基于神经网络结构的压缩加速研究

- 答辩人：赖碧兰 MG1933032
- 导　师：申富饶 教授

南京大学
NANJING UNIVERSITY

目录
CONTENTS

南京大学
NANJING UNIVERSITY

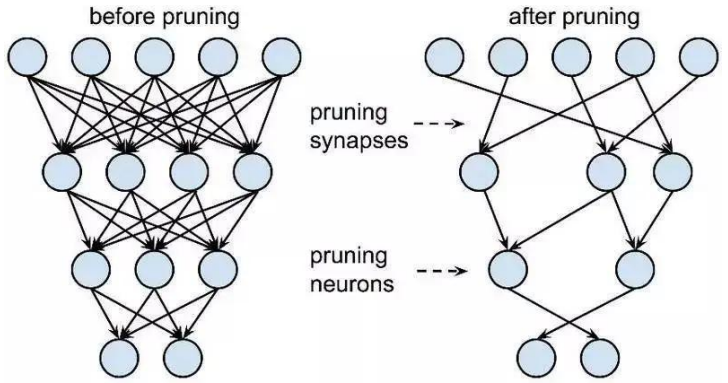
第一部分

Research Background

研究背景

背景简介

- 神经网络压缩加速：在不影响神经网络性能的情况下，通过某些方法降低网络计算代价和存储空间



- 应用场景——小型边缘设备应用智能化的需求日益增加，模型的轻量化需求日渐重要

移动设备

人脸识别

智能家居

自动驾驶

研究意义

计算量下降

测试时间减少
训练时间减少
模型系统运行加速

参数量下降

模型收敛加速
缓解梯度消失现象
缓解过拟合
降低模型存储量

相关工作

压缩方法	方法简介	应用结构
网络剪枝	删除对网络性能影响不大的参数及结构	卷积层和全连接层
紧凑卷积核设计	设计精巧的卷积核来搭建网络	卷积层
低秩分解	使用矩阵方法对参数进行分解估计	卷积层和全连接层
知识蒸馏	搭建一个紧凑网络从大型网络中蒸馏学习	卷积层和全连接层

困难与挑战

- **模型性能尽可能好**
  - 高精度智能化需求
  - 安全性和便捷性

- **模型参数尽可能少**
  - 模型存储轻量化
  - 满足边缘设备算力

- **最好的模型是具有最优的性能和最精简的模型结构。**

南京大学
NANJING UNIVERSITY

第二部分

Proposed Methods

研究内容

- 基于影响函数的CNN剪枝压缩方法
- 基于影响力剪枝和低秩分解的LSTM压缩方法

基于影响函数的CNN剪枝压缩方法：研究动机

- 剪枝**优越性**：最精简结构，广泛应用
- 剪枝**核心问题**：剪枝指标的选择

  之前的工作，指标难以与权重影响力直接相关

  AutoML多需要额外的辅助网络或者代理

寻找合适的指标

经验影响函数可以测量因素F对于函数T的影响力

$$IF(x;T;F) := \lim_{t\to 0^+} \frac{T\left(t\Delta_x + (1-t)F\right) - T(F)}{t}$$

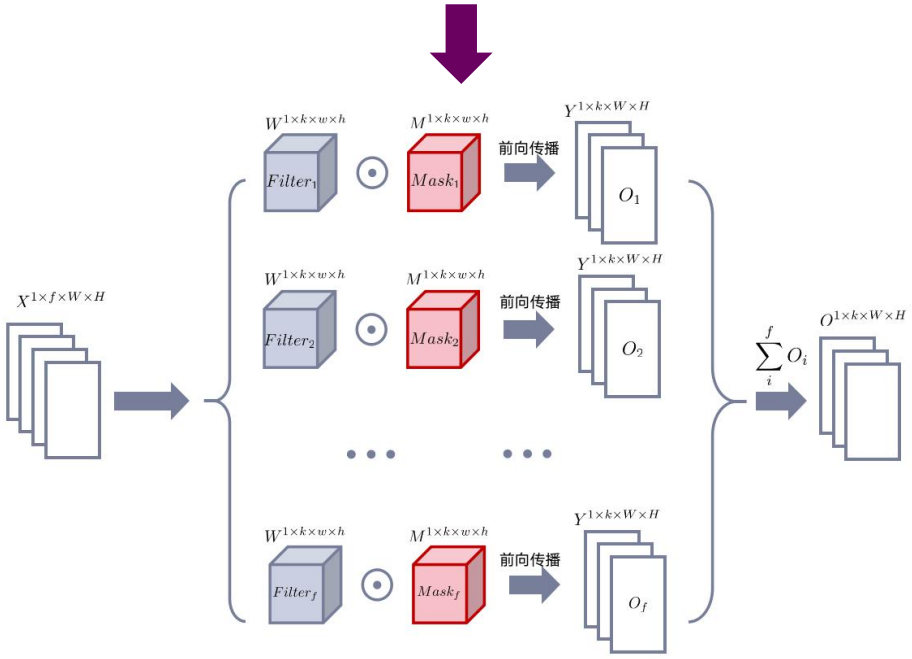
- 对于函数T，加入极小的扰动t，测得参数F对于函数T的影响程度。

- **通过经验影响函数计算**网络中的权重W对于损失函数L的影响力

基于影响函数的CNN剪枝压缩方法：前向传播与反向传播

前向传播

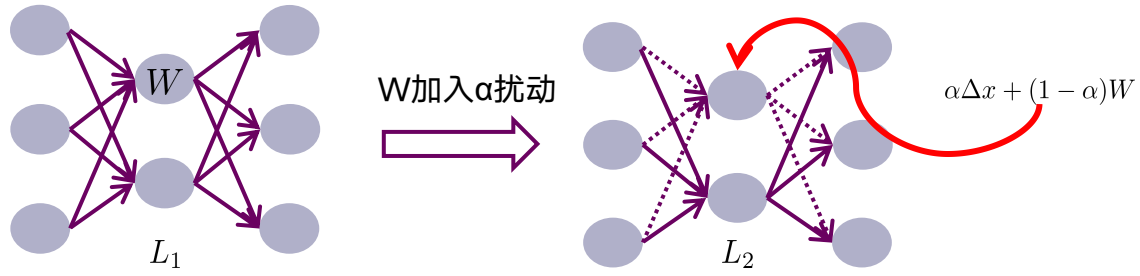
- 在权重矩阵W中加入同尺寸矩阵M
- M矩阵与W权重进行进行Hadamard乘积计算



反向传播

$$IF(x;T;F) := \lim_{t\to 0^+} \frac{T\left(t\Delta_x + (1-t)F\right) - T(F)}{t}$$

相似性

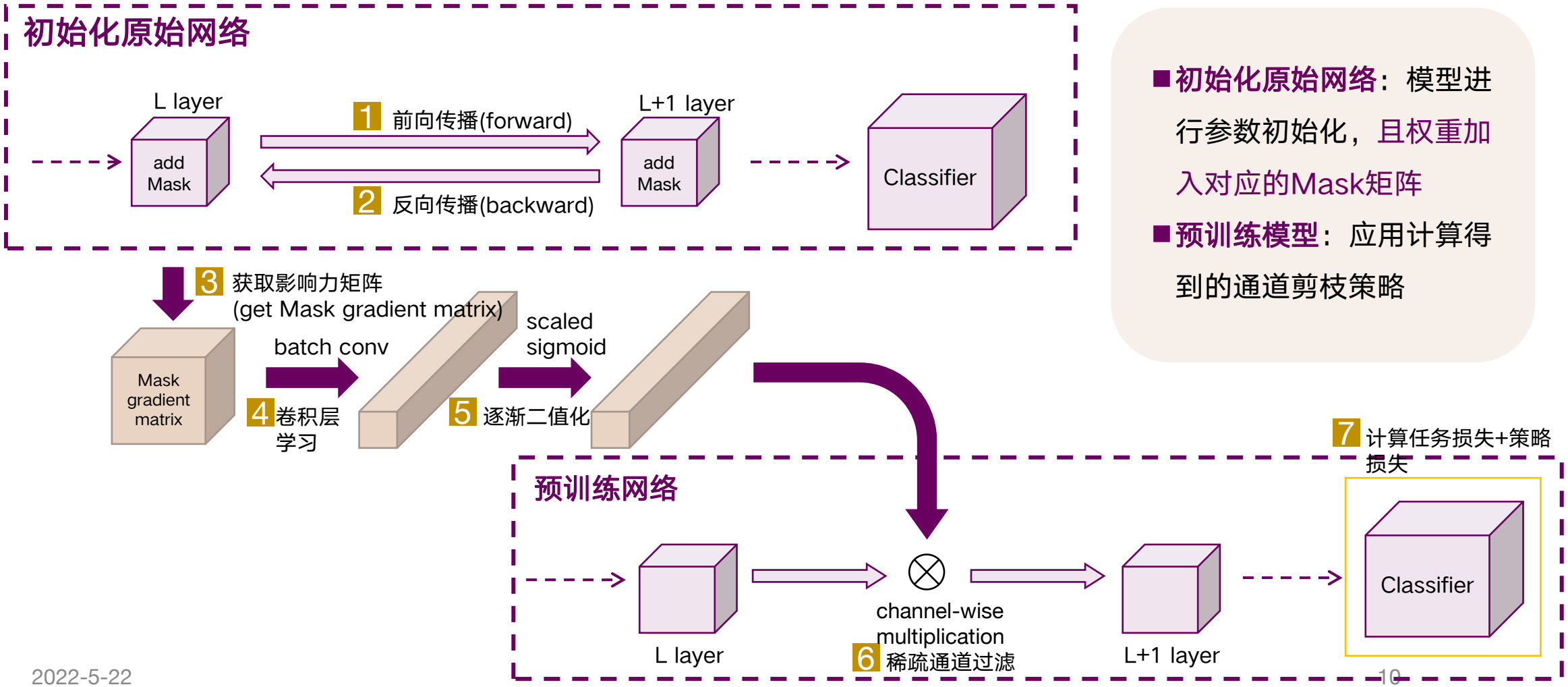


实际公式转化

$$IF(x;L;W) := \lim_{\alpha\to 0^+} \frac{L\left(\alpha\Delta_x + (1-\alpha)W\right) - L(W)}{\alpha}$$

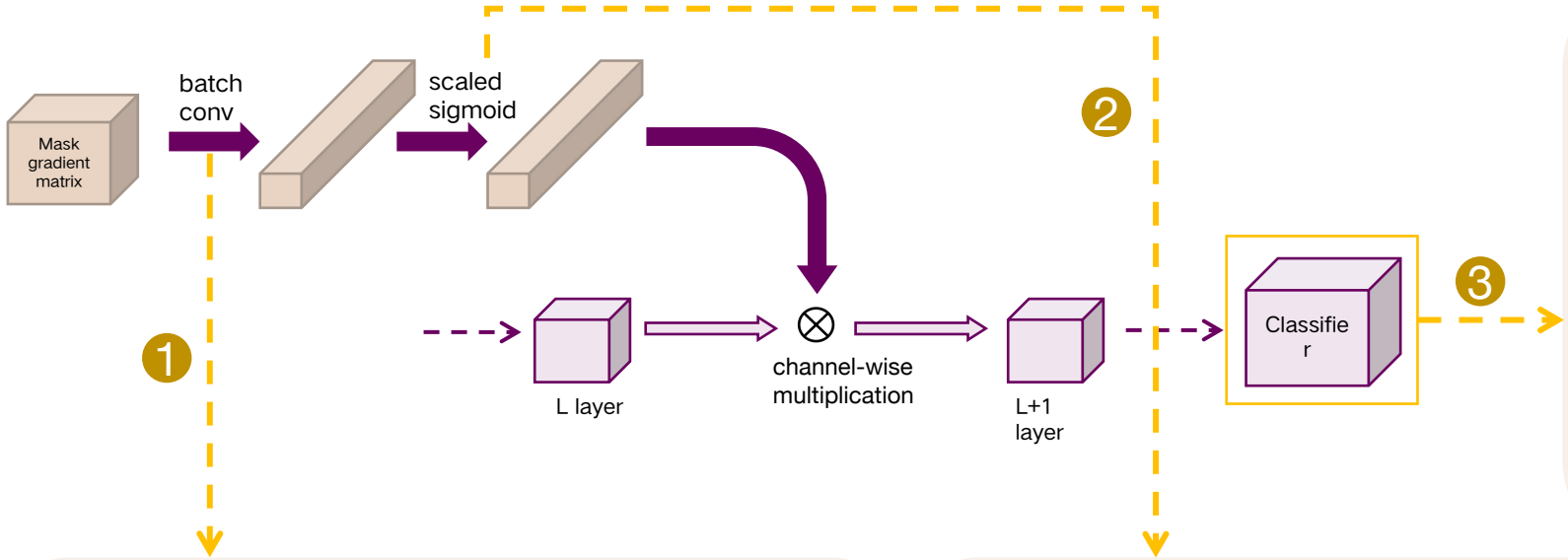
- 对M矩阵进行反向传播求梯度，其梯度结果即为对应权重W的影响力

基于影响函数的CNN剪枝压缩方法：训练流程



- **初始化原始网络**：模型进行参数初始化，且权重加入对应的Mask矩阵
- **预训练模型**：应用计算得到的通道剪枝策略

基于影响函数的CNN剪枝压缩方法：核心要点



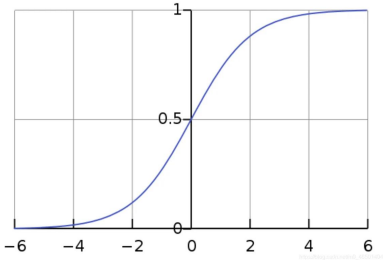
（1）卷积层学习

- 提取影响力特征，缓解不同批数据测量结果存在的偶然性误差
- 根据输入数据大小，使用FC和Conv结构

（2）二值化调控

- 通过sigmoid函数实现二值化映射
- β初始化为较小的值，随着训练逐步增大

$$E_l = \mathrm{Sigmoid}\left(\beta e_l\right)$$



（3）损失函数

$$\mathcal{L} = \mathcal{L}_{\text{classification}} + \lambda \left\| E_l - T_l \right\|_2^2$$

- 兼顾任务损失和剪枝策略损失
- λ与模型的通道压缩率有关

$$\lambda = \begin{cases} 5.0 \times \left| \frac{T_l}{|C_l|} + \frac{B_l}{|C_l|} - 1 \right|, & 1 - \frac{B_l}{|C_l|} \geq \frac{T_l}{C_l} \\ 0, & \text{otherwise} \end{cases}$$

基于影响函数的CNN剪枝压缩方法：对比实验

表 3-2 VGG-16 在 CIFAR-10 上的剪枝算法性能比较

方法	FLOPs Drop Rate	Top-1 Acc. ↑(%)	Params Drop Rate
FPGM[61]	34%	-0.04	-
NS[62]	51%	-0.26	-
NSP[59]	54%	0.04	-
Ours($r = 0.3$)	**57%**	**0.28**	87.69%
Ours($r = 0.4$)	**48%**	**0.07**	80.18%

表 3-3 ResNet-56 在 CIFAR-10 上的剪枝算法性能比较

方法	FLOPs Drop Rate	Top-1 Acc. ↑ (%)	Params Drop Rate
NS[62]	48%	-0.53	-
CP[63]	50%	-1.00	-
CCP[58]	47%	-0.04	-
NSP[59]	47%	0.03	-
Our($r = 0.4$)	**48%**	**0.19**	59.03%

表 3-5 VGG-16 在 CIFAR-100 上的剪枝算法性能比较

模型	FLOPs Drop Rate	Top-1 Acc.↑ (%)	Params Drop Rate
COP[64]	43%	-0.82	-
NS[62]	38%	0.37	-
NSP[59]	43%	0.42	-
Ours($r = 0.2$)	**45%**	**1.08**	73.74%

表 3-6 ResNet-56 在 CIFAR-100 上的剪枝算法性能比较

模型	FLOPs Drop Rate	Top-1 Acc.↑(%)	Params Drop Rate
NS[62]	24%	-1.09	-
NSP[59]	**25%**	-0.06	-
Ours($r = 0.6$)	**25%**	**0.09**	47.70%

■影响力自动剪枝在最大FLOPs下降率，我们方法的压缩性能最好，精度甚至有小幅度提升

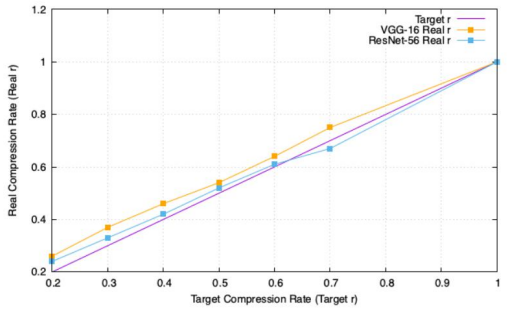
基于影响函数的CNN剪枝压缩方法：消融实验

表 3-7　在 CIFAR-10 数据集中不同目标压缩率 *r* 对 VGG-16 的压缩性能比较

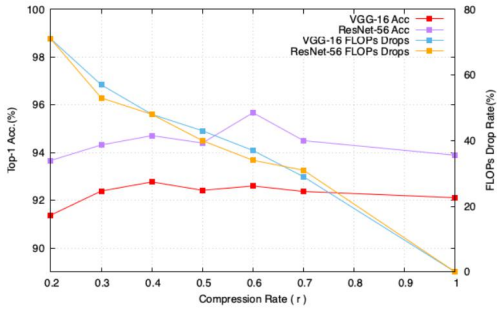
Target Compression Rate	Real Compression Rate	FLOPs	FLOPs Drop Rate	Params	Params Drop Rate	Top-1 Acc.(%)
1.0	-	314.031M	-	14.728M	-	92.11
0.7	0.75	222.696M	29%	8.139M	44.74%	92.37
0.6	0.64	196.987M	37%	5.726M	61.12%	92.60
0.5	0.54	179.204M	43%	4.092M	72.22%	92.42
0.45	0.50	171.454M	45%	3.446M	76.60%	92.29
0.4	0.46	162.574M	48%	2.908M	80.26%	92.77
0.3	0.37	136.123M	57%	1.957M	86.71%	92.39
0.2	0.26	89.581M	71%	1.019M	93.13%	91.38

表 3-8　在 CIFAR-10 数据集中不同目标压缩率 *r* 对 ResNet-56 的压缩性能比较

Target Compression Rate	Real Compression Rate	FLOPs	FLOPs Drop Rate	Params	Params Drop Rate	Top-1 Acc.(%)
1.0	-	1.305G	-	23.521M	-	93.89
0.8	0.74	941.088M	28%	15.404M	34.51%	95.02
0.7	0.67	899.946M	31%	13.593M	42.21%	94.50
0.6	0.61	857.713M	34%	11.978M	49.08%	94.67
0.5	0.52	772.347M	40%	9.635M	59.04%	94.40
0.4	0.42	682.811M	48%	7.754M	67.03%	94.71
0.3	0.33	610.142M	53%	6.255M	73.41%	94.32
0.2	0.24	499.827M	53%	5.013M	78.69%	93.67



(a) 目标压缩率和实际压缩率变化曲线



(b) 不同的目标压缩率在 VGG-16 和 ResNet-56 中的性能变化曲线

- 实际压缩率 > 目标压缩率，模型能够兼顾剪枝策略损失和精度损失
- 不同压缩率其测试Top-1精度基本高于原始模型，
- 该方法存在合适的压缩率，此时精度提升最大

基于影响函数的CNN剪枝压缩方法：消融实验

表 3-9　VGG-16 在 CIFAR-10 中 β 初始值和 Top-1 测试精度效果比较

β	FLOPs	FLOPs Drop Rate	Params	Params Drop Rate	Top-1 Acc.(%)
0.005	163.822M	47%	2.930M	80.11%	92.00
0.01	163.622M	47%	2.927M	80.13%	92.46
0.05	161.244M	48%	2.883M	80.43%	93.05
0.1	162.574M	48%	2.908M	80.26%	92.77
0.2	163.367M	47%	2.915M	80.21%	91.06

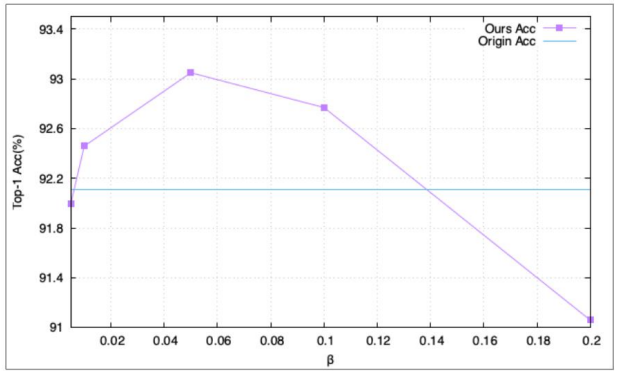
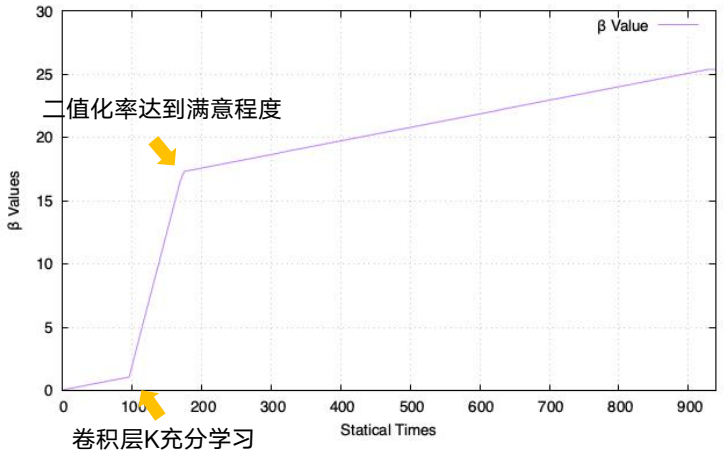


图 3-5　VGG-16 在 CIFAR-10 中 β 初始值和 Top-1 测试精度变化曲线



■β的初始化影响模型剪枝性能，其最优值与模型有关，证明β的初始化的必要性

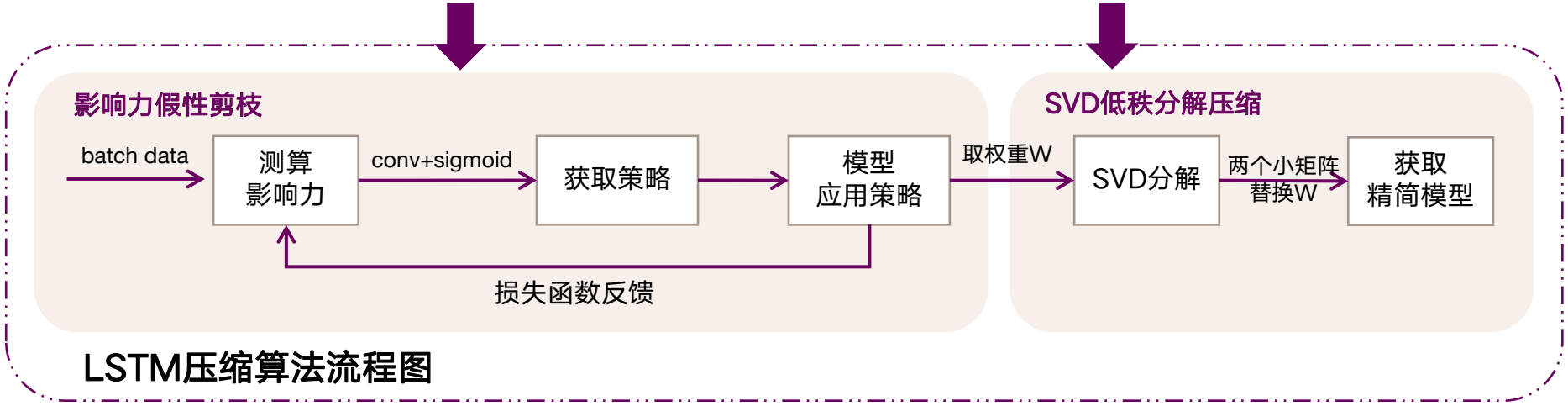
基于影响力剪枝和低秩分解的LSTM压缩方法：研究动机

启发性

- 经验影响函数适用范围广泛
- LSTM模型结构仍然需要反向传播(backward)
- LSTM权重参数仍然存在大量冗余

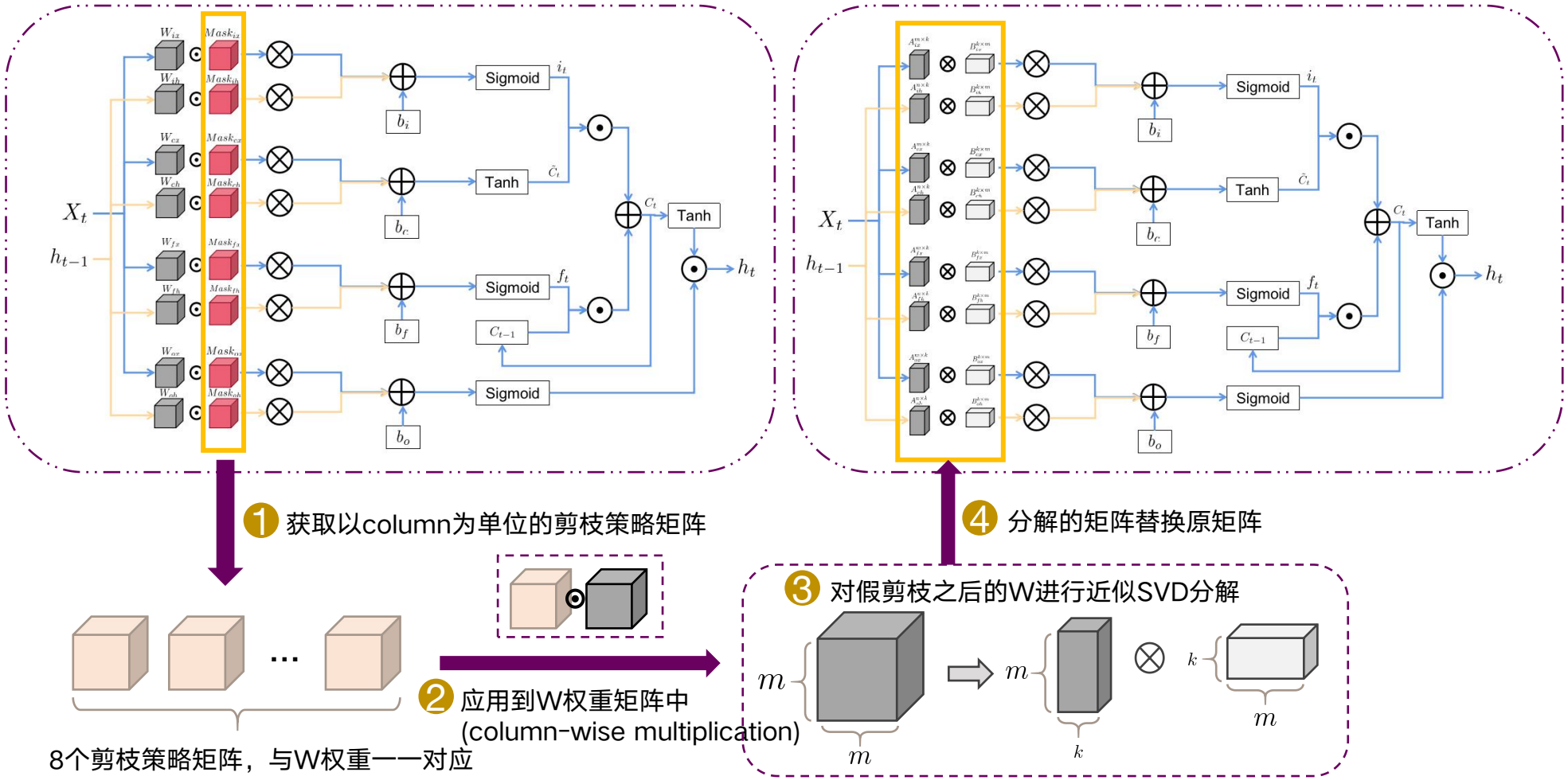
LSTM剪枝压缩问题

- LSTM剪枝压缩通常需要硬件支持
- SVD分解能够对矩阵重构降维
- 稀疏矩阵使用SVD分解，实现近似低秩矩阵

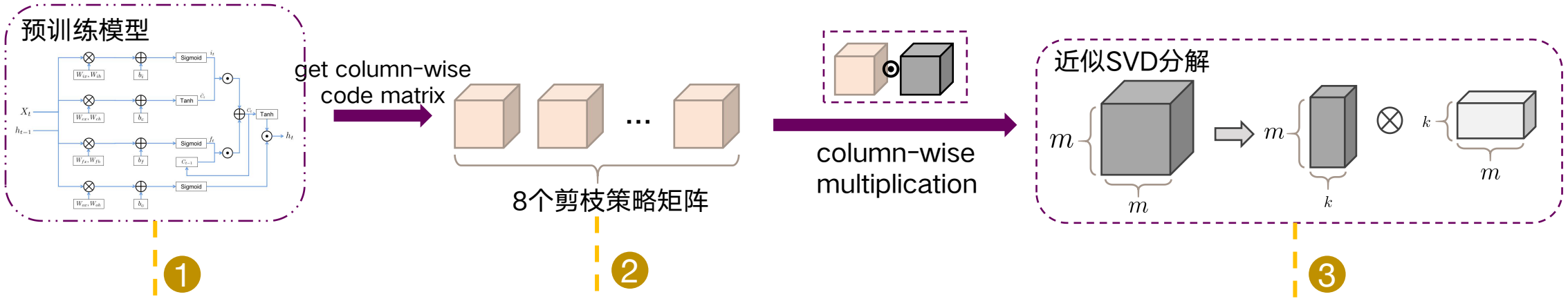


LSTM压缩算法流程图

基于影响力剪枝和低秩分解的LSTM压缩方法：训练流程



基于影响力剪枝和低秩分解的LSTM压缩方法：核心要点



（1）损失函数

■兼顾任务损失和压缩策略损失

$$\mathcal{L} = \mathcal{L}_{\text{classification}} + \lambda \left\| \tilde{B}_{W,S} - B_{W,S} \right\|_2^2$$

■λ根据压缩的实际情况自动变化

$$\lambda = \begin{cases} 5.0 \times \left| \frac{B_{W,S}}{|E_W|} + \frac{\tilde{B}_{W,S}}{|E_W|} - 1 \right|, 1 - \frac{\tilde{B}_{W,S}}{|E_W|} \geq \frac{B_{W,S}}{E_W} \\ 0, \quad \text{otherwise.} \end{cases}$$

（2）剪枝策略矩阵

■剪枝策略矩阵是0和1组成的二进制矩阵

■以column为基本单位，即矩阵每一列全为1或者全为0

（3）近似低秩分解

■设定 *thresh*阈值决定奇异值保留数量

■设定矩阵分解的最大秩，保证参数不增加

■根据上述条件限制，选择最小秩作为所有矩阵低秩分解的秩

基于影响力剪枝和低秩分解的LSTM压缩方法：对比实验

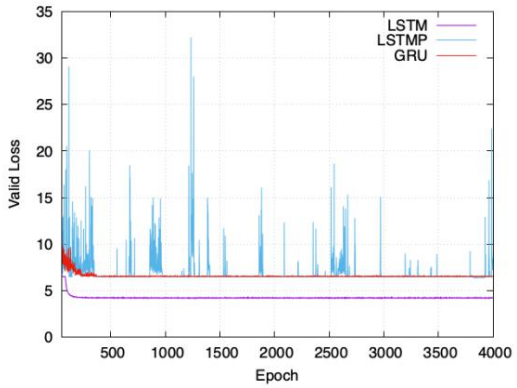
表 4-1 基于 Penn TreeBank 语料库的模型压缩实验效果

模型	Test Loss	PPL	BPC	Params	Params Drop Rate(%)
Origin LSTM	4.15	63.33	5.985	35.442M	-
LSTMP	6.33	563.44	9.138	32.060M	9.54
GRU	6.47	643.44	9.330	27.576M	22.19
Ours(Pruning)	**4.14**	**62.85**	**5.974**	-	-
Ours(Pruning+SVD)	**4.29**	**73.21**	**6.194**	**21.368M**	**39.71**

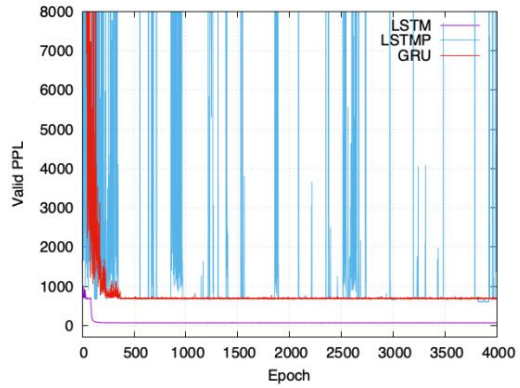
表 4-2 基于 Enwik8 数据集的模型压缩实验效果

模型	Test Loss	PPL	BPC	Params	Params Drop Rate(%)
Origin LSTM	1.29	3.64	1.864	31.514M	-
LSTMP	3.55	34.79	5.121	28.132M	10.73
GRU	4.44	84.46	6.400	23.648M	24.96
Ours(Pruning)	**1.31**	**3.70**	**1.888**	-	-
Ours(Pruning+SVD)	1.59	4.88	2.287	**16.487M**	**47.68**

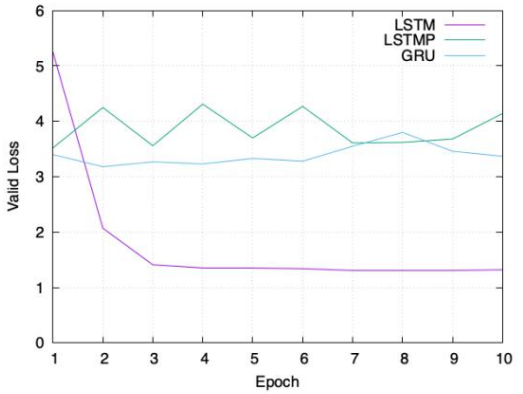
- 更多的参数下降量下，我们方法的精度损失是最少的，证明了我们方法的优越性
- 我们的方法在实验中性能变化最稳定。



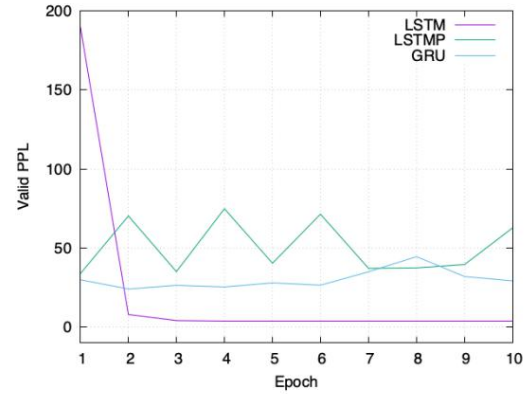
(a) 在 Penn TreeBank 进行 4000 个 Epoch 训练的验证集 Loss 变化曲线



(b) 在 Penn TreeBank 进行 4000 个 Epoch 训练的验证集 PPL 变化曲线



(a) 在 Enwik8 进行 10 个 Epoch 训练的 Valid Loss 变化曲线



(b) 在 Enwik8 进行 10 个 Epoch 训练的 Valid PPL 变化曲线

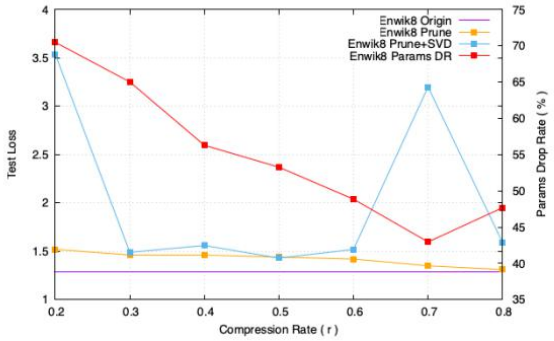
基于影响力剪枝和低秩分解的LSTM压缩方法：消融实验

表 4-3 在 Enwik8 数据集中针对不同压缩率的模型性能比较

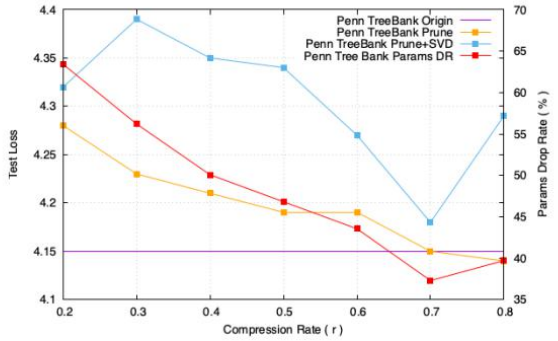
Compression Rate	Prune			Prune+SVD			Params Drop Rate(%)
	Loss	PPL	BPC	Loss	PPL	BPC	
1.0	1.29	3.64	1.864	-	-	-	-
0.8	1.31	3.70	1.888	1.59	4.88	2.287	47.68
0.7	1.35	3.87	1.951	3.20	24.53	4.616	42.96
0.6	1.42	4.12	2.043	1.52	4.55	2.187	48.86
0.5	1.44	4.20	2.071	1.43	4.18	2.063	53.27
0.4	1.46	4.30	2.104	1.56	4.74	2.245	56.29
0.3	1.46	4.30	2.105	1.49	4.44	2.152	65.03
0.2	1.52	4.58	2.194	3.54	34.39	5.104	70.50

表 4-4 在 Penn TreeBank 数据集中针对不同压缩率的模型性能比较

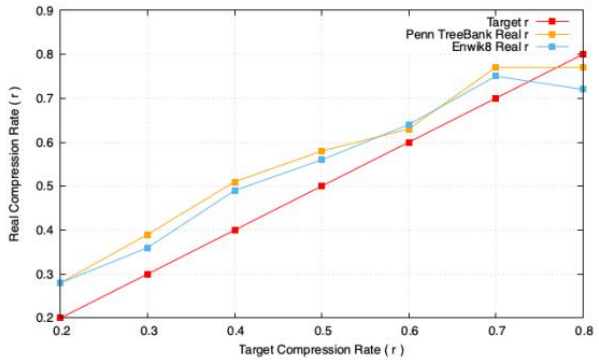
Compression Rate	Prune			Prune+SVD			Params Drop Rate(%)
	Loss	PPL	BPC	Loss	PPL	BPC	
1.0	4.15	63.33	5.985	-	-	-	-
0.8	4.14	62.85	5.974	4.29	73.21	6.194	39.71
0.7	4.15	63.63	5.992	4.18	65.61	6.036	37.30
0.6	4.19	66.35	6.052	4.27	71.73	6.164	43.55
0.5	4.19	66.11	6.047	4.34	77.08	6.268	46.81
0.4	4.21	67.04	6.067	4.35	77.51	6.276	50.07
0.3	4.23	69.00	6.109	4.39	80.27	6.327	56.20
0.2	4.28	71.95	6.169	4.32	74.85	6.226	63.40



(a) 在 Enwik8 任务中不同 *r* 的性能变化曲线



(b) 在 Penn TreeBank 任务中不同 *r* 的性能变化曲



(c) 在不同 *r* 下的实际压缩率变化曲线

- 参数下降率随着r增加逐步下降，说明我们方法的有效性
- 通过影响力剪枝造成的损失极小，证明了影响力剪枝的优越性
- SVD分解免除了硬件支持，证明了SVD的必要性

基于影响力剪枝和低秩分解的LSTM压缩方法：消融实验

β_x和β_h二值化调控

表 4-5 不同的 β_x 和 β_h 的初始化值对剪枝模型性能影响效果比较

β_x 和 β_h 的初始化值	Prune		
	Loss	PPL	BPC
0.01	4.20	66.44	6.054
0.02	4.20	66.38	6.053
0.05	4.20	66.40	6.053
0.08	4.20	66.48	6.055
0.10	4.19	66.11	6.047
0.12	4.19	66.13	6.047
0.15	4.18	65.67	6.037
0.20	4.19	65.93	6.043

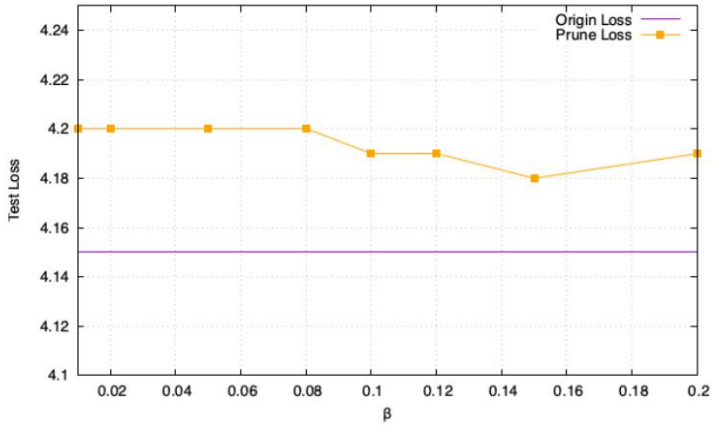
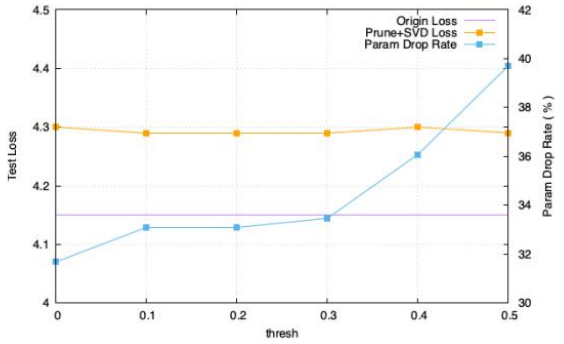
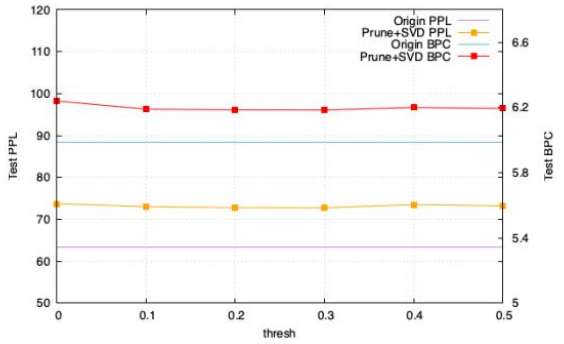


图 4-13 不同的 β_x 和 β_h 对剪枝模型性能影响曲线

*thresh*阈值



(a) 删减的参数量变化和 Test Loss 变化曲线



(b) Test PPL 和 BPC 变化曲线

■我们的方法设定βx 和 βh能够激发模型最优性能，证明了βx 和 βh 初始化的必要性

■*thresh*处于较小范围中，模型损失量变化不敏感，证明了我们方法的鲁棒性

南京大学
NANJING UNIVERSITY

第三部分

Applications

实际应用

- 网络加速在多通道语音增强系统的应用

多通道语音增强系统：系统需求

- 边缘计算能力强
  - 减少信息流传输、降低部署成本
- 非平稳噪声的抑制能力强
  - 保证语音数据纯净，且不失真
- 模型迭代能力高效
  - 应当尽可能保证系统稳定

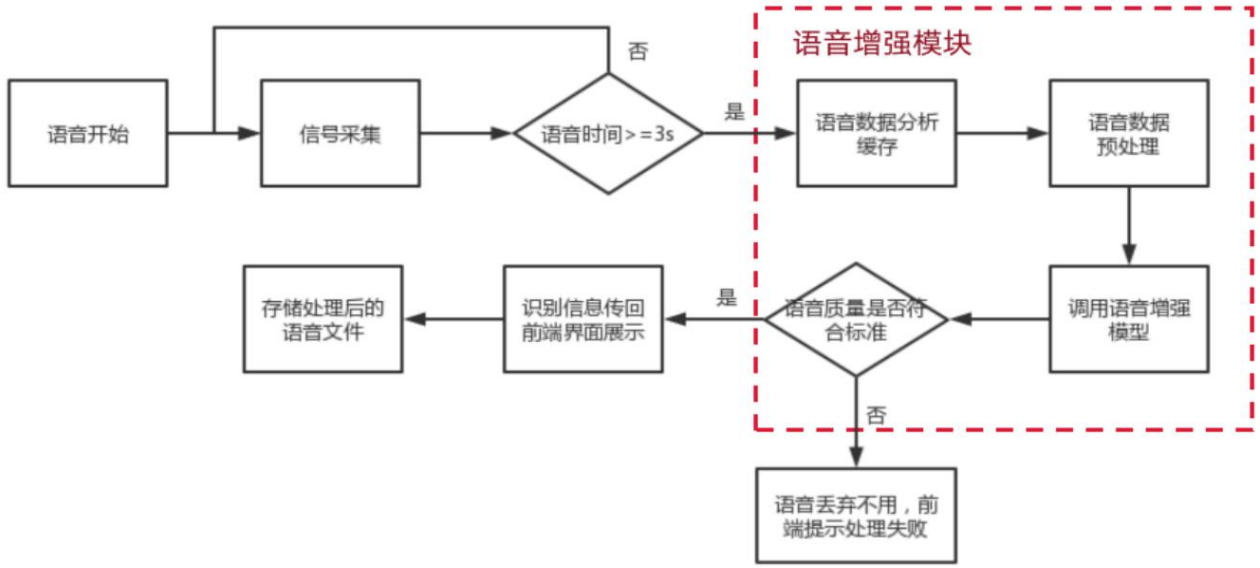
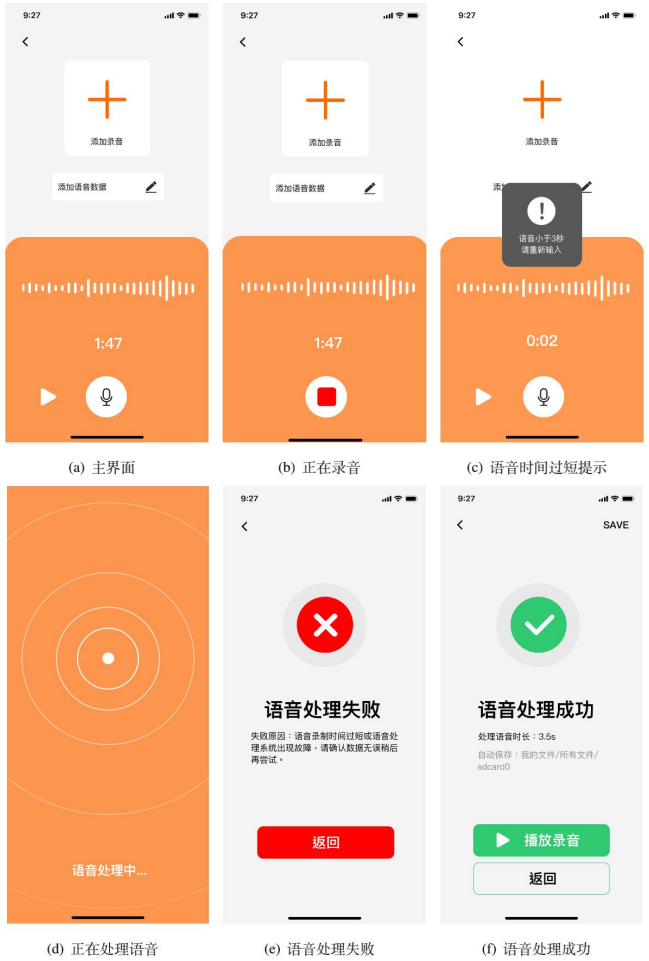


图 5-1　语音增强系统流程图

既要保证语音增强性能又要保证模型轻量化，能够部署于小型设备中

多通道语音增强系统：系统效果



(a) 主界面　(b) 正在录音　(c) 语音时间过短提示

(d) 正在处理语音　(e) 语音处理失败　(f) 语音处理成功

表 5-1　影响力剪枝及 SVD 分解压缩在语音增强系统中的应用效果

方法	SI_SDR	STOI	PESQ	Params	RTF
FullSubNet	20.34	0.9208	2.5201	6.32M	1.02
FullSubNet(影响力剪枝 +SVD 分解)	19.52	0.9156	2.4451	**1.39M**	**0.81**
TCN	20.52	0.9244	2.7129	6.16M	0.99
TCN(影响力剪枝 +SVD 分解)	**20.57**	**0.9267**	**2.7373**	**0.81M**	**0.41**

- RTF下降明显，实现了实时性系统运行
- 模型性能下降在可接受范围内，甚至有部分模型性能提升
- 参数量大幅下降，完全可以在小型设备中部署

表明了我们方法的具有**实际应用价值。**

研究生期间工作总结 | Summary

论文

- **Lai Bilan**, Xiang Haoran, Shen Furao, “Inf-CP: A Reliable Channel Pruning based on Channel Influence”

专利

- 申富饶，赖碧兰，安俊逸，赵健. “一种复杂环境下的三维定位追踪方法”(202010927152.1)
- 申富饶，赖碧兰，赵健. “用于室内定位的基于实时轨迹动态进行二维跳点修正方法”(202110047580.X)

项目

- 国家自然科学基金“基于深度感知增量式联想记忆神经网络的信息融合系统研究”

荣誉

- 南京大学二等学业奖学金

南京大学
NANJING UNIVERSITY

第四部分

Summary

总结与展望

全文工作总结

基于影响函数的CNN剪枝压缩方法

- 影响函数测算权重影响力
- 训练模式：微调+剪枝
- 损失函数监督剪枝策略

基于影响力剪枝和低秩分解的LSTM 压缩方法

- 模型结构分析
- 利用影响函数进行剪枝
- SVD低秩分解替代硬件

网络加速在多通道语音增强系统的应用

- 分析系统需求
- 对CNN、FC和LSTM结构进行影响力剪枝
- 对LSTM进行SVD分解重构权重矩阵

未来工作展望

超参β的初始化设计

- β初始化难以自动调整
- 考虑使用更智能的二值化过程

对更多结构的压缩进行扩展

- 将影响力剪枝多结构推广

集成模型剪枝的工具包

- 考虑将剪枝算法集成为工具包
- 实现端到端压缩模型

南京大学
NANJING UNIVERSITY

谢谢大家！