

# TextCNN 论文分享

论文: <https://arxiv.org/pdf/1408.5882.pdf>

代码: <https://github.com/649453932/Chinese-Text-Classification-Pytorch>

以若

# Convolutional Neural Networks for Sentence Classification

**Yoon Kim**

New York University

yhk255@nyu.edu

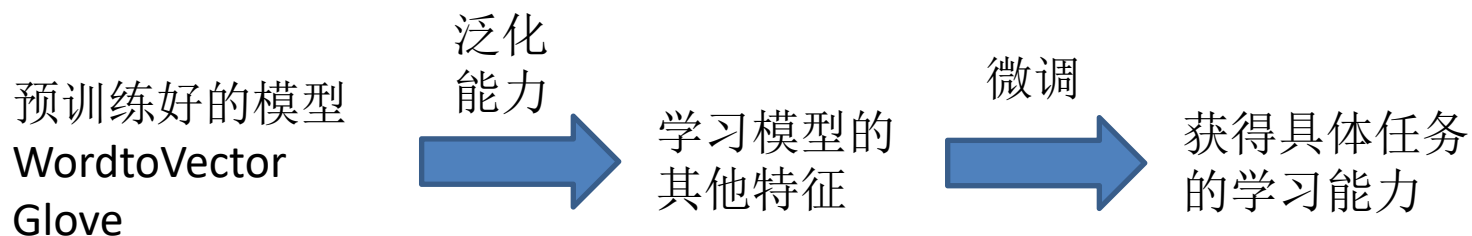
## 目录

1. Introduction
2. 模型架构
3. 训练流程
4. 总结
5. 实验结果

# Introduction

灵感：受启发于图像分类中，从一个预训练好的模型中学到的特征能够应用于不同分类的任务中。

主要思路：



# 模型架构

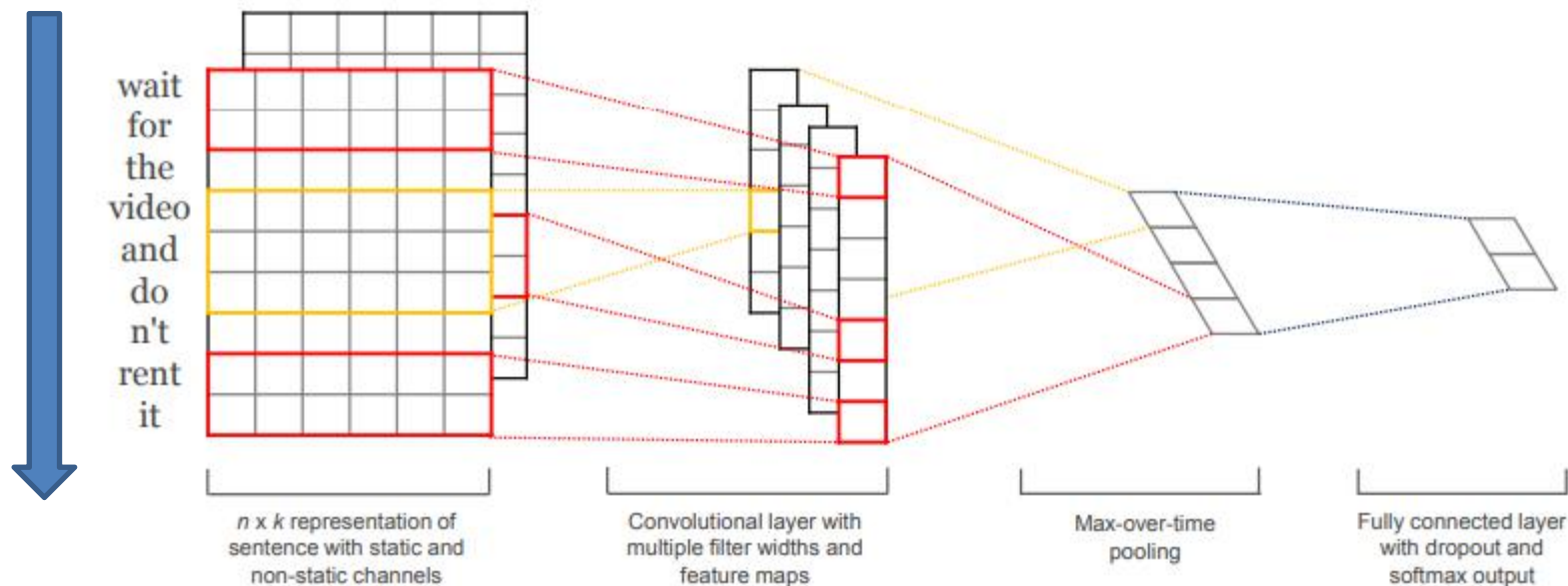


Figure 1: Model architecture with two channels for an example sentence.

**Embedding layer**

(WordToVector  
或者Glove的输出)

卷积输出

结果拼接

最大池化

全连接层

dropout

0轴: 单词

1轴: 维度 (固定)

# 模型架构—关键点

## **CNN-rand**

作为一个基础模型，Embedding layer所有words被随机初始化，然后模型整体进行训练

## **CNN-static**

模型使用预训练的word2vec初始化Embedding layer，对于那些在预训练的word2vec没有的单词，随机初始化。然后固定Embedding layer，fine-tune整个网络

## **CNN-non-static**

训练的时候，Embedding layer跟随整个网络一起训练

## **CNN-multichannel**

两个channel，一个channel为static，一个为non-static。然后整个网络fine-tune时只有一个channel更新参数。两个channel都是使用预训练的word2vec初始化的

# 训练流程

## Word Embedding

今天” -> [0,0,0,0,1] “天气” ->[0,0,0,1,0] “很好” ->[0,0,1,0,0]

今天	0	0	0	0	1
天气	0	0	0	1	0
很好	0	0	1	0	0
,	0	1	0	0	0
出来	1	0	0	0	0
玩	0	0	0	1	1

- 每个词都可以表征成一个向量，得到一个嵌入矩阵M
- M里的每一行都是词向量。
- M可以是静态(static)的，也就是固定不变。
- M可以是非静态(non-static)的，也就是可以根据反向传播更新

# 训练流程

## Convolution 卷积

3X5

4X5

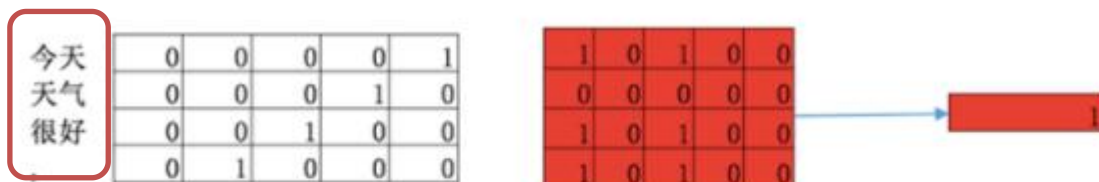
5X5

100个

输入词向量矩阵

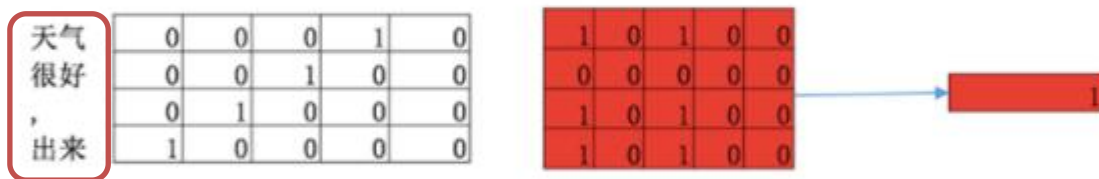
卷积核

Feature map



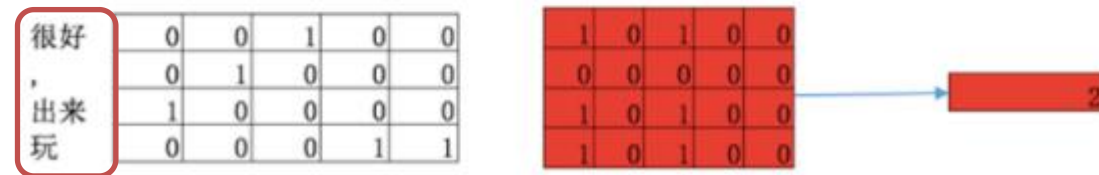
Step .1

输入4\*5



输出3\*1

Step .2



Step .3

# 训练流程

## Channel

channel

1	0	1	0	0
0	0	0	0	0
1	0	1	0	0
1	0	1	0	0

Channel 1

Static

0	0	1	0	0
0	0	0	0	0
1	0	1	0	0
0	0	0	0	0

Channel 2

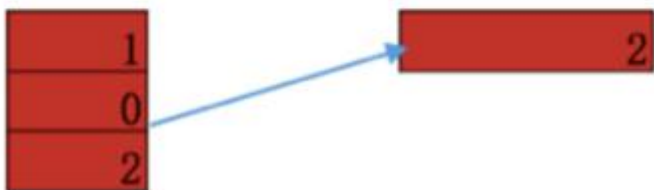
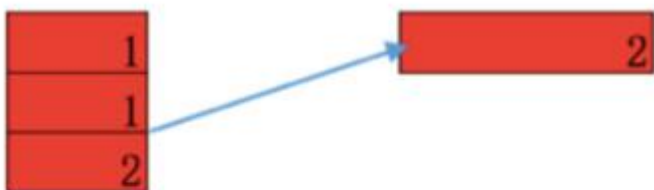
No-static

- 引入channel 是希望防止过拟合--通过保证学习到的vectors 不要偏离输入太多)来在小数据集合获得比单channel更好的表现，后来发现其实直接使用正则化效果更好。
- 可以在no-static(梯度可以反向传播) 的channel 来fine tune 词向量，让词向量更加适用于当前的训练



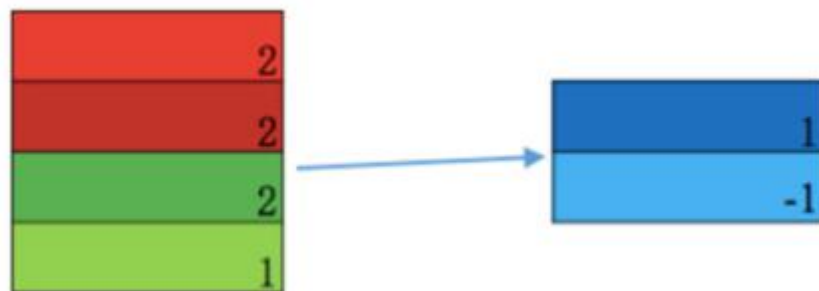
# 训练流程

## Pooling 池化



- 保持平移不变性
- 减少特征维度，防止过拟合

## 使用softmax k分类



- 全连接层
- 正则化

# 实验结果

Model	MR	SST-1	SST-2	Subj	TREC	CR	MPQA
CNN-rand	76.1	45.0	82.7	89.6	91.2	79.8	83.4
CNN-static	81.0	45.5	86.8	93.0	92.8	84.7	<b>89.6</b>
CNN-non-static	<b>81.5</b>	48.0	87.2	93.4	93.6	84.3	89.5
CNN-multichannel	81.1	47.4	<b>88.1</b>	93.2	92.2	<b>85.0</b>	89.4
RAE (Socher et al., 2011)	77.7	43.2	82.4	—	—	—	86.4
MV-RNN (Socher et al., 2012)	79.0	44.4	82.9	—	—	—	—
RNTN (Socher et al., 2013)	—	45.7	85.4	—	—	—	—
DCNN (Kalchbrenner et al., 2014)	—	48.5	86.8	—	93.0	—	—
Paragraph-Vec (Le and Mikolov, 2014)	—	<b>48.7</b>	87.8	—	—	—	—
CCAE (Hermann and Blunsom, 2013)	77.8	—	—	—	—	—	87.2
Sent-Parser (Dong et al., 2014)	79.5	—	—	—	—	—	86.3
NBSVM (Wang and Manning, 2012)	79.4	—	—	93.2	—	81.8	86.3
MNB (Wang and Manning, 2012)	79.0	—	—	<b>93.6</b>	—	80.0	86.3
G-Dropout (Wang and Manning, 2013)	79.0	—	—	93.4	—	82.1	86.1
F-Dropout (Wang and Manning, 2013)	79.1	—	—	<b>93.6</b>	—	81.9	86.3
Tree-CRF (Nakagawa et al., 2010)	77.3	—	—	—	—	81.4	86.1
CRF-PR (Yang and Cardie, 2014)	—	—	—	—	—	82.7	—
SVM <sub>S</sub> (Silva et al., 2011)	—	—	—	—	<b>95.0</b>	—	—

# 总结

## 优势

- 1.在预训练好的基础上，提取公共特征，再根据具体任务再次学习。
- 2.最大池化可以减少特征维度，有利于更快收敛
- 3.网络结构简单，一次卷积、一次池化