

抗震韧性城市评价方法研究

摘要

评价结构在地震下的结构响应非常重要，本文提出了一种使用卷积神经网络模块的循环神经网络 FesNet，用于结构地震时程分析，模型预测地震下结构所有有限元单元结点的加速度响应时程。通过将大量地震加速度分别作用于三层和五层钢筋混凝土框架的有限元模型，以数值计算结果作为数据集，完成对模型的训练和测试。结果表明，FesNet 能够以较高精度和效率快速得到结构的地震结构响应，具有一定的泛化性，有利于抗震评价和结构设计。

关键词：深度学习，循环神经网络，有限元单元法，结构地震响应

装

订

线

Study on Urban Evaluation Method of Seismic Resilience

ABSTRACT

It is very important to evaluate the structural response of the structure under earthquake. In this paper, a cyclic neural network FESNET using the convolutional neural network module is proposed for structural seismic time-history analysis. The model predicts the acceleration response time history of the finite element node of the structure under earthquake. A large number of seismic accelerations were applied to the finite element models of three-story and five-story reinforced concrete frames respectively, and the numerical calculation results were taken as the data set to complete the training and testing of the models. The results show that FESNET can quickly obtain the seismic response of the structure with high accuracy and efficiency, and has certain generalization, which is beneficial to seismic evaluation and structural design.

Key words: deep learning, recurrent neural network, finite element method, seismic structure response

装

订

线

目录

1 引言	1
1.1 课题背景与意义	1
1.2 结构抗震评价研究现状	1
1.3 主要研究内容	3
2 理论部分	5
2.1 神经网络	5
2.2 卷积神经网络	6
2.2.1 卷积运算	6
2.2.2 卷积层	6
2.2.3 池化	7
2.2.4 全连接层	8
2.2.5 Dropout 层	8
2.3 循环神经网络	8
2.3.1 门控循环单元神经网络	9
2.4 表示学习	11
2.5 迁移学习	12
3 实验部分	14
3.1 实验准备	14
3.1.1 结构概况	14
3.1.2 地震动选取与计算结果	15
3.1.3 数据集生成	16
3.2 算法设计	17
3.2.1 预处理	19
3.2.2 编码器-解码器	19
3.2.3 堆叠 GRU	20
4 结果和讨论	21
4.1 实验结果	21
4.1.1 实验环境	21
4.1.2 训练过程	21
4.2 测试与验证	22
4.2.1 时程分析	23
4.2.2 五层框架上的迁移学习	24
5 结论和展望	26
5.1 结论	26
5.2 展望	26
参考文献	28
谢 辞	28

1 引言

1.1 课题背景与意义

中国是世界上地震灾害最严重的国家之一，地震所造成的经济损失也十分巨大，由于地震对建筑结构造成的伤害是灾难性的，预测地震下的结构响应，对于损害评估方法来说必不可少。^[1-2]，结构响应预测的核心是给出结构模型，快速准确地预测给定地震动下的模型结构响应，但以往的预测手段很难兼顾准确性和效率。地震动时程具有量值大，强非线性，非周期性，非平稳性等特点^[2]，且建筑物在不同地震动作用下的结构响应差异较大，这就给地震预测方法带来了挑战，建筑结构响应预测依然非常困难，先前的研究仍然存在耗时较长，分析复杂，准确性欠缺等缺陷^[3]，因此，建立准确高效的结构地震响应预测算法是非常必要的。

当前地震的结构响应计算方法可分为两大类：一种方法是时程分析法，对结构及构件进行建模后使用基于一些假设的优化改进的积分方法如 ABAQUS 使用的 Newmark^[4]等近似数值计算；另一种方法则是通过其他途径避开时程分析中难以处理的恢复力模型，积分迭代方法等，采用其他计算或仿真工具估计结构地震响应。但现有方法往往均无法兼顾准确性和效率。无论是为了减小地震灾害给建筑结构造成的损失，在灾后迅速对结构破坏情况进行评估，还是用于结构设计时的抗震评价分析，在现有包含大量地震动的数据库基础上，建立快速高效的结构响应预测算法是非常必要。根据通用近似定理，神经网络能拟合任意非线性函数^[5]，即使是在具有在模糊因果关系的现实问题中下，也往往能够依据大量训练数据，来合理完成输入输出映射关系建模的强大能力，且往往具有预测时非常快速的求解速度，神经网络的这些性质因此特别适合解决工程和现实生活中许多非线性问题。近年来经过不断发展和研究，得到了一定程度的认可和重视，成为了研究者用于结构抗震计算的重要工具之一。

使用神经网络等深度学习方法对建筑结构地震响应预测具有一些显著的优点。第一，基于深度学习的端到端方法能够使用建筑结构先前时刻的动力响应和地震加速度输入来直接对未来时刻的建筑结构响应进行预测，且这一运算过程往往具有较快的计算速度。第二，对结构的抗震性能设计具有一定的辅助作用。第三，能够紧密的与智慧城市和结构健康检测等技术结合，具有一定的工程应用和研究意义^[6]。

因此，本研究的基本目标是建立一个尽可能兼顾效率与准确性的可迁移深度学习模型算法。该算法的核心为包含卷积神经网络模块和门控循环神经网络（Gate Recurrent Unit，GRU）模块的自编码器结构，最终实现将输入的地面运动时间序列数据映射到结构有限元节点的加速度响应序列，得到完整地震时程分析。

1.2 结构抗震评价研究现状

建筑结构的动力非线性计算对于结构设计和健康监测至关重要^[7]。各种数值方法已成功用于结构抗震分析，如有限单元法（Finite Element Method）^[8]。然而，上述基于力学理论的数值分析方法由于迭代求解过程，不可避免地效率偏低且往往需要大量耗时。近年来，已有研究表明使用深度学习（Deep Learning）技术，可以加快通过复杂迭代运算方法求解偏微分方程的过程，通过建立多层神经网络来拟合非线性算子^[9]。因此，深度学习方法在结构动力分析中应用前景广

阔。

目前大量基于神经网络的结构地震响应预测模型已经被国内外学者提出，人工神经网络模型及其各种变种被用来对结构在静态或动态荷载下的响应进行建模。

高经纬等提出了一种基于长短时记忆网络（Long Short Term Memory Neural Network, LSTM）的深度学习框架来预测地震响应，具体目测目标为该模型顶层位移，研究了某框架-剪力墙模型，使用三条地震加速度作为数据集，训练并与试验数据进行对比^[10]。

Morfidis, K 等人从逼近未知函数和模式识别角度，研究了使用人工神经网络预测钢筋混凝土建筑地震损伤的可行性，数据集为 30 个具有不同特征的钢筋混凝土建筑物，和 65 次实际地震动。以通过有限元软件进行非线性时程分析得到的结构最大层间位移角作为建筑物的地震损伤指标，并研究了人工神经网络超参数对于训练速度和预测精度的影响，最后，设计实验研究了在三种工况下神经网络的泛化能力^[11]。

Peng 研究了深度学习技术来分析和预测土木工程中结构在周期性，冲击性地震荷载下非线性动力问题的方法，分别使用分段线性最小二乘法（PLLS），全连接前馈神经网络（Forward Connected Neural Network, FCNN）和长短时神经网络（进行了实验，验证了基于自回归和时间序列估计模型的有效性和泛化性，数据集包括使用有限元方法分析的六层钢框架的动力响应。结果显示基于自回归模型的分段线性最小二乘法和全连接前馈神经网络方法的性能不如基于时间序列估计模型的长短时记忆神经网络方法，并且具有较大的预测峰值均方误差。此外，分段线性最小二乘法方法对噪声敏感，但基于深度学习的全连接前馈神经网络和长短时记忆神经网络方法具有很强的鲁棒性和抗噪声性能。这些揭示了深度学习方法在土木工程结构行为分析中应用的可行性^[12]。

Perez-Ramirez 等通过集成循环神经网络模型（Recurrent Neural Network），使用贝叶斯方法进行训练，提出一种用于大型建筑结构的精确结构地震响应预测的方法。它使用基于概率的贝叶斯训练，来对神经网络模型中神经元的最佳数量等超参数进行优化。模型最终应用于比例为 1:20 的 38 层高层建筑和 5 层钢框架，分别在其上完成了训练和测试^[13]。

Azimi 提出基于卷积神经网络（Convolution Neural Network）的结构健康监测方法，并结合了迁移学习技术（Transfer Learning）通过压缩历史加速度响应，作为卷积神经网络的输入，对结构损伤情况进行预测，这一算法同样支持在现实大规模系统基于传感器进行的结的损伤识别和定位^[14]。

Zhang 开发了一种物理学指导的深层卷积神经网络（PhyCNN），用于结构地震响应估计。输入为来自模拟或感测得地震输入，输出为楼层位移等指标，使用例如动力学定律等物理条件为网络输出提供约束，以减轻过拟合以及对大型训练数据集的需求，提高模型的鲁棒性，进而进行更可靠的预测。实验结果表明，基于物理学指导得卷积神经网络得能够以数据驱动的方式准确预测建筑物的地震响应，而无需基于物理的分析/数值模型。PhyCNN 范式也优于非物理引导的神经网络，能够最大限度地利用有限的数据集^[15]。

Kwan 等模拟了大量人工地震，搭建神经网络建立了基于结构相关性的建筑物地震响应预测模型，输入最大地面加速度和最大地面加速度，将最大层间漂移率和最大位移作为输出层，实验数据集包括具有不同自然周期的四自由度（MDOF）结构，在其上进行地震响应预测并通过

数值模拟证实了所提出模型的有效性^[16]。

Liang 使用深度学习模型来求解主动脉的应力分布，输入有限元分析，直接输出主动脉壁的应力分布，从而绕过了有限元计算^[17]。

Lu 等根据通用逼近定理，具有单个隐藏层神经网络可以准确逼近任何非线性连续算子，因此深层神经网络具有从散乱数据流中学习连续运算符或复杂系统得潜力，等提出了 DeepONets，并证明其可以学习各种显式运算法如积分，拉普拉斯算子^[9]。

Xu 等使用长短期记忆神经网络建立于区域建筑地震损失评估，输入为地震动，输出为区域损害指标（HAZUS-MH 损坏状态类别）。这项研究提出了一个基于长期短期记忆（LSTM）神经网络架构的实时区域地震破坏评估框架。提议的框架不是专门为单个结构类型设计的，而是在区域范围内提供快速估计。该框架围绕工作流程构建，该工作流程通过特定区域的模型在地面运动和结构损坏之间建立高性能映射规则。该工作流程包括三个主要部分，即特定区域的数据库生成、LSTM 模型训练和验证以及模型用于损伤预测的利用。各种 LSTM 架构的影响，系统地分析了超参数选择和数据集重采样程序。作为已建立框架的测试平台，在清华大学校园建筑上进行了案例研究。结果表明，开发的 LSTM 框架可以在区域范围内实时进行损伤评估，具有较高的预测精度和可以接受的方差^[18]。

综上所述，目前结构地震响应预测模型按照其预测目标主要分为两类：第一类模型偏向于损害评估，其预测结果是结构的关键参数，包括结构的顶层的最大位移、每层的最大层间位移角、最大基底剪力以及结构整体是否破坏、失效等指标，从而直接给出结构在给定地震动下的损坏情况；第二类模型偏重于时程分析，模型的预测结果主要是结构每层和顶部的加速度或位移的时程结果，比如预测结构在完整地震过程中顶层的加速度时程曲线，更具有可信度^[9]。在结构地震响应的预测模型输入数据上而言，第一类模型的输入主要是结构的尺寸、材料以及地震动信息等数个非时间序列的特征，当然也有将结构时程通过融合卷积的手段进行输入，但总体而言还是从特征的角度出发；而第二类模型的输入则主要是地震动的时间序列数据。调研的参考文献中主要采用以神经网络为基础，通过大量数据预测进行建立预测模型，这充分利用了神经网络这种深度学习技术的优势，也是本课题的出发点之一。

目前，神经网络更多被用在计算机视觉，自然语言处理，模式识别等领域，但其一系列特性同样适用于地震下建筑结构响应问题，然而先前研究所提出的算法布置到新的结构上时需要大量数据进行重新训练，这无疑提高了应用成本和可行性。针对以上问题，本文提出一种基于有限单元法的深度学习模型，用于结构地震响应预测。通过提取结构有限元模型的在多种地震动作用下的时程加速度值，利用有限元软件 ABAQUS 进行时程分析构建数据集，进行实验对模型参数进行优化，最后将其用于五层框架，并与试验数据进行对比，证明该方法的预测值与实测值拟合较好，泛化能力强，速度较快，且具备一定程度的可重用性，是一种可行的预测方法。

1.3 主要研究内容

基于上述课题背景与国内外研究现状，针对已有研究的不足，本文考虑从求解有限单元法的结构动力离散偏微分方程角度出发，提出了一种基于深度学习的神经网络模型 FesNet 用于地

震响应预测，能可自回归地向前计算得到结构所有结点的地震时程响应曲线。通过 PEER 地面运动数据库（PEER Ground Motion Database）获得的地震动数据，导入 ABAQUS 有限元计算软件，并对建筑结构建模，得使用有限单元法进行数值计算得到时程分析的结果。再从 ABAQUS 接口中导出建筑结构，基于有限元方法划分网格所得的刚度和质量矩阵，并导出结构在所用地震动数据下数百个单元结点的加速度响应，这些数据经过处理，输入模型中，完成对所设计模型的训练，优化，分析，测试。最后将其用于五层框架的地震响应预测，并与有限元计算结果进行对比，以验证其有效性。本文的章节安排如下：

第一章：绪论

阐述了本文的研究背景与意义，并对结构的地震响应预测方法，以及神经网络求解偏微分方程和用于时间序列分析的循环神经网络模型的国内外研究现状进行了总结和分析，提出了现有模型存在的不足，并对本文的研究思路进行了概括。

第二章：理论部分

阐述了本文所使用的深度学习技术的理论基础，介绍了神经网络和其训练方法，介绍了使用卷积神经网络和循环神经网络分别对矩阵和时间序列进行建模的理论依据，并对本文所提出算法的原理进行了简单阐述。

第三章：实验部分

介绍了本文所使用的三层和五层框架的基本情况，并对其 ABAQUS 有限元对建筑结构建模的参数细节以及在计算过程中所选用的地震动信息进行了说明，阐述了如何通过有限元软件 ABAQUS 进行抗震时程分析得到本文模型训练所用数据集，以及一些必要的数据处理。并详细介绍了本文所提出模型 FesNet 的设计思路与内部结构。

第四部分：结果与讨论

阐述了模型训练和测试的过程与结果，将其在另一五层框架上进行迁移学习以验证其泛化性能。分别与有限元真实计算结果计算平方差损失和置信度作为性能指标。同时以某一地震动数据作为示例，自回归地进行前向预测，得到其可视化结果，并与有限元计算结果进行对比。

2 理论部分

2.1 神经网络

人工神经网络 (Artificial Neural Network, ANN) 是指一系列受生物学和神经科学启发的数学模型机器变种。这些模型共同点是通过线性代数和概率论思想, 并借助对人脑的神经元网络的抽象, 并按照一定拓扑结构来构建人工神经元之间的前后, 上下连接, 通过运筹学和凸优化等理论进行训练, 通过大量参数来构建某种映射关系的近似或者拟合, 这种函数关系在现实问题中往往是非线性的。在人工智能领域, 人工神经网络也常常被简称为神经网络 (Neural Network, NN) 或神经模型 (Neural Model)。早期的神经网络模型更强调结构的生物学合理 (Biological Plausibility), 但随着深度学习技术的不断实践和研究的不断深入, 发展为更关注对某种特定任务的建模, 比如物体识别、机器翻译、摘要生成等^[19]。特别地, 在引入误差反向传播算法来训练其参数, 增强其学习能力后, 神经网络在各种机器学习任务得到了越来越多的应用。随着训练数据量的增大特别是随着算力的提升, 神经网络在很多人工智能任务上已经取得了很大的突破, 特别是在语音、图像等感知信号的处理上, 神经网络表现出了卓越的学习能力。

一个神经元节点接受 D 个输入 x_1, x_2, \dots, x_D , 令向量 $\mathbf{x} = [x_1; x_2; \dots; x_D]$ 表示该次输入, 传入的输入经过线性变换, 如式 2.1 所示,

$$z = \sum_{d=1}^D w_d x_d + b = \mathbf{w}^T \mathbf{x} + b \quad (2.1)$$

其中 $\mathbf{w} = [w_1; w_2; \dots; w_D] \in \mathbb{R}^D$ 是 D 维的权重向量 (Weight), $z \in \mathbb{R}$ 表示一个神经元所获得输入信号 \mathbf{x} 的加权和, $b \in \mathbb{R}$ 是偏置向量 (Bias)。

之后 z 需输入非线性激活函数 (Activation Function) $f(\cdot)$, 得到该神经元的输出 (Activation) a :

$$a = f(z) \quad (2.2)$$

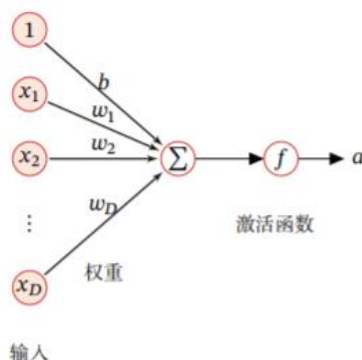


图 2.1 典型神经元结构

训练神经网络的过程就是在训练集上使用梯度下降算法更新网络参数, 具体而言: 每次迭代:

- (1) 前向传播: 逐层计算输出, 以本层输出作为下一层输入, 最终得到预测结果。
- (2) 计算损失: 计算预测结果与真实值的损失。
- (3) 反向传播: 通过链式求导法则逐层反传计算损失。

(4) 更新参数：根据损失，使用梯度下降法更新参数。

2.2 卷积神经网络

卷积神经网络（Convolution Neural Network）是一种带有卷积层的深层前馈神经网络，具有局部连接，权重共享等特性^[20]。通常用于分类和计算机视觉如图像和视频分析任务(如目标检测，识别，语义分割，追踪等)上^[21]。卷积神经网络拥有可学习的各种权重和偏差，对输入张量的各个对象分配重要性，从某种意义上卷积神经网络是表示学习的一种，与其他机器学习方法对比，卷积神经网络对预处理要求要低，有能力自动学习特征。近年来卷积神经网络也广泛地应用到自然语言处理，推荐算法等领域^[22]。

2.2.1 卷积运算

卷积（Convolution），也叫褶积，是数学中一种重要的运算。在信号处理或图像处理中，经常使用一维或二维卷积。给定一个图像 $\mathbf{X} \in \mathbb{R}^{M \times N}$ 和一个滤波器 $\mathbf{W} \in \mathbb{R}^{U \times V}$ ，一般 $U \ll M, V \ll N$ ，其卷积为

$$y_{ij} = \sum_{u=1}^U \sum_{v=1}^V w_{uv} x_{i-u+1, j-v+1} \quad (2.3)$$

为了简单起见,这里假设卷积的输出 y_{ij} 的下标 (i, j) 从 (U, V) 开始。

输入信息 \mathbf{X} 和滤波器 \mathbf{W} 的二维卷积定义为

$$\mathbf{Y} = \mathbf{W} * \mathbf{X} \quad (2.4)$$

其中 $*$ 表示二维卷积运算。图给出了二维卷积示例。

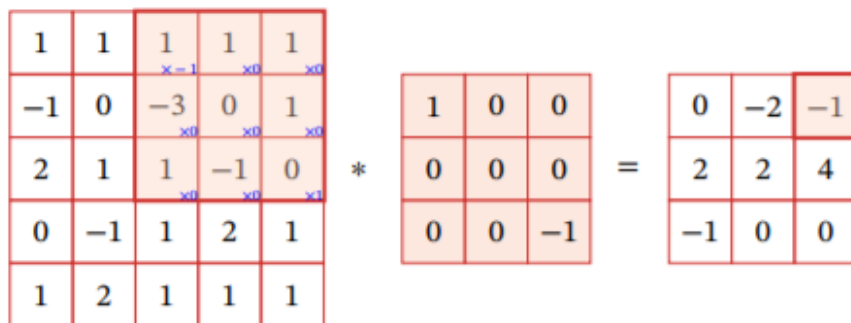


图 2.2 二维卷积

卷积神经网络一般由卷积层、汇聚层和全连接层构成。

2.2.2 卷积层

卷积层是卷积神经网络的核心模块，被用来提取一个局部区域的特征，也是从人工设计的特征工程到使用深度学习自动进行特征提取的重要改进之一，不同的卷积核相当于不同具有感受野的特征提取器。由于卷积网络主要应用在图像处理上，而图像输入为二维张量，因此为了更充分地利用图像的局部信息，通常将卷积核为三维结构的张量，其大小为高度 $M \times$ 宽度 $N \times$ 深度 D ，由 D 个 $M \times N$ 大小的特征映射构成^[19]。随着近年来注意力机制的研究，卷积层也被认为是

一种具有较窄范围的注意力机制^[23]。不失一般性，假设一个卷积层的结构如下：

- (1) 输入特征映射组： $x \in \mathbb{R}^{M \times N \times D}$ 为三维张量（Tensor），其中每个切片矩阵 $X^d \in \mathbb{R}^{M \times N}$ 为一个输入特征映射， $1 \leq d \leq D$ ；
- (2) 输出特征映射组： $y \in \mathbb{R}^{M' \times N' \times P}$ 为三维张量，其中每个切片矩阵 $Y^p \in \mathbb{R}^{M' \times N'}$ 为一个输出特征映射， $1 \leq p \leq P$ ；
- (3) 卷积核： $W \in \mathbb{R}^{U \times V \times P \times D}$ 为四维张量，其中每个切片矩阵 $W^{p,d} \in \mathbb{R}^{U \times V}$ 为一个二维卷积核， $1 \leq p \leq P, 1 \leq d \leq D$ 。

图 2.3 给出卷积层的三维结构表示。

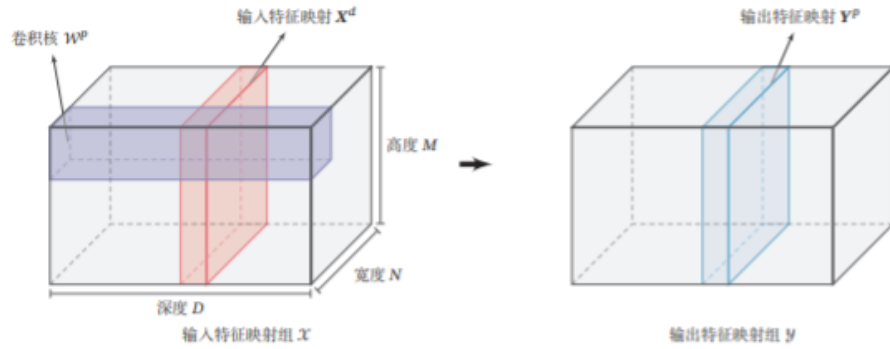


图 2.3 卷积层的三维结构表示

按照如下公式计算输出特征：

$$Z^p = W^p \otimes X + b^p = \sum_{d=1}^D W^{p,d} \otimes X^d + b^p \quad (2.5)$$

$$Y^p = f(Z^p) \quad (2.6)$$

其中 $W^p \in \mathbb{R}^{U \times V \times D}$ 为三维卷积核， $f(\cdot)$ 为非线性激活函数，一般用 ReLU 函数。

2.2.3 池化

池化（Pooling Layer）也叫子采样层（Subsampling Layer），其作用是进行特征选择，降低特征数量，从而减少模型数量，加快收敛提升效率。实际上，在卷积神经网络中，卷积层所输出的特征映射中的神经元数量和维度并没有显著减少，若在其后接分类器，输入分类器的维数依然很高，很容易出现过拟合（Overfit）^[24]，也会给训练带来问题。为了解决这个问题，可以在卷积层之后加上一个池化层，从而降低特征维数，避免过拟合的同时提升效率。

假设池化层的输入特征映射为 $x \in \mathbb{R}^{M \times N \times D}$ ，对于其中任一特征映射 $X^d \in \mathbb{R}^{M \times N}$ ， $1 \leq d \leq D$ ，划分为任意的可重叠区域 $R_{m,n}^d$ ， $1 \leq m \leq M'$ ， $1 \leq n \leq N'$ （通常在网络设计时决定）。池化（Pooling）是指对每个区域进行下采样（Down Sampling）得到一个值，作为这个区域的输出，舍弃剩余值。本文使用最大汇聚（Maximum Pooling 或 Max Pooling）：对于一个区域 $R_{m,n}^d$ ，选择这个区域内所有神经元的最大活性值作为这个区域的表示，即

$$y_{m,n}^d = \max_{i \in R_{m,n}^d} x_i \quad (2.4)$$

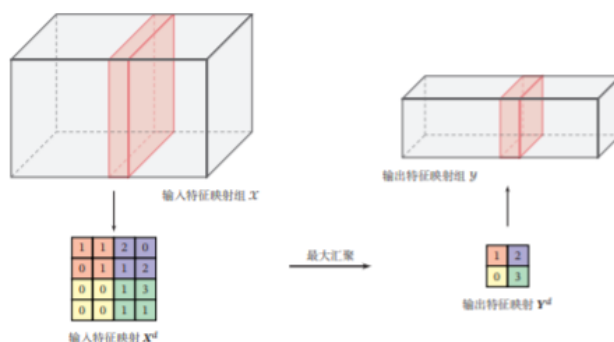


图 2.4 汇聚层中最大汇聚过程示例

2.2.4 全连接层

全连接层（Full Connected Layer）与前一层中的所有神经元全连接，将输入乘以权重矩阵，然后和偏置向量相加。全连接层常用于卷积神经网络的最后阶段，连接到目标输出层并构建出预测目标所需维数的输出向量。

2.2.5 Dropout 层

Dropout 层添加在每个卷积层和全连接层之后，通过抑制对训练数据的复杂协同适应来减少过拟合^[25]。实际上，Dropout 层在某种意义上可视为集成学习的一种，通过在训练期间以一定的丢弃率随机断开神经元之间的连接并从连接层丢弃神经元，来“逼迫”网络提高其鲁棒性。除此之外，Dropout 层还可以调高模型的训练速度。

一个典型的卷积网络是由卷积层、汇聚层、全连接层交叉堆叠而成。目前常用的卷积网络整体结构如图所示。一个卷积块通常为连续 M 个卷积层和 b 个池化层（ M 通常设置为 $2 \sim 5$ ， b 为 0 或 1）。一个卷积网络中可以堆叠 N 个连续的卷积块，然后在其后接 K 个全连接层（ N 的取值区间比较大，比如 $1 \sim 100$ 或者更大； K 一般为 $0 \sim 2$ ），最终经过 Softmax 得到输出。

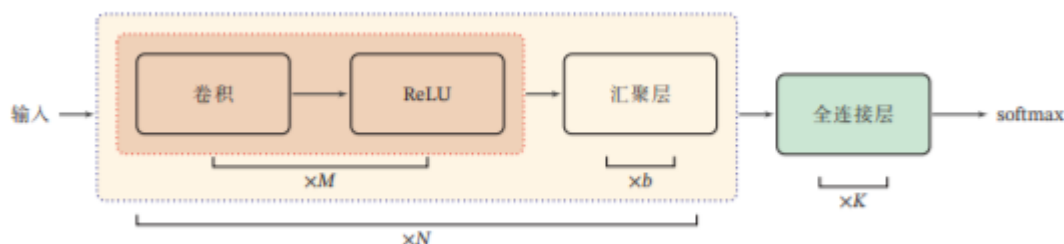


图 2.5 汇聚层中最大汇聚过程示例

2.3 循环神经网络

在前馈神经网络中，输入从前向后层层单向传递，这种限制虽然使得网络变得更容易训练，但在一定程度上减弱了神经网络模型的能力^[19]。前馈神经网络可以看作一个复杂的函数，每次输入都是独立的，即网络的输出只依赖于当前的输入。但是在很多现实任务中，网络的输出不仅和当前时刻的输入相关，也和过去一段时间的输入相关。比如一个有限状态自动机，其下一个时刻的状态（输出）不仅仅和当前输入相关，也和当前状态（上一个时刻的输出）相关。此

外，前馈神经网络难以处理时序数据，比如视频，语音，文本等^[19]。

循环神经网络（Recurrent Neural Network）是一类具有短期记忆能力的神经网络。在循环神经网络中，神经网络不但可以接受其他神经元的信息，也可以接受自身保留的信息（即能读能写），在时间上形成具有环路的网络结构^[26]。循环神经网络的参数学习可以通过随时间反向传播算法来学习。

由于循环神经网络具有短期记忆能力，相当于存储装置，因此其计算能力理论上可以近似任意的非线性动力系统^[27]。显然地，循环神经网络是图灵完备的。

2.3.1 门控循环单元神经网络

本文使用的门控循环单元神经网络（Gate Recurrent Unit）是循环神经网络（Recurrent Neural Network, RNN）的一种^[28]。和 LSTM（Long-Short Term Memory）一样^[3]，是为了解决长期记忆和反向传播中的梯度消失和爆炸等问题而提出来的。相比 LSTM，GRU 能达到相当的效果，且更容易训练^[29]。

GRU 结构如下：

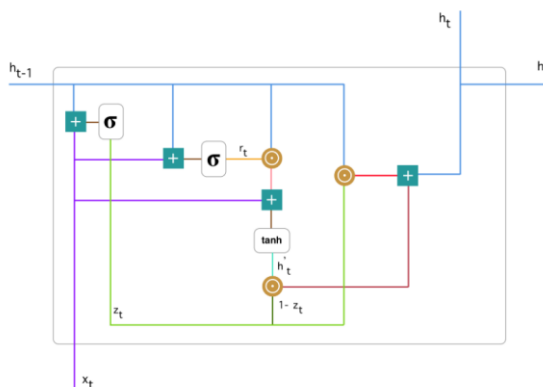


图 2.6 GRU 结构

对于输入 x^t ，有输出 y^t ，其计算公式如下：

$$\begin{aligned} r_t &= \sigma(W_{ir}x_t + b_{ir} + W_{hr}h_{(t-1)} + b_{hr}) \\ z_t &= \sigma(W_{iz}x_t + b_{iz} + W_{hz}h_{(t-1)} + b_{hz}) \\ h'_t &= \tanh(W_{in}x_t + b_{in} + r_t * (W_{hn}h_{(t-1)} + b_{hn})) \\ h_t &= (1 - z_t) * h'_t + z_t * h_{(t-1)} \end{aligned} \quad (2.5)$$

2.3.1.1 更新门

在时间步 t ，我们首先需要使用以下公式计算更新门：

$$z_t = \sigma(W_{iz}x_t + b_{iz} + W_{hz}h_{(t-1)} + b_{hz}) \quad (2.6)$$

其中 x_t 为第 t 个时间步的输入向量， $h_{(t-1)}$ 保存的是前一个时间步 $t-1$ 的信息，他们各自经过线性变换，即分别与对应权重矩阵 W 相乘，两部分信息加和后投入到 Sigmoid 激活函数中，

输出被映射到 0 到 1 之间的结果。下图是更新门的计算在整个单元中的位置与示意。更新门设计思想是帮助模型决定到底要将多少过去的信息传递到未来，这一点非常重要，因为模型需要决定从过去复制那些信息以减少梯度消失的风险。

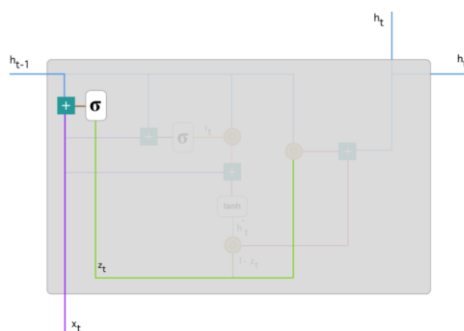


图 2.7 更新门

2.3.1.2 重置门

本质上来说，重置门主要负责决定了有多少上一时刻传来的信息需要遗忘，根据如下表达式计算重置门：

$$r_t = \sigma(W_{ir}x_t + b_{ir} + W_{hr}h_{(t-1)} + b_{hr})$$

该表达式与更新门的表达式是一样的，只不过线性层分别拥有各自独立的参数。下图展示了该运算过程：

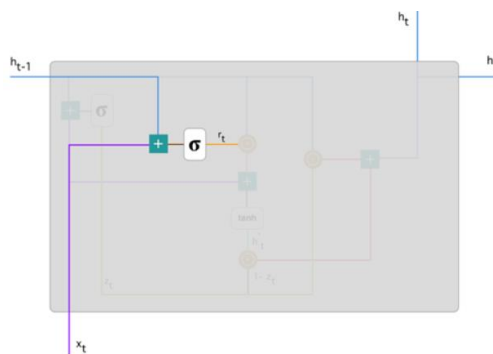


图 2.8 重置门

如前更新门所述， $h_{(t-1)}$ 和 x_t 先经过一个线性变换，再相加投入 Sigmoid 激活函数以输出激活值。

2.3.1.3 当前记忆内容

使用更新门与重置门的输出，最终得到当前的记忆，其的计算表达式为：

$$h'_t = \tanh(W_{in}x_t + b_{in} + r_t * (W_{hn}h_{(t-1)} + b_{hn}))$$

输入 x_t 与上一时间步的信息 $h_{(t-1)}$ 先经过一个线性变换，重置门的输出向量全部元素的值为 0 到 1 之间，它会衡量门控开启的大小。例如某个元素对应的门控值为 0，那么它就代表这个

元素的信息完全被遗忘掉。该乘积将确定所要保留与遗忘的以前信息。将这两部分的计算结果相加再投入双曲正切激活函数中。该计算过程如图所示：

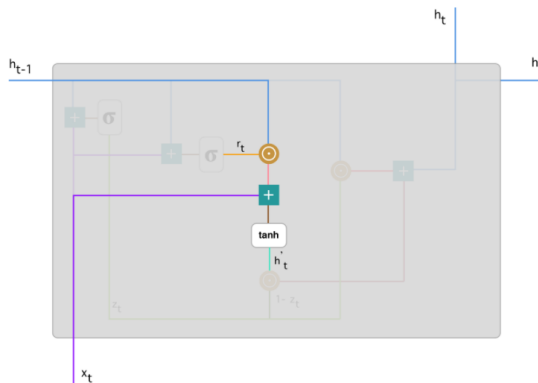


图 2.9 计算当前记忆

2.3.1.4 最终记忆

在最后一步，网络需要计算 h_t ，该向量将存储当前单元的信息和之前所有记忆的总结，并传递到下一个单元中。在这个过程中，使用更新门，决定当前记忆内容 h'_t 和前一时间步 $h_{(t-1)}$ 所需要总结的信息值。这一过程可以表示为：

$$h_t = (1 - z_t) * h'_t + z_t * h_{(t-1)}$$

z_t 为更新门的激活结果，其最终以门控的形式控制了信息的流入。 z_t 与 $h_{(t-1)}$ 的乘积表示前一时间步中保留到最终记忆的信息，该信息加上当前记忆保留至最终记忆的信息就等于最终门控循环单元输出的内容，注意最终的输出也就是循环神经网络的隐藏状态。这一过程如图所示：

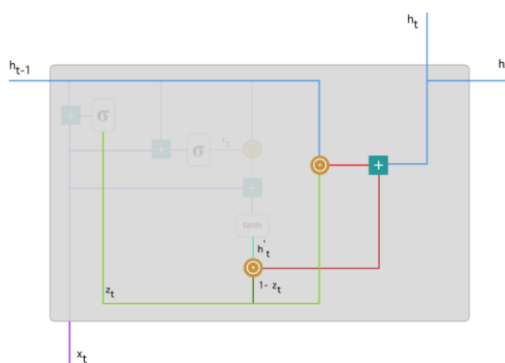


图 2.10 计算最终输出

2.4 表示学习

机器学习（Machine Learning, ML）是指从有限的观测数据中学习（也即“猜测”）出具有一般性的规律，并利用这些规律对未知数据进行预测的方法。传统的机器学习主要关注如何学习一个预测模型。一般需要首先将数据表示为一组特征，特征的表示形式可以是连续的数值、离散的符号或其他形式。然后将这些特征输入到预测模型，并输出预测结果。这类机器学习可

以看作浅层学习。浅层学习的一个重要特点是不涉及特征学习，其特征主要靠人工经验或特征转换方法来抽取。为了提高机器学习系统的准确率，我们就需要将输入信息转换为有效的特征，或者更一般性地称为表示。从原始输入数据中自动地学习出有效的特征，并提高最终机器学习模型的性能的方法叫表示学习^[19]。传统的特征学习一般是通过人为地设计一些准则，然后根据这些准则来选取有效的特征。特征的学习是和最终预测模型的学习分开进行的，因此学习到的特征不一定可以提升最终模型的性能。本文使用自编码器，来进行时间序列的建模，是它先将输入压缩成潜在空间表征，然后通过这种表征来重构输出。自编码器由两部分组成，编码器部分用于将输入压缩成另一空间的表征，解码器部分用来重构回到样本空间^[30]。

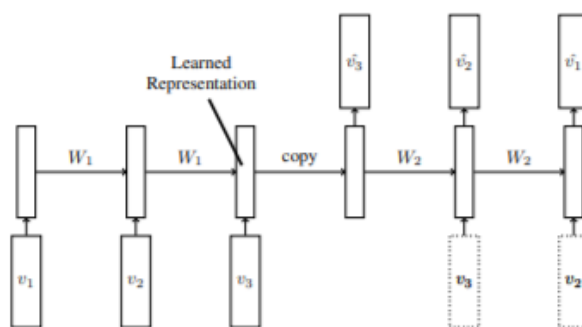


图 2.11 自编码器结构

自编码器具有表示学习算法的功能，循环神经网络自动编码器是使用编码器-解码器 RNN 体系结构的序列数据自动编码器的实现。对于给定的序列数据集，编码器 RNN 配置为读取输入序列，对其进行编码，解码器对其进行解码并重新创建。基于模型重新创建输入序列的能力来评估模型的性能。拟合后，模型的编码器部分可用于编码或压缩序列数据，而序列数据又可用于数据可视化或用作监督学习模型的特征向量输入自编码器结构。因此，整个自编码器可以用函数 $g(f(x)) = r$ 来描述，其中输出 r 与原始输入 x 相近。

2.5 迁移学习

标准机器学习的前提假设是训练数据和测试数据的分布是相同的，即所要拟合的函数在训练数据和测试数据上均匀的体现。如果不满足这个假设，在训练集上学习到的模型在测试集上的表现往往会比较差。而在本文的研究背景下，拟合结构地震时程必须要有目标建筑结构大量已知数据，才可通过机器学习方法进行预测，先前的研究已经证实了这一点。通过有限元软件模拟或计算，或是真实实验所得到的数据，即是对在目标建筑结构的刚度矩阵和质量矩阵下，确定参数的唯一确定的动力学微分方程的输入与输出映射，而一旦更换目标建筑，这一过程需从头再来，如何减少这一过程的成本，即解输入为有限元参数和地震动的泛函微分方程，这带来了巨大的挑战。

因此，如果存在一个有能力通过算力来抽取不同结构的响应特点，便可在已有的大量的训练数据基础上不断叠加，利用这些训练数据的不同结构和其在不同地震动下的响应，便能实现通用结构预测莫测，泛化性甚至可以预测未见过的建筑结构，现阶段这一构想仍是非常困难

和具有挑战性的问题，具体来说，我们至少可以使用迁移学习来减少已训练模型部署到一新结构上所需的训练数据量和时间，通过将相关任务的训练数据中的可泛化知识迁移到目标任务上，使用迁移学习（Transfer Learning）解决问题^[30]。

装
订
线

3 实验部分

3.1 实验准备

3.1.1 结构概况

建筑结构在不同地震作用下的响应具有一定的不确定性，目前主要的方法是时程分析，在本文中，预测目标包结构所有结点在整个地震中结构加速度响应最大值以及整个地震每个时间步的结点加速度。

本文采用的结构包括一个三层框架结构模型，一个五层框架模型。均使用有限元软件 ABAQUS 进行时程分析

在有限元软件 ABAQUS 中采用梁柱单元建立钢筋混凝土悬臂梁模型，三层框架的结构长宽高分别为 16.2*10.8*9m，五层框架的长短高为：27*16.2*15m。混凝土密度取 7800kg/m³，弹性模量取 2.0×10^4 MPa，泊松比取 0.3；钢材料密度取 2500 kg/m³，弹性模量取 3×10^4 MPa，泊松比取 0.2，

三层框架结构简图与构件截面如图所示：

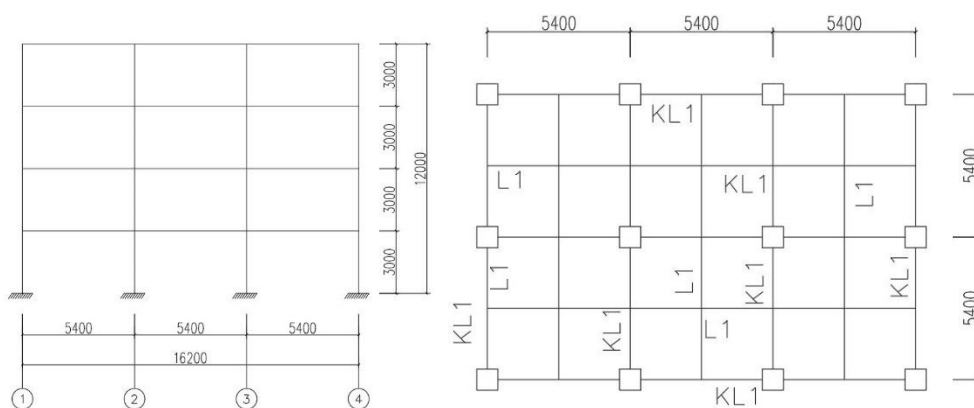


图 3.1 三层框架结构简图

五层框架结构简图与构件截面如图所示：

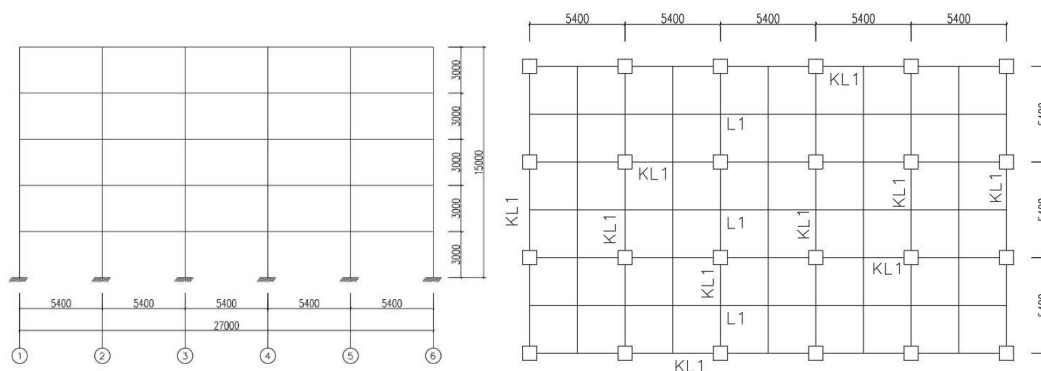


图 3.2 五层框架结构简图

3.1.2 地震动选取与计算结果

3.1.2.1 地震动选取

在建筑结构的抗震设防信息方面，结构场地抗震设防烈度为 8 度，场地类别为II类场地，设计地震分组为第一组，设计基本地震加速度为 0.2g，地震动影响系数最大值为: 0.16，结构特征周期为 0.45s，结构阻尼比采取全楼统一的方式，结构抗震设防类别为重点设防类别，抗震设防等级为二级，设计基准期 50 年。根据以上参数选取地震动。

本文使用的地面运动数据来自 PEER 地面运动数据库，共计 14 条地震波，28 条地震动数据，包括：

表 3.1 地震动数据

地面运动名称	采样步数	采样间隔	持续时间
		秒/s	秒/s
Borrego	10000	0.005	50
CPC Topanga Canyon	699	0.01	6.99
Helena, Montana-01	5103	0.01	51.03
Helena, Montana-02	2110	0.01	21.1
Humbolt Bay	8000	0.005	40.0
Imperial Valley-01	6001	0.005	30.005
Imperial Valley-02	5372	0.01	53.72
Imperial Valley-03	8000	0.005	53.46
NGA_no_187_H	7876	0.005	39.38
NGA_no_2958_CHY054	17999	0.005	89.995
NGA_no_949_ARL	2000	0.02	40.0
Northern Calif-01	8000	0.005	40.0
Northwest Calif-01	8000	0.005	40.0
Northwest Calif-02	8000	0.005	40.0

由于地震动原始数据的采样间隔不同，为了保证有限单元法的计算精度，采用原始地震步长作为输入，为了方便进行大规模有限元分析，使用 Python 编程语言调用 ABAQUS 接口，实现了批量计算，计算机 CPU 为 Intel-i7-10700，内存为 32G，显卡为 NVIDIA RTX3080。

3.1.2.2 计算结果

结构施加的竖向荷载为重力荷载代表值，横向荷载考虑为 x 轴单向输入。选取 Borrego 地震波作为示例展示计算结果，图 3.3 为地震动可视化结果，图 3.3.1 为所有节点的加速度时程曲线。

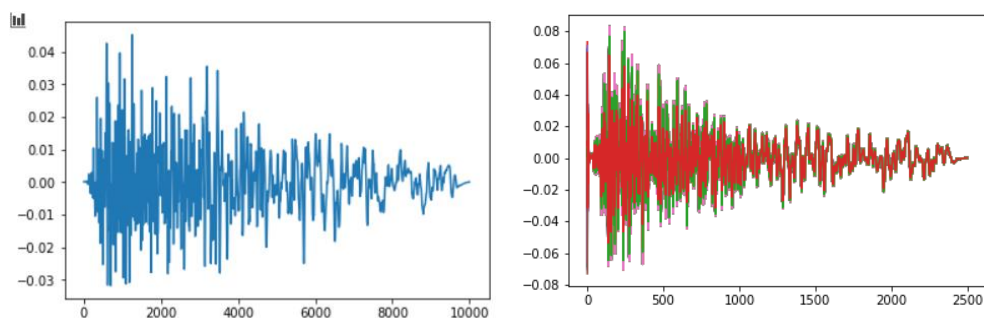


图 3.3 Borrego

注意到计算结果在 0 时刻附近的响应本应为全 0，但是由于存在一起软件参数设置的问题，误差较大，故在数据预处理阶段做截断处理。同时，为了减少训练难度，增大采样间隔。

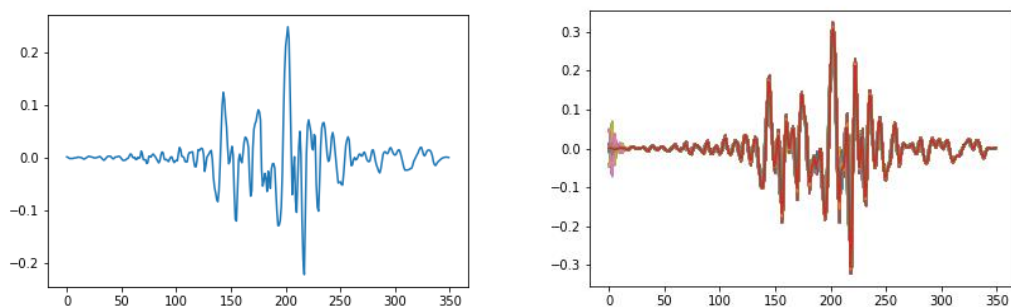


图 3.4 地震动和响应时程

通过检查，可判断计算结果基本合理，作为模型训练使用的原始数据。

3.1.3 数据集生成

用于模型训练的数据集氛围输入的地震动和输出的结构响应，每个样本包含要表示为一维向量的一条地震动，和对应一个结构响应的二维矩阵。由于后续模型需要进行结构地震响应的时程预测，所以这里采用统一时间步的做法。

3.1.3.1 数据格式化

在前文计算方法中已提到，由于各个地震动不同的长度和持续时间，数据不能保持固定长度的矩阵，同时为了保证有限元的计算精度，并没有进行插值而是使用原始数据进行计算。

因此，为了导入到模型进行训练，对地震动数据进行采样来形成相对合理的长度，实际上从算法的合理性考虑，维持相同的时间步长是相当有必要的，因此本文选取了超参数采样周期 0.5 来作为算法的期望输入步长，即根据原始地震数据的采样周期与超参数的比值，来从结构响应中抽取，本文并不对时间序列作截断处理，因为自回归的时间序列预测模型并不依赖于单个序列的长度^[19]。

3.1.3.2 数据预处理

为了加快模型收敛，降低拟合难度，标准化是机器学习中一种常用的预处理方法^[3]，本文采用 0-1 标准化方法，定义如下：

$$x^* = \frac{x-u}{\sigma} \quad (3.1)$$

式中：

x^* 标准化后数据

u 数据均值

σ 数据方差

使用 Python 的 Scikit-learn 机器学习库完成此过程。最终数据划分比例为训练集合：验证集：测试集 = 8：1：1：

3.2 算法设计

使用一个 GRU 对结构节点加速度的时间序列进行建模。

算法设计围绕：

$$[M]\{\ddot{u}\} + [C]\{\dot{u}\} + [K]\{u\} = \{F(t)\} \quad (3.2)$$

式中：

$[M]$ 结构的质量矩阵；

$[C]$ 结构的阻尼矩阵；

$[K]$ 结构的刚度矩阵；

$\{\ddot{u}\}$ 节点位移列阵；

$\{\dot{u}\}$ 节点速度列阵；

$\{u\}$ 节点加速度列阵；

$\{F(t)\}$ 节点荷载列阵。

先前使用神经网络模型直接对输入地震动-输出结构响应的研究本质为求解该离散偏微分方程（Partial Different Equation）^[3]在固定参数下，例如输入为不同地震动，输出为固定结构的地震响应（通过有限元方法计算或通过模拟或是真实实验），且地震响应大多较为固定，如某层的最大位移，某层的层间位移角，（且多为顶层）输入序列数据最终进行回归分析即时间序列估计问题。另外，这类研究的证明在于神经网络能够求解固定参数的动力学方程^[32]，当需要对另一结构进行分析时，则需要重新进行训练，这就要求了重新获得大量有限元计算结果作为训练数据集，从这个角度，算法的泛化性没有得到体现。另一方面，所研究的算法大都为损伤预测，即建立地震-损害的端到端映射，没有体现出时程分析的逐个时间步向前过程，缺少可信度^[18]。最后，针对以上几点问题，本文提出了一个双层堆叠的编码器解码器循环神经网络模型，输入为结构的有限元矩阵和地震运动，输出为结构每个有限元结点在整个地震过程中的加速度。

选用基于卷积神经网络的编码器解码器模块是为了从有限元参数矩阵中提取特征，实际上

该方法面对的主要挑战是非常高维的矩阵，参考自然语言处理中（Nature Language Process, NLP）的语义分析思想，本文使用奇异值分解(singular value decomposition, SVD)^[19]进行降维，理论上使用主成分分析法（Principal Components Analysis, PCA）或其他方法也是可行的，或是使用深度学习的结构特征抽取例如生成对抗网络（Generative Adversarial Network, GAN）^[32]。事实上该步骤的主要目的是通过输入有限元矩阵到深层神经网络中得到不同结构的横向泛化性。

在时间序列建模问题上，使用双层 GRU 是考虑高维输入，在弹性时程分析中，往往具有较高的结点数量，拟合如此高维的向量具有较大挑战性，本文的三层框架结构结点数量为 294 个，五层框架为 529 个。每个堆叠 GRU 的输入为上一层输出和对应编码器-解码器模块输出的融合，以期能够利用结构特征。

最终模型输出为结构每个时间步，所有有限元结点的加速度，计算与真实值的均方损失来进行反向传播并更新参数。预测时，只需简单更改模型，自回归地前向进行即可得到完整地震下的结构响应。该响应可精确到地震过程中的任意时间步，结构中任意结点的加速度。

同时考虑模型的迁移性，即移植到完全不同维度的新结构上能够快速训练，考虑在输入和输出层加入简单线性层，这样迁移到新结构上只需要微调即可。

FesNet 模型结构如图所示：

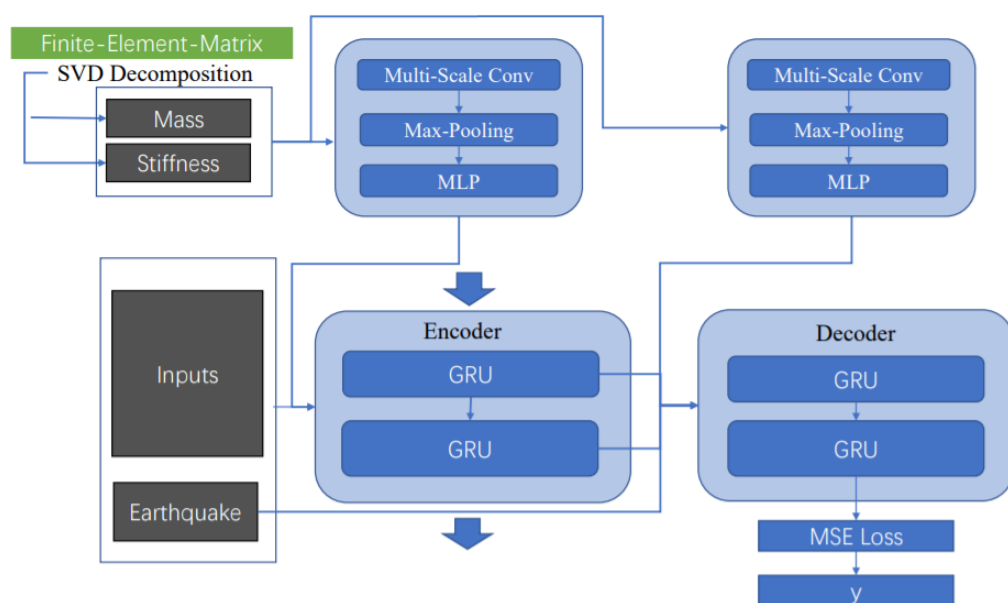


图 3.5 FesNet 模型结构

在本文的实验中，通过 ABAQUS 有限元软件导出的结构动力偏微分方程参数包括质量矩阵，刚度矩阵，维度为 1784*1784，分别通过奇异值分解降维后合并为 144*1784 的举证，作为编码器模块的输入，通过卷积，池化，全连接，Dropout 层后作为堆叠 GRU 模块的输入。

循环神经网络的输入为地震动加速度和结构当前时刻的加速度，与来自编码器的输出合并后输入到双层门循环神经网络中，其输入与编码器模块的输出合并后作为最后双层门循环神经网络模块的输入，得到预测值。

3.2.1 预处理

奇异值分解即本文所使用的矩阵因子分解方法,其特点是分解的矩阵正交。其定义如下:将一个非零的 $m \times n$ 实矩阵 A , $A \in \mathbf{R}^{m \times n}$ 表示为以下三个实矩阵乘积形式的运算文即进行矩阵的因子分解^[19]

$$A = U\Sigma V^T \quad (3.3)$$

其中 U 是 m 阶正交矩阵 (orthogonal matrix), V 是 n 阶正交矩阵, Σ 是由降序排列的非负的对角线元素组成的 $m \times n$ 矩形对角矩阵 (rectangular diagonal matrix), 满足:

$$\begin{aligned} UU^T &= I \\ VV^T &= I \\ \Sigma &= \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_p) \\ \sigma_1 &\geq \sigma_2 \geq \dots \geq \sigma_p \geq 0 \\ p &= \min(m, n) \end{aligned} \quad (3.4)$$

$U\Sigma V^T$ 称为矩阵 A 的奇异值分解, σ_i 称为矩阵 A 的奇异值, U 的列向量称为左奇异向量 V 的列向量称为右奇异向量。

在矩阵的奇异值分解中,只取最大的 k 个奇异值 ($k < r, r$ 为矩阵的秩) 的对应部分,就得到矩阵的截断奇异值分解。实际应用中提到矩阵的奇异值分解时,通常指截断奇异值分解。本文使用截断奇异值对输入的使用有限单元法得到的微分方程参数矩阵进行降维,压缩矩阵的维度并期望从中进行减少冗余信息,实际上,自然语言处理中词嵌入所用的奇异值分解分解已经由词向量方法 (WORD2VECTOR)^[3] 所取代,这也是本文的不足之处之一,为了得到泛化性和加速计算目的,更多更深的矩阵特征抽取是十分必要的。

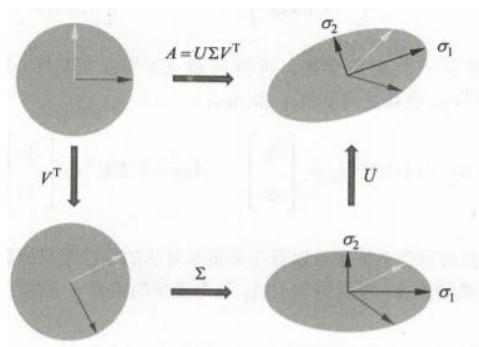


图 3.6 SVD 分解示意图

3.2.2 编码器-解码器

编码器-解码器模块使用相同架构的卷积神经网络,分别具有独立的参数,用于从降维后的参数矩阵中抽取特性,来获得对于不同结构的泛化能力。

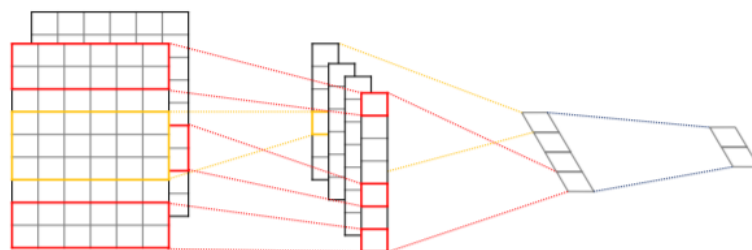


图 3.7 卷积神经网络结构

如图所示，编码器与解码器采用相同的卷积神经网络架构，在训练时具有各自独立的参数，卷积层包括 3 个不同尺寸的卷积核，卷积层的输出经过最大池化后输入全连接神经网络得到最终结果。卷积层包括尺寸分别为 3，4，5 的三个卷积核期望从不同感受野范围抽取特征。

3.2.3 堆叠 GRU

使用堆叠的两层门循环神经网络模型用于时间序列分析，单个 GRU 的输入为当前所有结点的加速度和地震加速度，与经过卷积神经网络模块的输出合并后输入第一个 GRU 模块。

堆叠 GRU 即将单个门循环神经网络的输出作为下一个循环神经网络的输入，一般情况堆叠循环神经网络层数不会高于 3 层，因为过高的 GRU 层数会导致时间和内存开销迅速以指数级增长，并且会出现严重的梯度消失问题，导致更新迭代放缓，收敛效果和效率急剧下降^[3]。考虑到地震响应预测的高度非线性，本文使用双层 GRU。

第一个堆叠 GRU 模块的输入由当前时刻响应和地震动与编码器输入融合而成，这是为了门循环神经网络能够在进行时间序列建模时利用到地震，先前的响应，与结构信息。编码器模块自回归地前向计算，最终得到当前序列的特征。解码时将下一时刻的地震动与另一个卷积神经网络模块的输入融合，同时使用编码器输出的隐藏层为作为初始隐藏层，对下一时刻响应进行预测。实际上由于要预测整个地震过程每个 k 时刻时所有结点的加速度值，实际问题中有限元所给出的结点数量往往非常巨大，这一难题给所应用的时间序列模型的非线性拟合能力提出了非常高的要求，基于前 k 步时程信息的假设远强于一般长短时记忆神经网络仅仅和前一步有关的假设，实际上由于地震具有非常强的非线性特征，这一假设并不能很好的完成预测任务或是会引起预测时程曲线的滞后效应^[3]。

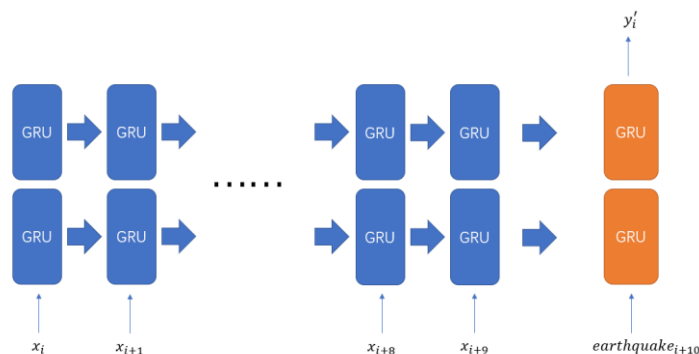


图 3.8 编码器-解码器结构

4 结果和讨论

4.1 实验结果

4.1.1 实验环境

整个模型训练过程是在 Python 环境中使用 Pytorch 完成的^[34]。Pytorch 由 Facebook 人工智能研究院（FAIR）团队开发，是一个构建在 Torch 上的开源深度学习算法库，它提供简单便捷的神经网络 API 供调用，使得算法设计无需关系底层优化即可快速部署测试模型。Torch 是一个基于 BSD License 的开源的机器学习的框架，一直聚焦于大规模的机器学习应用，尤其是图像或者视频应用等领域，本文所提出的模型在 NVIDIA GTX1080Ti 上进行训练。整个训练过程大约需要两小时，单步推理所需时间少于 0.1 秒，完整地震时程预测过程在数秒内左右。

本文使用的数据和代码已经在 GitHub 上公开，网址为 <https://github.com/piglaker/fesnet>

4.1.2 训练过程

对于式 3.2 的求解有解析方法和数值方法两种。对于线弹性结构，动荷载为解析函数时，可使用时域分析方法或频域分析方法得到其解析解^[4]。由于地震输入在时间上离散，求解析解往往非常复杂和困难，因此抗震结构时程分析一般使用数值方法。

进一步地，地震荷载下，上式可化为：

$$[M]\{\ddot{x}(t)\} + [C]\{\dot{x}(t)\} + [K]\{x_t\} = -[M]\{l\}\ddot{a}_g(t) \quad (4.1)$$

其中：

$\{\ddot{a}_g(t)\}$ 为地震动加速度；

$\{l\}$ 为单位列向量。

假定 $i \sim i+1$ 时刻加速度变化规律，，进一步有：

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx \end{cases} \quad (4.3)$$

$$A = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix}$$

$$B = [0, 1]^T, C = [1, 0]$$

其中 $x = [y, \dot{y}]$; u 为地震动加速度, $u = -\ddot{a}_g(t)$; $0, I$ 分别为全零矩阵，单位矩阵。

一般性地，假设位移，速度和加速度在每一时间间隔内连续变化，即为介于上一时刻与当前时刻之间的某一值使用前几个时刻的地震响应和加速度计算下一个时刻的结构地震响应，因此我们可以尝试用机器学习方法来拟合这一过程。

模型使用 Pytorch 进行搭建。所有网络权重参数使用正态分布随机初始化。优化器使用 Adam（Adaptive Momentum Estimation，自适应动量估计）^[34]，其公式如下：

$$\begin{aligned} m_t &= \mu * m_{t-1} + (1 - \mu) * g_t \\ n_t &= \nu * n_{t-1} + (1 - \nu) * g_t^2 \\ \hat{m}_t &= \frac{m_t}{1 - \mu^t} \\ \hat{n}_t &= \frac{n_t}{1 - \nu^t} \\ \Delta\theta_t &= -\frac{\hat{m}_t}{\sqrt{\hat{n}_t + \epsilon}} * \eta \end{aligned} \quad (4.4)$$

式中： η 为学习率，大小为 0.0002，超参数 μ, ν 分别为 0.9, 0.999。

使用多项式学习率衰减，间隔为 20 epoch，衰减系数为 0.1，其公式如下：

$$\text{decayed_learning_rate} = \text{learning_rate} * \text{gamma}^{\text{power}} \quad (4.5)$$

数据集与测试集的划分使用 Scikit-learn 的 `train_test_split` 函数进行随机打乱。损失函数使用均方误差，其公式如下：

$$\text{Loss} = \text{MSE}(y_i, \hat{y}_i) = \frac{1}{N} \sum_{t=1}^N (y_{i,t} - \bar{y}_{i,t})^2 \quad (4.5)$$

其中 N 为结点， $y_{i,t}$ 为 i 时刻第 t 维度对应的加速度真实值， $\bar{y}_{i,t}$ 为 i 时刻对应的位移预测值，均方误差衡量了预测值与真实值的绝对误差，平方损失越小，说明预测模型描述实验数据的精确度越高。

在数据集上训练共计 20 个时代耗时 2 小时左右。损失曲线如下图所示：

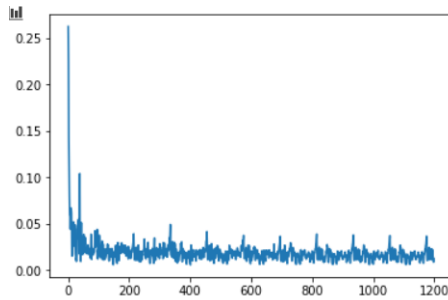


图 4.1 单步损失曲线

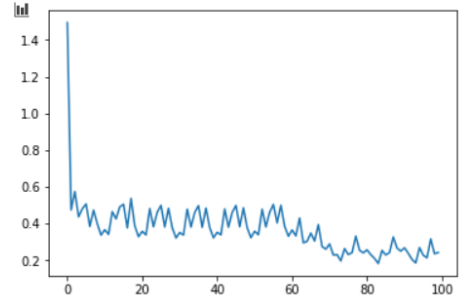


图 4.2 每个 EPOCH 上的损失曲线

通过观察可见模型收敛良好，可进行下一步测试。

4.2 测试与验证

为了研究模型在地震下的预测性能，本部分由两部分构成，第一部分在测试集上使用均方误差和 R^2 指标来对模型进行评价。最后会直接给出模型预测的可视化结果。

均方误差是直接衡量预测值与真实情况的指标，也是训练时使用的损失函数，这一结果可直观刻画模型预测的精确程度。 R^2 也称决定系数，用于衡量两个变量的相互关系和依赖程度，其计算公式为：

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (4.2)$$

其中 y_i 为第 i 个样本的真实值， \hat{y}_i 为第 i 个样本的预测值， \bar{y} 为所有样本的平均值。

4.2.1 时程分析

对测试集的 8 条地震波，自回归地进行前向预测，每条波预测时长均为 8s 左右，每次回归输入前 10 步的时程信息，一般为了保证时间序列预测的稳定性，会在训练时使用教师强迫（Teacher Force）^[19]，即每次预测都用真实值，而非自己的输出，下图给出了教师强迫和自回归两种方式分别的预测结果的差异。

最终，在测试集上，教师强迫预测的均方误差为 0.0067，R2 指标为 0.6850，平均绝对误差为 0.0571，平均相对误差为 13.32%，自回归预测的平均均方误差为 0.0066，R2 指标为 0.6859，平均绝对误差为 0.057，平均相对误差为 13.25%。

预测示例可视化如图所示：

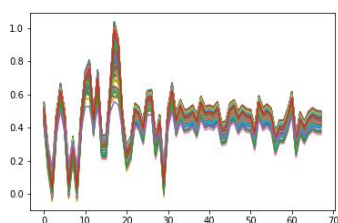


图 4.3 教师强迫预测结果

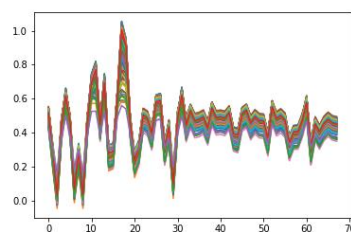


图 4.4 自回归预测结果

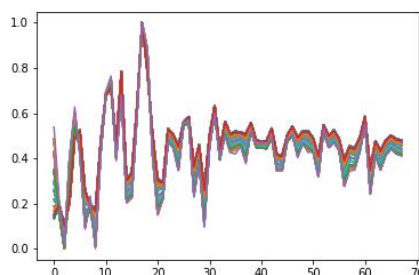


图 4.5 真实曲线

在测试集上，将采样步长为 0.5 秒改为 0.02 秒，可得更精细预测结果，耗时显著增加，单条地震波时程分析耗时约 110 秒左右（平均为 115.21 秒），教师强迫预测的平均均方误差为 0.0033，R2 指标为 0.6114，平均绝对误差为 0.0481，平均相对误差 12.05%，自回归预测的平均均方误差为 0.0034，R2 指标为 0.6110，平均绝对误差 0.0481，平均相对误差 12.08%。

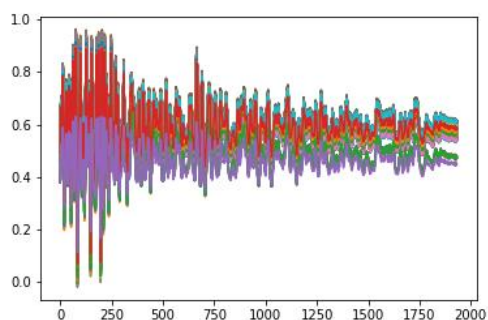


图 4.5 教师强迫预测结果

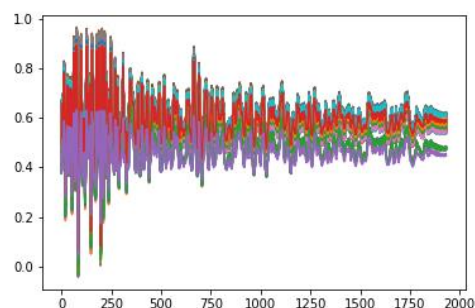


图 4.6 自回归预测结果

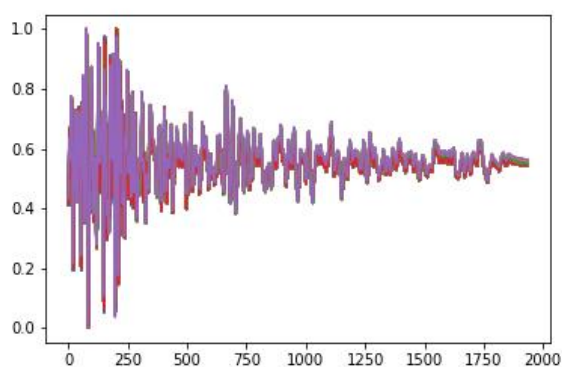


图 4.7 真实地震响应

4.2.2 五层框架上的迁移学习

为了验证模型的迁移性，在五层框架结构上进行迁移学习，五层框架的节点数为 529，使用双层全连接神经网络构建嵌入层，所用地震动数据与前相同，结果如下：

训练过程中的损失曲线：

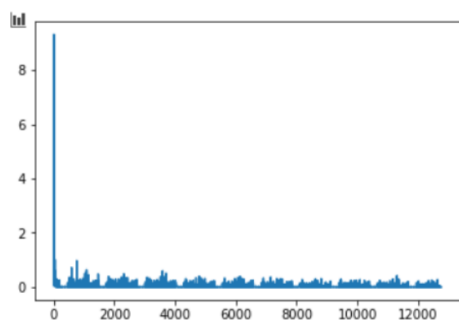


图 4.8 单步 LOSS 曲线

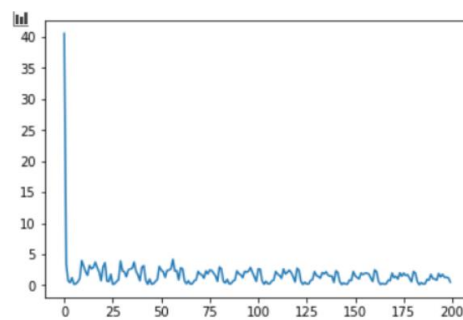


图 4.9 单个 EPOCH 上的 LOSS 曲线

从损失曲线可以看出收敛速度较快，且曲线较为平稳。在测试集上性能：预测单条波速度在 10 秒左右，整个测试集耗时 1 分钟左右，教师强迫预测的平均均方误差为 0.0086，R2 指标为 0.5814，平均绝对误差为 0.082，平均相对误差 22.10%，自回归预测的平均均方误差为 0.0091，R2 指标为 0.5512，平均绝对误差 0.081，平均相对误差 21.79%。

预测示例可视化如图所示：

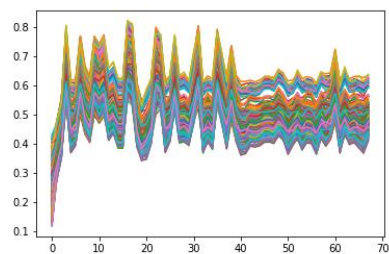


图 4.10 教师强迫预测结果

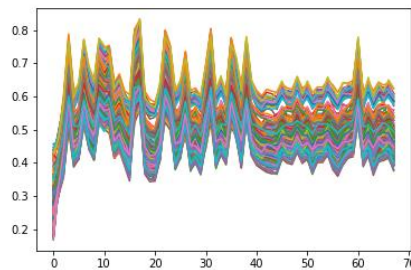


图 4.11 自回归预测结果

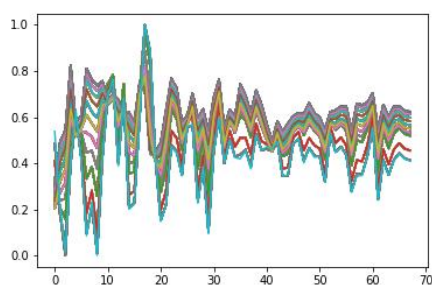


图 4.11 真实地震响应

可以看出除个别区间外，总体效果令人较为满意。

5 结论和展望

5.1 结论

本文对于国内外结构地震响应预测模型研究现状进行了调研，使用 ABAQUS 有限元软件进行了建筑结构的建模，使用 PEER 数据库中的地震动数据，处理之后输入 ABAQUS 进行计算，最终得到响应预测模型所需的训练数据集，之后，提出一种使用包含卷积神经网络模块的自编码神经网络（FesNet），用于地震下结构时程分析。FesNet 包含基于卷积神经网络的特征抽取模块以及用于时间序列建模的编码器-解码器模块，每个编码器，解码器由多个门控循环神经网络堆叠而成，来求解适用于求解含矩阵参数和离散输入的动力学偏微分方程。最后，就模型的预测效果进行了测试与实验，并就其泛化性进行了检验，作为对比，在另一五层框架结构上进行迁移学习以验证其迁移性。最终得出以下结论：

- （1）模型对于抗震响应的预测性能提升，通过与先前方法性能的对比，可以发现新提出的模型效果精度更好，与有限元方法对比，本文所提出的模型对结构时程分析预测效率更高。
- （2）在预测精度上，先前研究的预测目标多为层间位移或者加速度，本文提出的算法实现了结构全体有限元结点的响应预测，补充了结构损失评估的时程分析过程，可信度更高。
- （3）在五层建筑结构上进行的迁移学习实验，可以快速收敛，验证了所提出算法的迁移性。先前的研究为求解确定参数的动力学偏微分方程，不具备迁移性，应用到新结构上时耗费巨大，本文所提出的模型降低了应用成本。
- （4）本文所提出的模型在不同结构上具有一定的可重用性，且计算非常迅速，能够在数秒内完成地震下整个结构所有有限元结点的时程分析，实验结果表明，调整预测步长，模型仍具有很强的泛化性，这一算法为有限元精度的城市级抗震时程分析带来了可能性。

最后，本文虽然提出了改进的结构地震响应预测算法，解决了有限元精度的快速结构响应时程分析的快速计算和可迁移的结构表示学习方法，但仍存在不足之处，比如：训练集数据规模仍不够大，在有限单元法参数矩阵的降维处理和特征提取模块设计有很大改良空间，同时，在处理现实大型建筑结构时，仍有很多细节需要完善和改进。

5.2 展望

- （1）本文所实验的模型结构仍较为单薄，参数数量较少，实际上为了提高模型的拟合能力，可以尝试更大规模数据集在更多参数的模型上进行实验，更深层网络结构也是十分有必要的。
- （2）该方法仅提出了一种可迁移的结构地震响应快速计算方法，其核心思想仍是得到结构的低维表示，但与在真实大型结构的实际应用任务仍存在一定距离，距离韧性城市抗震仍需大量研究。
- （3）由于本文所研究问题本质为泛函微分方程，且具有非常强的非线性，特别是随着输入维

度的提高，拟合方程所需的参数数量至少是多项式上升的，所以对于实际输入的参数矩阵的处理仍需要更多优化。

装

订

线

参考文献

- [1] 吕西林.抗震设计理论与实例[J].同济大学出版社, 2014(4):13-14.
- [2] Lu, Xinzhen; McKenna, Frank; Cheng, Qingle; Xu, Zhen; Zeng, Xiang; Mahin, Stephen A (2020). An open-source framework for regional earthquake loss estimation using the city-scale nonlinear time history analysis. *Earthquake Spectra*, (), 875529301989172-. doi:10.1177/8755293019891724
- [3] 汪鑫. 基于优化神经网络的多层钢筋混凝土结构地震响应预测[D].哈尔滨工业大学,2020.
- [4] 王新敏. ANSYS 结构动力分析与应用[J].人民交通出版社, 2014(4):80-83
- [5] Zi Wang & Aws Albarghouthi. Interval Universal Approximation for Neural Networks. 2021. arXiv:2007.06093
- [6] 何玉敖,胡贤忠,詹胜.应用人工神经网络(ANN)预测结构的地震响应[J].Transactions of Tianjin University,1996(02):41+38-40.
- [7] 叶列平,陆新征,马千里,汪训流,缪志伟.混凝土结构抗震非线性分析模型、方法及算例[J].工程力学,2006(S2):131-140.
- [8] Tekkaya A.E., Soyarslan C. (2018) Finite Element Method. In: Int. Academy for Production Engineering, Chatti S., Tolio T. (eds) CIRP Encyclopedia of Production Engineering. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-35950-7_16699-3
- [9] Lu, L., Jin, P., Pang, G. et al. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nat Mach Intell* 3, 218–229 (2021). <https://doi.org/10.1038/s42256-021-00302-5>
- [10] 高经纬,张春涛.基于长短时记忆网络的结构地震响应预测[J].工程抗震与加固改造,2020,42(03):130-136.
- [11] Morfidis, K., & Kostinakis, K. (2018). Approaches to the rapid seismic damage prediction of r/c buildings using artificial neural networks. *Engineering Structures*, 165, 120–141. doi:10.1016/j.engstruct.2018.03.028
- [12] Hong Peng, Jingwen Yan, Ying Yu, Yaozhi Luo, Time series estimation based on deep Learning for structural dynamic nonlinear prediction,Structures,Volume 29,2021,Pages 1016-1031,
- [13] Perez-Ramirez CA, Amezcua-Sanchez JP. Recurrent neural network model with Bayesian training and mutual information for response prediction of large buildings [J]. *Engineering Structures*, 2019,178: 603-15\$.
- [14] Azimi, M., & Pekcan, G. (2019). Structural health monitoring using extremely compressed data through deep learning. *Computer-Aided Civil and Infrastructure Engineering*. doi:10.1111/mice.12517
- [15] Ruiyang Zhang, Yang Liu, Hao Sun, Physics-guided convolutional neural network (PhyCNN) for data-driven seismic response modeling, *Engineering Structures*, Volume 215, 2020, 110704, ISSN 0141-0296,
- [16] Byung Kwan Oh, Branko Glisic, Sang Wook Park, Hyo Seon Park, Neural network-based seismic response prediction model for building structures using artificial earthquakes, *Journal of Sound and Vibration*, Volume 468, 2020, 115109, ISSN 0022-460X
- [17] Liang L, Liu M, Martin C, Sun W. 2018 A deep learning approach to estimate stress distribution: a fast and accurate surrogate of finite-element analysis. *J. R. Soc. Interface* 15: 20170844. <http://dx.doi.org/10.1098/rsif.2017.0844>
- [18] Yongjia Xu, Xinzhen Lu, Barbaros Cetiner, Ertugrul Taciroglu. Real-time regional seismic damage assessment framework based on long short-term memory neural network. *INDUSTRIAL APPLICATION*. 16 October 2020 <https://doi.org/10.1111/mice.12628>.
- [19] 邱锡鹏, 神经网络与深度学习, 机械工业出版社, <https://nndl.github.io/>, 2020.
- [20] Yamashita, R., Nishio, M., Do, R.K.G. et al. Convolutional neural networks: an overview and application in radiology. *Insights Imaging* 9, 611–629 (2018). <https://doi.org/10.1007/s13244-018-0639-9>
- [21] Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems* (pp. 1097–1105).
- [22] Kim, Y. (2014). Convolutional neural networks for sentence classification. *arXiv preprint*

arXiv:1408.5882.

- [23] I. Bello, B. Zoph, Q. Le, A. Vaswani and J. Shlens, Attention Augmented Convolutional Networks,"2019 IEEE/CVF International Conference on Computer Vision (ICCV), 2019, pp. 3285-3294, doi: 10.1109/ICCV.2019.00338.
- [24] Scherer, Dominik et al. Evaluation of Pooling Operations in Convolutional Architectures for Object Recognition. ICANN (2010).
- [25] Nitish Srivastava et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting,15(56):1929–1958, 2014.
- [26] Gang Chen. A Gentle Tutorial of Recurrent Neural Network with Error Backpropagation. 2018. arXiv:1610.02583.
- [27] Schäfer, Anton Maximilian and H. Zimmermann. Recurrent Neural Networks Are Universal Approximators. International journal of neural systems 17 4 (2006): 253-63 .
- [28] Cho, K., Van Merriënboer, B., Bahdanau, D., & Bengio, Y. (2014). On the properties of neural machine translation: encoder-decoder approaches. arXiv preprint arXiv:1409.1259.
- [29] Chung, J., Gulcehre, C., Cho, K., & Bengio, Y. (2014). Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv preprint arXiv:1412.3555.
- [30] Shen, C. & Guo, Y.. (2018). Unsupervised Heterogeneous Domain Adaptation with Sparse Feature Transformation. Proceedings of The 10th Asian Conference on Machine Learning, in PMLR 95:375-390
- [31] Kevin Clark et al. (2020).ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. arXiv:2003.10555
- [32] Young, Tom et al. Recent Trends in Deep Learning Based Natural Language Processing [J]. IEEE Computational Intelligence Magazine 13 (2018): 55-75.
- [33] Kingma, D. P., & Ba, J. (2014). Adam: a method for stochastic optimization. arXiv preprint arXiv:1412.6980.

装

订

线

谢辞

哈哈哈哈哈。

时光荏苒，同济大学的四年本科生活终于画上了句号，我本以为这时候的话，会狂喜，或是会悲伤，或者是淡定坦然，或者是洒脱不羁……或者说我本以为四年的大学生活会让我成熟点……唉好像确实也是，我成熟了，不过此时此刻写致谢的我依然是个吊儿郎当的人，不知道为啥，可能是因为我论文字数不够吧，就很想额外说废话。

首先按感谢我的父母，没有他们的精神和经济帮助，我是无法完成本科学业。虽然我爸妈没能给我什么特别有用的帮助，甚至还反对过我，不过大体上还是过得去的。

我也非常感谢在同济遇到的几位导师，感谢周颖老师对我学术素养的培养，王胤老师对我视野的开阔，丛培盛老师对我人生的指导，周勇老师对我生活的关心。

其次，我要感谢在同济遇到的一些同学，孟同学与我即是朋友又是搭档，我俩性格和风格完美互补，他大巧不工，我剑走偏锋。同样，我感谢赵奕昕同学和我之间的美好时光，和江勇同学以及刘维翰同学共度的室友情谊。

在土木工程的四年生活，磨砺了我，他们说苦难是人生的财富，现在的我会说，确实如此。不过我也非常幸运，我这把刀切过了四年时光，保持住了他的锋利。