

概 述

1. 研究背景
2. 文章贡献点以及提出的方法
3. 预期结果和复现中遇到的问题

1. 研究背景

在工业中，预测和健康管理（prognostics and health management, PHM）被用于提高系统的可靠性和效率。剩余使用寿命（remaining useful life, RUL）预测是PHM的核心任务，它预测机械故障、预防事故并降低运营成本。通常，RUL预测是指使用来自多个传感器的时间序列数据（振动信号）来预测机器的剩余寿命。一般RUL预测包含两类方法，基于模型的方法和数据驱动的方法。基于模型的方法需要系统的专业知识，局限性较大；基于数据驱动的方法将机械系统视为一个黑匣子，利用传感器数据进行预测，更适用于复杂的机械系统。长短期记忆（long short-term memory, LSTM）在自然语言处理领域非常流行，它擅长捕捉和学习序列信息。

2. 文章贡献点以及提出的方法

贡献点：

1. 提出了手工特征流（**handcrafted feature flows, HFF**）来抑制原始输入信号中存在的噪声。利用HFF，可以容易地捕获序列信息并提高RUL预测精度。
2. 提出了一种基于**双LSTM的双流网络**。

提出的方法

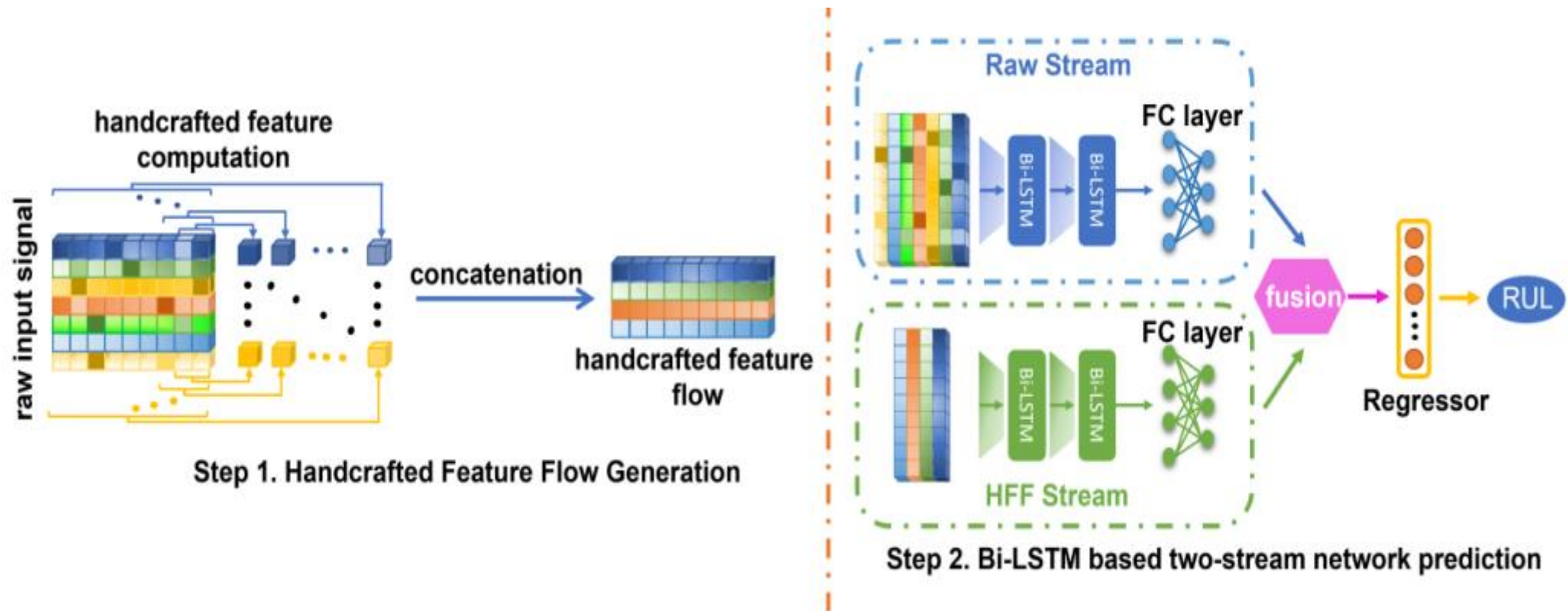


图1. 提出的方法

Step1. 提取振动信号HFF特征

Step2. 将原始特征和HFF特征分别输入到两个Bi-LSTM中，将全连接层输出结果进行融合，之后输入一个全连接层中得到输出结果。

振动信号HFF特征

1.振幅平均值

$$H_{avg}^c = F_{avg}(X^c, t) = \frac{1}{t} \sum_{i=1}^t X^c(i)$$

2.线性模型的趋势系数

$$\hat{y}(i) = w_0 + w_1x(i) \rightarrow F_{pf} = w_1$$

3. 最终的HFF

$$F_{hff}(X^c) = connect(F_{avg}(X^c, 2), F_{pf}(X^c, 2), F_{avg}(X^c, 3), F_{pf}(X^c, 3), \dots, F_{avg}(X^c, T), F_{pf}(X^c, T))$$

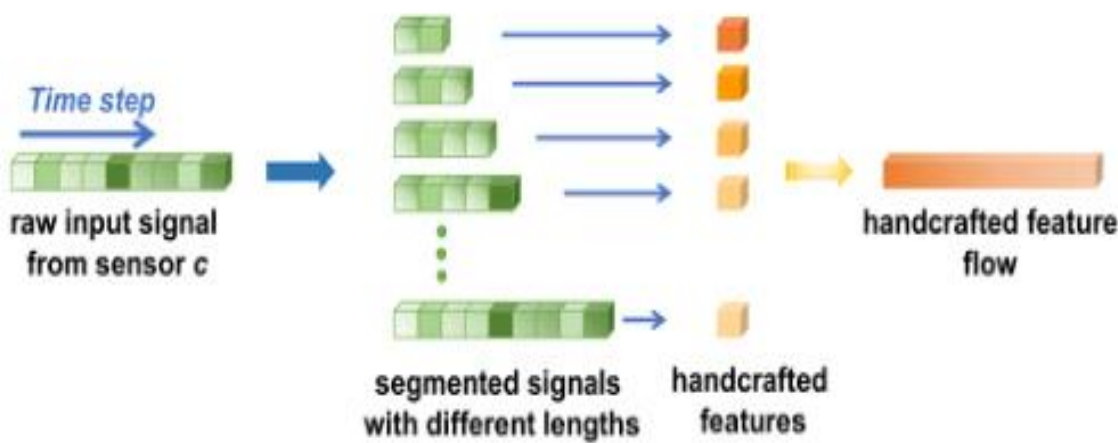


图2. HFF特征提取方法

Bi-LSTM

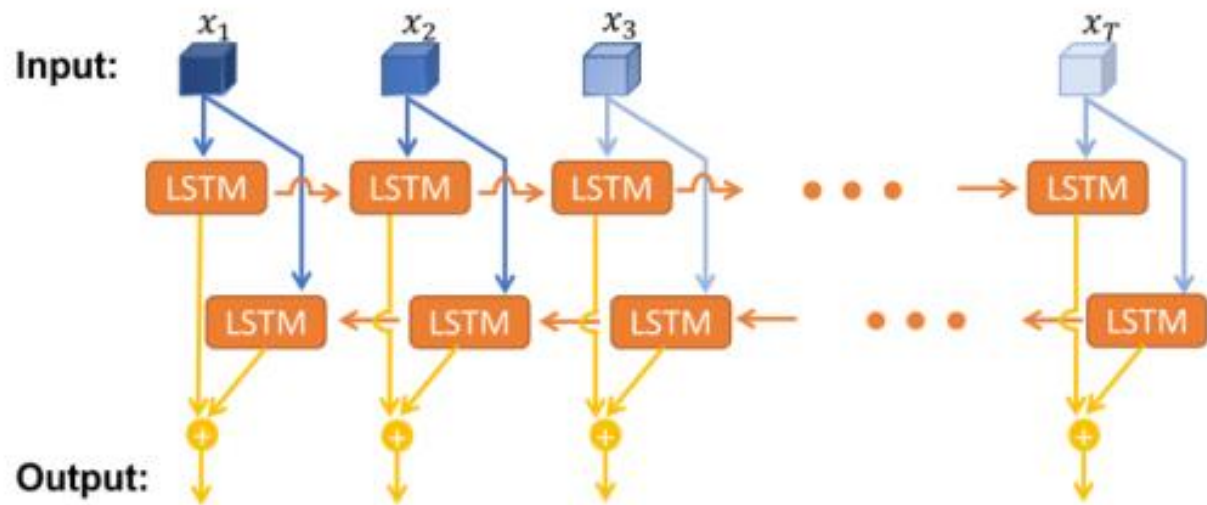


图3. Bi-LSTM

$$\begin{aligned} \vec{h}(t) &= f(\vec{x}(t), \vec{h}(t-1), w) \\ &= \begin{cases} \vec{i}(t) = \sigma(\vec{w}_{ii} \vec{x}(t) + \vec{w}_{hi} \vec{h}(t-1)) \\ \vec{f}(t) = \sigma(\vec{w}_{if} \vec{x}(t) + \vec{w}_{hf} \vec{h}(t-1)) \\ \vec{g}(t) = \tanh(\vec{w}_{ig} \vec{x}(t) + \vec{w}_{hg} \vec{h}(t-1)) \\ \vec{o}(t) = \sigma(\vec{w}_{io} \vec{x}(t) + \vec{w}_{ho} \vec{h}(t-1)) \\ \vec{c}(t) = \vec{f}(t) \odot \vec{c}(t-1) + \vec{i}(t) \odot \vec{g}(t) \\ \vec{h}(t) = \vec{o}(t) \odot \tanh(\vec{c}(t)) \end{cases} \end{aligned}$$

$$\begin{aligned} \overleftarrow{h}(t) &= f(\overleftarrow{x}(t), \overleftarrow{h}(t+1), w) \\ &= \begin{cases} \overleftarrow{i}(t) = \sigma(\overleftarrow{w}_{ii} \overleftarrow{x}(t) + \overleftarrow{w}_{hi} \overleftarrow{h}(t+1)) \\ \overleftarrow{f}(t) = \sigma(\overleftarrow{w}_{if} \overleftarrow{x}(t) + \overleftarrow{w}_{hf} \overleftarrow{h}(t+1)) \\ \overleftarrow{g}(t) = \tanh(\overleftarrow{w}_{ig} \overleftarrow{x}(t) + \overleftarrow{w}_{hg} \overleftarrow{h}(t+1)) \\ \overleftarrow{o}(t) = \sigma(\overleftarrow{w}_{io} \overleftarrow{x}(t) + \overleftarrow{w}_{ho} \overleftarrow{h}(t+1)) \\ \overleftarrow{c}(t) = \overleftarrow{f}(t) \odot \overleftarrow{c}(t+1) + \overleftarrow{i}(t) \odot \overleftarrow{g}(t) \\ \overleftarrow{h}(t) = \overleftarrow{o}(t) \odot \tanh(\overleftarrow{c}(t)). \end{cases} \end{aligned}$$

$$o_{\text{final}}(t) = \vec{o}(t) + \overleftarrow{o}(t).$$

融合方法

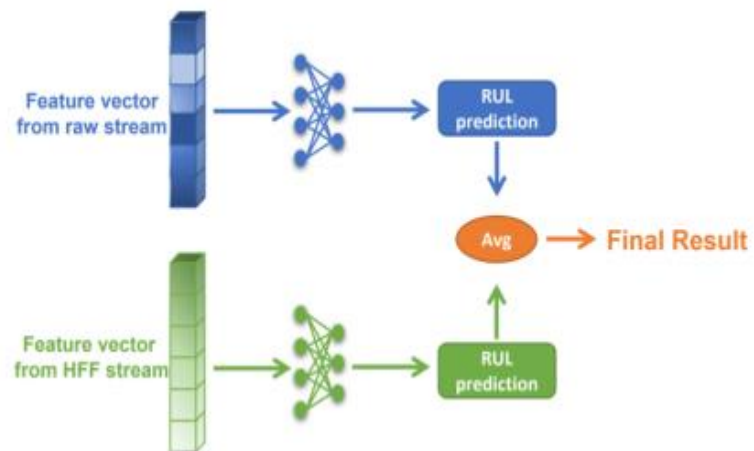


图4. 方法1

$$Y_0 = Avg \left(Y_r(f_r(X)) Y_h \left(f_h(\tilde{X}) \right) \right),$$

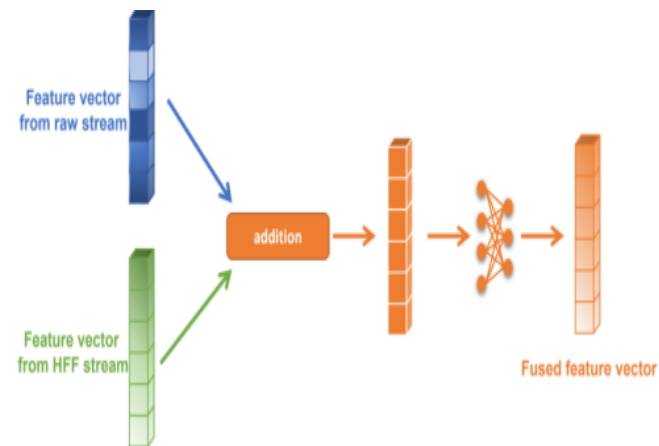


图5. 方法2

$$f_o = f_r(X) + f_h(\tilde{X})$$

3. 预期结果和复现中遇到的问题

实验设置：

使用（C-MAPSS）数据集，最大RUL = 125，选择了14个传感器数据（共21个）

网络结构：Bi-LSTM隐藏层2个，单元数分别为16和32，全连接层节点设置为16。

融合模块中全连接层节点数为8，全连接层使用ReLU激活函数

使用dropout, dropout rate=0.2。

在训练过程中，batch size为10，epoch为50

预期结果

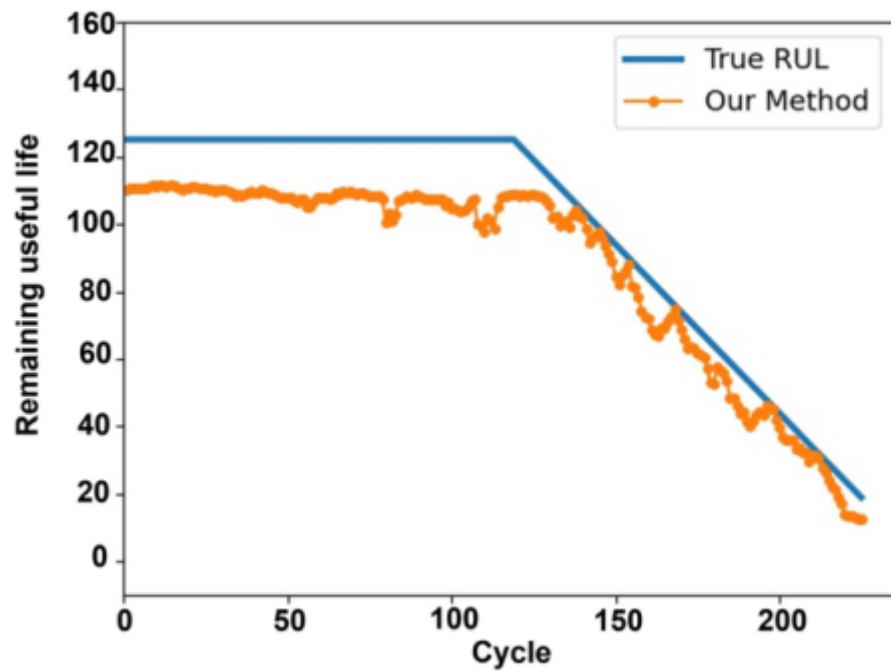


图7. FD001—RUL

遇到的问题

1. 网络架构方面
2. 最后训练的模型，没有得到完全相同的结果

Thank you