Process Monitoring methods for high-dimensional, time-dependent Processes[J]. AIChE Journal, 2016, 62(5): 1478-1493.
- [5] Yin G, Zhang Y T, Li Z N, et al. Online fault diagnosis method based on incremental support vector data description and extreme learning machine with

- incremental output structure [J]. *Neurocomputing*, 2014, 128: 224-231.
- [6] Eslamloueyan R. Designing a hierarchical neural network based on fuzzy clustering for fault diagnosis of the Tennessee-Eastman process [J]. *Applied Soft Computing*, 2011, 11(1): 1407-1415.
- [7] LeCun Y, Bengio Y, Hinton G. Deep learning [J]. *Nature*, 2015, 521(7553): 436-444.
- [8] 胡越, 罗东阳, 花奎, 等. 关于深度学习的综述与讨论 [J]. *智能系统学报*, 2019, 14(1): 1-19.
- [9] Mnih V, Heess N, Graves A, et al. Recurrent models of visual attention [C]//*Proceedings of the 27th International Conference on Neural Information Processing Systems: Volume 2*, Montreal, Cambridge: MIT Press, 2014: 2204-2212.
- [10] 邓远远, 沈炜. 基于注意力反馈机制的深度图像标注模型[J]. *浙江理工大学学报*, 2019, 41(2): 208-216.
- [11] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate[J]. *International Conference on Learning Representations*, 2016, 12(7): 366-381.
- [12] Vaswani A, Shazeer N, Parmar N, et al. Attention is all you need [C]//*Advances in Neural Information Processing Systems*. Cambridge: MIT Press, 2017: 5998-6008.
- [13] 施方迤, 汪子扬, 梁军. 基于半监督密集阶梯网络的工业故障识别[J]. *化工学报*, 2018, 69(7): 3083-3091.
- [14] He K M, Zhang X Y, Ren S Q, et al. Deep residual learning for image recognition [C]//2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, USA. New York: IEEE, 2016: 4651-4659.
- [15] Downs J J, Vogel E F. A plant-wide industrial process control problem [J]. *Computers & Chemical Engineering*, 1993, 17(3): 245-255.
- [16] Boldt F D A, Rauber T W, Varejao F M. Evaluation of the extreme learning machine for automatic fault diagnosis of the tennessee eastman chemical process [C]// *Conference of the IEEE Industrial Electronics Society*, Dallas, USA. New York: IEEE, 2014: 2551-2557.

(责任编辑:康 锋)