# Predicting Defective Engines using Convolutional Neural Networks on Temporal Vibration Signals

# 摘要

这篇文章首次解决了使用大量不平衡的结构性噪声的信号数据预测与发动机内部激活有关的损坏问题。提出在我们的时间输入信号上使用卷积神经网络，随后结合额外的静态特征。 在训练期间我们考虑了数据的不平衡使用信息量小的批次。 实验结果表明，在我们的大型实际用例中，检测少数类别的性能很好。

**关键词：预测，保持预测性，不平衡数据，深度学习**

# 简介

燃烧发动机变得越来越复杂，以满足各种标准。 为了保证顺利的装配过程和功能性发动机，每条生产线的末端都要进行高效的低成本低温试验。 在冷测试期间，发动机被电动机无源地驱动。 在这样做的过程中，为了确定发动机材料的特性（Cremer and Heckl，2013），测量了几种测量结果，包括结构载荷 - 在50 Hz或100 Hz的较低频率下的机械振动。 基于这些结构测量，我们工作的目标是预测测试发动机的耐久性运行结果：发动机是否功能完好或有缺陷？

然而，收集到的测量结果带来了各种挑战，使得预测任务非常困难：（1）从不同工艺步骤收集到的可用信号指的是不同的视图。在冷测试期间收集结构测量结果的同时，还会考虑其他视图，例如一般的发动机信息或环境特征。这种“多视角”数据具有不同的类型，其次是不同的统计分布和不同类型的不确定性（Sun，2013）。 （2）此外，收集到的信号的高维度构成了巨大的挑战：每个信号本身都代表一个时间序列。值得指出的是，这些时间序列的长度在信号之间不同。因此，它不能简单地被视为一个多元时间序列。 （3）最重要的是，数据遭受了高度不平衡的类别标签分布。也就是说，与缺陷类别相比，缺陷类别引擎的代表性很低，这使得学习相当困难。

近年来，由于深度学习技术的有效性，我们研究这项工作 - 使用来自真实世界用例的数据 - 这样的架构是否也能够从收集的**不平衡数据中预测有缺陷的引擎**。

# 相关工作

**深度学习在图像或视频识别领域的许多任务（Schmidhuber，2015; Karpathy等，2014）以及自然语言处理（Mikolov等，2013）中都取得了极高的性能。 同样，研究人员已经研究了深度神经网络架构和算法，以评估它们在时间序列任务中的表现 - 特别是以多元时间序列分类为重点的实际应用变得越来越重要，例如在医疗保健，人类活动识别或可再生能源 （Doucoure等，2016; Yang等，2015; Zheng等，2014; Borovykh等，2017）。**

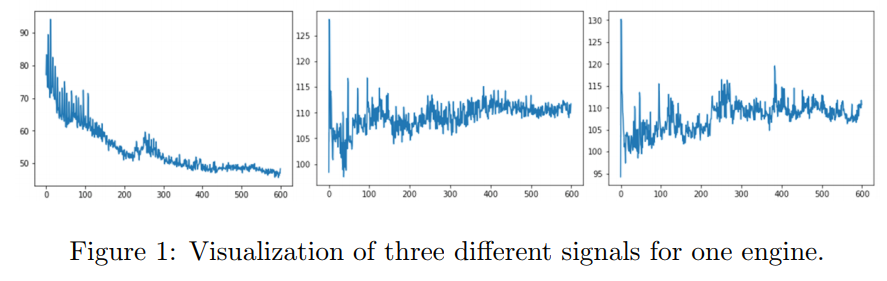
**由于神经网络能够从数据中提取非线性信息，因此它们很自然地适合时间序列，而这些时间序列很少只代表线性效应。 尽管第一眼看来，像RNNs / LSTMs这样的时序/序列神经网络模型似乎是时态数据分类的自然选择，但Kim（2014）的早期工作表明，简单的卷积神经网络（CNNs）是平等的或甚至是 更适合这样的任务。 考虑到这个想法，（Cui et al。，2016; Borovykh et al。，2017）已经表明，CNN不仅在经典的用例如人脸验证和音频分类方面表现出色，而且在诸如时间序列分类等任务中表现卓着。**

在这种情况下CNN成功的主要原因是由于它能够从局部信息中学习，并通过沿时间轴滑动卷积滤波器与更多特征相关联。 将卷积表示为该模型的主要优点，本文中我们将重点放在以下问题上：（a）具有高维度的现实世界多视图数据集，是一种CNN架构，能够预测受高故障影响的引擎 曲线下面积（AUC）？ （b）即使数据严重不平衡，CNNs能否表现良好？

**为了回答这些问题，我们调查CNN深度学习体系结构（第4节），以使用来自实际用例的数据预测有缺陷的引擎（第3节）。 我们研究了CNN的不同结构，包括不同的层数，在我们的实验研究中使用不同的池化函数和激活函数 - 以及数据的不同属性及其对结果的影响（第5节）**

# 数据描述

为了收集每个引擎的结构噪声，使用振动计来测量时域中不同的机械表面振动。 此外，测量发动机振动对气压的影响。 使用傅立叶变换，测量的信号在连续的频率域中被转换。 图1显示了一个引擎的3个测量信号的片段。 尽管x轴示出不同的频谱序列，但y轴呈现指向例如 振动信号或曲轴运动等。



形式上，结果数据是一组矩阵Si∈R N×Li，其中每个i∈M表示对所有实例测量的一个信号。 这里，N表示实例/引擎的数量，并且Li表示关于信号i的频谱的长度。 相应地，Si的第j行表示关于信号i收集用于引擎j∈N的频谱。 为了简单起见，我们用Xij表示这些行向量。 再次注意，对于每个信号，这些向量Li的长度可能不同。

此外，对于每个引擎，我们有一组非连续（环境）变量，这些变量由矩阵E∈R N×LE表示，其中ej表示第j行向量。 并且每个实例j∈N被分配给yj∈{0,1}的标签yj。 这里，yj = 1表示具有特定问题的实例的类别,而yj = 0表示相反的意思。 我们用N1表示所有标签为1的实例，N0表示标签为0的所有实例。

总的来说，对于我们的使用案例，我们有大约93,000个引擎。 对于每个发动机，考虑6个不同的基于序列的信号并补充7个环境特征。 有缺陷实例与无缺陷实例的比例约为1:32。

**值得注意的是，功能和缺陷实例的信号不容易区分，使得预测非常困难并且不可能被人类所预测。 同样值得注意的是，所示信号包括几个局部最小值和最大值的强烈波动。 通过在Si上使用CNN，我们的目标是利用这些信息来执行准确的预测。**

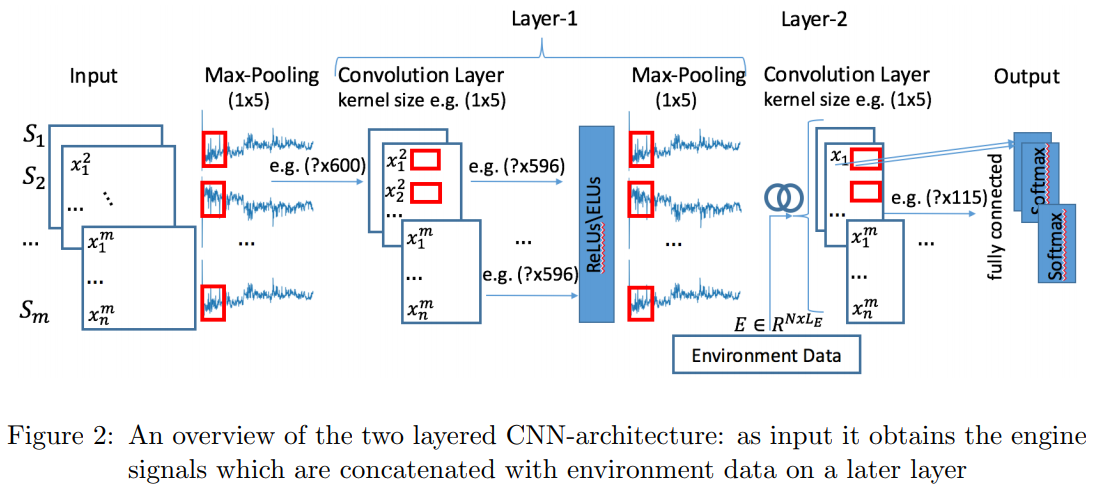
# 发动机振动信号的卷积神经网络

在下文中，我们将描述用于将发动机分类为发动机故障的深层CNN的结构。

## 4.1深度CNN结构

图2显示了我们提出的模型中使用的深度前馈神经网络的一般结构。

1. 首先，为了促进CNN层的训练并同时提高效率，在信号处理之后，我们首先通过应用（1x5）Max-Pooling对我们的信号进行下采样。 在这样做的过程中，我们逐渐减少了特征数量和计算复杂度的空间大小，同时保持模型中的位置灵敏度，因为池化的大小很小
2. 其次，我们使用多个CNN /池化层作为时间信号上的非线性可学习特征提取器。 更确切地说，我们结合了以下三个函数，我们在实验分析中调查了它们的变体：



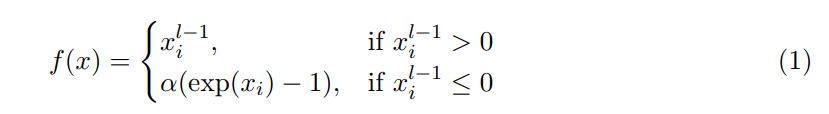
### 卷积层：

它在输入数据Si上卷积可学习的卷积核，以计算卷积核的条目和输入之间的点积。 对于每个信号（和层），我们学习不同的卷积核权重，因此考虑到数据的多视角特性。 因此，网络学习不同的卷积核，旨在检测特定位置的特定类型的特征。 为了从整个数据中提取最大的局部时间信息，限制滤波器的大小ml是有益的，其中l是相应的层，并且i是第i个信号。 注意，与例子图像相比，我们的滤波器是一维（即矢量）。

### 激活函数层

由卷积层生成的结果作为激活函数的输入给出，以从输入数据捕获一些非线性特征。 在我们的模型中，我们使用指数线性单位（ELU）激活函数（Clevert等，2015）。 虽然通常最常用的激活函数是整流线性单位（ReLUs）Nair和Hinton（2010），但事实证明，在我们的情景中，ELU更强大（参见实验研究）。 通过将负值映射到零，ReLU有助于解决密度较小时的解决方案。 然而，作为缺点，可以观察到死亡的ReLU现象 - 例如， 在前一层具有较大的负偏差项的情况下。

解决这个问题的一个尝试是指数线性单位（ELUs），定义为：



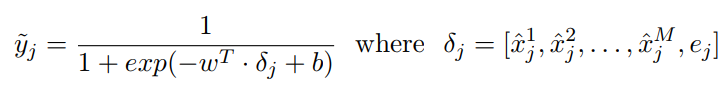
与ReLUs相反，ELU可以处理来自卷积层的特征映射的负值（Clevert et al。，2015）。 这通过α>0来实现。特征映射的负值将平均激活接近零，这使得能够更快地学习使渐变接近自然梯度。 当输入变小时，ELU饱和为负值。 这减少了变化，信息传播到下一层。 在第5节中分析了这两种功能的表现。

### （3）池化层：

最后，我们使用a）所选合并窗口内每个邻域内的平均值或b）最大值对来自上一层的输入的分辨率进行下采样。 直觉是，特征的确切位置并不重要，因为它相对于其他特征的粗略位置Zheng等（2014）。 请注意，最后的卷积层之后没有池化层。 我们在实验研究中分析了不同池窗口和卷积大小的影响

（C）最后，在执行隐藏层的多个CNN /池化层（对每个时间信号进行操作）之后，我们将输出与静态环境数据相结合，将它们共同馈送到完全连接的神经元中，然后通过softmax激活函数生成最终的分类决策。

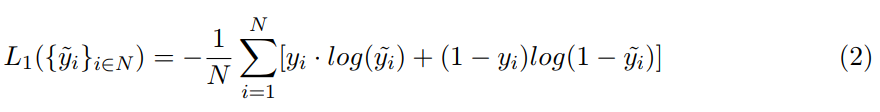
更准确地说，令∈R ki是最后一个CNN层的输出（对于发动机j和信号i），ej是发动机j的环境特征，故障发动机的预测概率由下式给出



因为我们在这里考虑一个两元问题，所以成为一个正确函数的引擎的概率就是1 -。

## 损失函数，类别失衡，批量学习

为了学习神经网络的参数，我们必须定义相应的损失函数 - 根据预测的类概率进行操作。 在（二元）分类任务中，可以使用交叉熵损失：



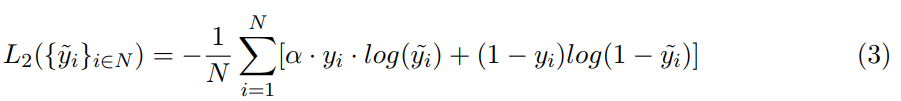
其中yi，分别表示实例i的真实标签和预测概率。 通过最小化损失 - 例如 使用梯度下降法 - 我们的目标是降低预测误差，希望尽可能保持测试集上的AUC。

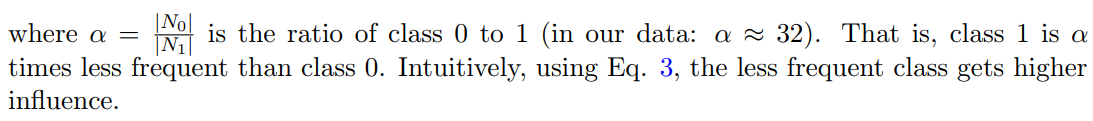
### 数据平衡问题

然而，在本文中分析的发动机数据集受到类别不平衡问题的影响（Japkowicz和Stephen，2002），其中肯定实例的数量（即本例中发生故障的发动机数量）明显少于非损坏引擎。当来自一个类的实例的数量比在另一个班级的实例的数量多很多时，来自训练数据的学习过程受到不利影响。由于这个问题的重要性，许多现有的研究工作已经解决了这个问题（Bermejo et al。，2011; Mazurowski et al。，2008）。总的来说，这些方法可以分为三大类（Galar et al。，2012）：（a）算法级方法试图修改学习算法，使学习偏向于把偶数班作为少数人。此处需要先知道为什么当算法的算法的学习行为失败时。受损发动机的类别是少数。 （b）数据级方法通过重新取样偶数级和非偶数级来重新平衡数据集。平等分配（Stefanowski和Wilk，2008）。 （c）成本敏感学习是前两个版本a）和b）的组合。虽然重点是将不同实例的成本分配给在少数群体中优先考虑他们，分类器通过最小化例如偏向于平等阶级的偏向。分类中使用的损失函数的结果（Maldonado等，2014; Gunnemann和Pfeffer，2017）。

在本文中，我们通过结合使用数据级和成本敏感的方法来解决阶级失衡问题。 更确切地说，我们使用随机学习（小批量）不仅加快了学习过程，而且处理了不平衡数据。

首先，我们旨在最大限度地减少以下损失函数





频率出现较少的类别会获得更大的权重。

其次，我们使用信息量小的批次使用随机梯度下降来最小化损失函数。 不是随机选择小批次的实例，而是每个小批次B包含所有实例N1以及来自相同比例的N0的实例的随机子集。 因此，批次B对少数群体“偏向”（与原始数据分布相比）。 如果少数群体在总体数据中非常小，这是特别有用的，因为在小批次中选择这些实例的可能性很小。

使用信息批量B，现在基于损失L1（不是L2）计算随机梯度下降步骤：L1（{yi}i∈B）。 类不平衡因子α已经被小批量吸收。 因此，总体而言，我们不仅处理类不平衡问题，而且还通过训练我们的CNN随机样本来加速学习过程。

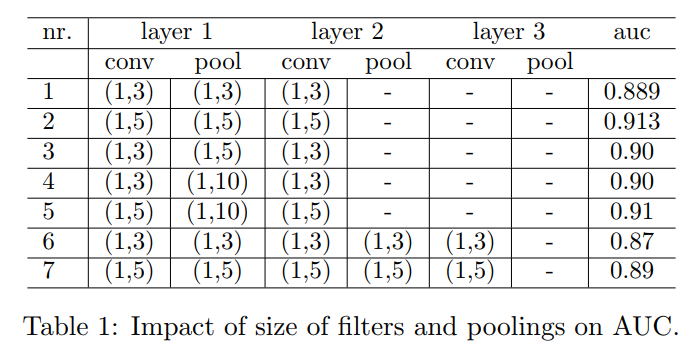
# 实验结果和讨论

正如在第3节中提到的那样，提出的CNN网络的性能是在由多个信号组成的具有天生噪声数据集上的结构进行测试的，每个信号又分别代表92890个频谱矢量。 在我们的例子中，每个向量呈现映射到二进制标签的引擎。 在我们的数据集上使用建议的CNN体系结构，我们的目标是将两组受损的引擎相互分开，并且AUC较高。 现在我们将更详细地分析我们的模型。 所有实验均使用80:20的比例将训练集和测试集分进开，并且使用ADAM优化器进行学习直至收敛。 所有数字都表示测试集的结果。 该模型 TensorFlow中实现，并在K80 GPU上执行，处理6个内核和56GB内存。

## 5.1 参数分析

我们首先评估我们的模型在改变架构和学习方法的不同参数时的性能。

窗口大小和层数。 表1总结了在我们的模型中使用不同大小的卷积滤波器或池化窗口产生的不同AUC - 以及架构中使用的层数（参见图2）。 如上所述，除非模型的最后一层，否则卷积层之后是池化层。 为了公平比较，我们对所有体系结构使用相同的激活函数（ELU），汇集函数（Max函数）和学习率0.0001。



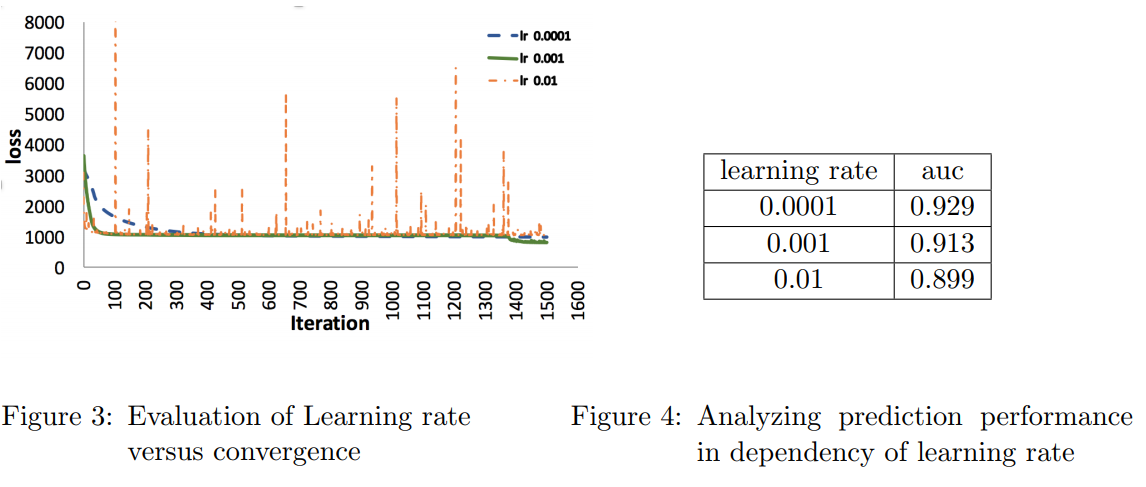
对于卷积层，我们考虑尺寸为（1x3）和（1x5）的两个滤波器，因为任何更大的尺寸都会导致时域中的信息丢失。 除此之外，我们还分析了大小（1×3）,（1×5）和（1×10）的最大池化窗口对AUC值的影响。 考虑到2层体系结构的结果，很明显使用大小为（1x5）卷积核使用池化层窗口大小为（1X10）导致最佳结果，其中AUC值略微下降。

考虑到3层架构的结果，没有取得更好的结果。 一个原因可能是参数数量较多，这使得学习方法更加困难，并且可能导致过度拟合。 因此，在下面的实验中，我们将重点放在窗口大小为（1x5）的两层体系结构上进行池化和过滤。

## 学习速度和融合

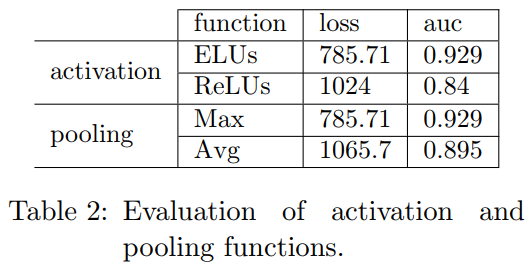
我们还根据学习速率参数评估了我们模型的AUC。 根据图3的结果，我们可以看出，通过将由lr表示的学习率的值设置为0.01，损失值不稳定地波动

然而，通过降低学习速率参数，可以获得连续减少的损失。 几种选择的学习速率的AUC值示于图4中，其中0.001达到最高的AUC值（并且同时有效），因此也在下文中使用。



池化和激活功能。 在图5和图6中，我们探讨了两种汇集函数max-pooling和averagepooling对AUC值以及激活函数ELU和ReLU的影响。 在图5中，显而易见的是，通过使用max-pooling函数，实现了更低的损失。 表2中的AUC结果证实，这种较低的损失反映在较好的AUC值中。 值得一提的是，两种情况下使用的激活函数都是ELU。 探索ReLU对AUC值与ELU的影响显示在图6以及激活行下的表2中。 再次，在这里我们可以得出结论，ELU结合max-pooling返回最佳结果。

总的来说，我们对现实世界数据集的实验分析表明，包含ELU激活函数和Max池的双层CNN架构可以获得最佳的AUC结果。



## 5.2 数据属性对混淆矩阵的影响

在工业领域，预测任务要达到高精度和召回少数类别至关重要 - 因为我们的主要目标是避免故障和高成本。 由于数据不平衡，比如在我们的用例中，使得这个目标很难实现通常会出现误报（即功能引擎将被预测为有缺陷）。

正如4.2节所讨论的那样，我们使用信息量小的批次来处理不平衡的数据，从而实现了公式 3.以下我们比较考虑或忽略不平衡方面的分类性能（有效地设置方程3中的α= 1）。

图7（左）通过考虑相应的召回精度曲线来显示结果。 虽然x轴显示召回值，但在真实损坏的实例集中呈现真正的比例，y轴指的是精确度。 红线表示我们的原则，而蓝线表示忽略不平衡的结果。 如清楚所示，即使对于较大的召回值，我们的小信息批次数据也可以获得更高的精确度。

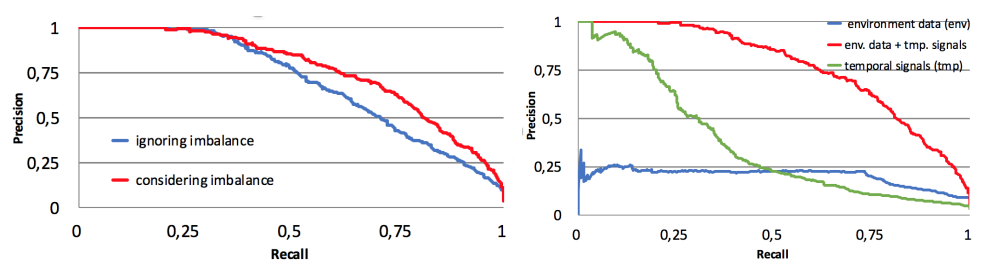


图7比较精度 - 召回曲线上的分类性能：左）考虑类别不平衡（红线）时的预测明显优于忽略类别不平衡（蓝线）时的预测。 右图）不同数据源对预测性能的影响。

考虑到红线，我们可以观察到，对于大约0.75-0.80的高召回值，我们明显优于蓝线 - 考虑到不平衡改善了结果。 与1:32的原始类别标签分布相比，即使对于高召回值，我们也可以达到1：1左右的水平（精度> 0.5） - 这是显着的改进。 同样值得注意的是，我们的方法并不是简单地导致所有实例都被预测为大多数类的简单解决方案 - 这是数据不平衡的常见问题。

多视图数据。 我们的数据集的另一个重要属性是数据源自不同的来源。 总的来说，我们有两个主要的数据源，一个与冷测试结果（时间数据）有关，另一个与（静态）环境数据相关，包括发动机型号或气缸数等特征。 为了检查每个来源对预测性能的相关性，我们分别分析了每个来源。 图7（右）显示了每个数据集对应的召回精度曲线。

考虑到仅用于环境数据的线条，清楚的是，分类器仅略微区分类别。 在时间数据的情况下，只有小的召回值才能达到高精度; 随后快速下降。 但是，显而易见，这两个来源涵盖了不同的组合：通过将它们结合起来（我们的方法;红线），我们明显优于各个来源。 我们提出的架构以有效的方式实现了这种组合。

# 总结

在本文中，现有技术的卷积神经网络被用于处理从现实世界的用例中预测有缺陷的发动机。 该数据包含多视图信息并且高度不平衡。 为了预测有缺陷的发动机，我们使用了一个在时间信号上运行的双层CNN架构，包括对不同激活和汇集功能的不同尝试以达到最佳结果。 为了处理不平衡问题，我们使用了数据级别和成本敏感方法的组合。 实验结果表明，这种方法导致更好的召回和精确结果

虽然本文首次分析了基于结构噪声测量的发动机分类极具挑战性的情景，但未来的工作仍有多个方向。 首先，我们旨在分析集合方法，以进一步提高精度并处理类不平衡。 此外，将我们的CNN体系结构与不同的顺序模型结合起来是有意义的。 Gunnemann等（2014a，b）的目标是捕捉进一步的依赖关系。