text-CNN模型相关

笔记本:	自然语言处理		
创建时间:	2018/10/4 15:36	**更新时间:**	2018/10/4 15:56
作者:	beyourselfwb@163.com		

原始论文：2014 EMNLP Convolutional neural networks for sentence classification
作者：Yoon Kim, New York University, yhk255@nyu.edu

Keras实现：
https://github.com/hongweijun811/wjgit/blob/master/text_cnn_demo.py
博客：
http://www.tensorflownews.com/2018/04/06/%E4%BD%BF%E7%94%A8keras%E8%BF%9I
cnn%E5%A4%84%E7%90%86%E8%87%AA%E7%84%B6%E8%AF%AD%E8%A8%80/

博客：http://www.wildml.com/2015/12/implementing-a-cnn-for-text-classification-in-tensorflow/
Tensorflow实现：https://github.com/rxt2012kc/cnn-text-classification-tf

链接：https://aclanthology.coli.uni-saarland.de/papers/D14-1181/d14-1181

主要工作：实现了四个利用CNN进行句子层面的文本分类工作，分别是CNN-rand、CNN-static、CNN-non-static、CNN-multichannel。

创新点：
1、实验采用预先训练好的词向量；
2、对CNN网络结构做了改动，能够同时使用task-specific and static vectors

背景信息：深度学习模型在计算机视觉(2012)和语音识别(2013)方面取得了喜人的成绩。在NLP领域好几篇研究都是用语言模型来学习word vector representation的，2003 2011 2013的三篇论文。CNN在语义解析semantic parsing、搜索查询检索search query retrieval、句子建模sentence modeling 方面被证明很有效(2014)。
使用了预训练的word2vector--用一千亿的Google News训练出的word vectors，适用于大部分分类任务。可以从这里下载(https://code.google.com/p/word2vec/)

实验设计：数据集说明

Data	c	l	N	$\|V\|$	$\|V_{pre}\|$	*Test*
MR	2	20	10662	18765	16448	CV
SST-1	5	18	11855	17836	16262	2210
SST-2	2	19	9613	16185	14838	1821
Subj	2	23	10000	21323	17913	CV
TREC	6	10	5952	9592	9125	500
CR	2	19	3775	5340	5046	CV
MPQA	2	3	10606	6246	6083	CV

Table 1: Summary statistics for the datasets after tokenization. c: Number of target classes. l: Average sentence length. N: Dataset size. $|V|$: Vocabulary size. $|V_{pre}|$: Number of words present in the set of pre-trained word vectors. *Test*: Test set size (CV means there was no standard train/test split and thus 10-fold CV was used).

模型实验结果对比

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	**89.6**
CNN-non-static	**81.5**	48.0	87.2	93.4	93.6	84.3	89.5
CNN-multichannel	81.1	47.4	**88.1**	93.2	92.2	**85.0**	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)	–	**48.7**	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	–	–	**93.6**	–	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	–	–	**93.6**	–	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_S (Silva et al., 2011)	–	–	–	–	**95.0**	–	–

Table 2: Results of our CNN models against other methods. **RAE**: Recursive Autoencoders with pre-trained word vectors from Wikipedia (Socher et al., 2011). **MV-RNN**: Matrix-Vector Recursive Neural Network with parse trees (Socher et al., 2012). **RNTN**: Recursive Neural Tensor Network with tensor-based feature function and parse trees (Socher et al., 2013). **DCNN**: Dynamic Convolutional Neural Network with k-max pooling (Kalchbrenner et al., 2014). **Paragraph-Vec**: Logistic regression on top of paragraph vectors (Le and Mikolov, 2014). **CCAE**: Combinatorial Category Autoencoders with combinatorial

case study: 用典型的例子来说明模型确实有效

	Most Similar Words for	
	Static Channel	Non-static Channel
bad	*good* *terrible* *horrible* *lousy*	*terrible* *horrible* *lousy* *stupid*
good	*great* *bad* *terrific* *decent*	*nice* *decent* *solid* *terrific*
n't	*os* *ca* *ireland* *wo*	*not* *never* *nothing* *neither*
!	*2,500* *entire* *jez* *changer*	*2,500* *lush* *beautiful* *terrific*
,	*decasia* *abysmally* *demise* *valiant*	*but* *dragon* *a* *and*

Table 3: Top 4 neighboring words—based on cosine similarity—for vectors in the static channel (left) and fine-tuned vectors in the non-static channel (right) from the multichannel model on the SST-2 dataset after training.

实验方法：控制变量法，列出变量有哪些，如何控制其他条件保持一致

实验结果：
1、CNN-rand：所有的词向量都是随机初始化的，在随机训练过程中作为参数进行调整；
2、CNN-static：所有的词向量都是预先通过word2vector训练好的(未知词汇则采用随机初始化的方法)，并且在实验过程中保持不变，只调整其他参数；
3、CNN-non-static：使用预先训练的词向量并在训练过程中调整；
4、CNN-multichannel：采用两组词向量，每组词向量作为一个通道，每组滤波器都同时作用在两个通道上，但反向传播时只更新其中一个通道，另一个保持不变。

专业词汇：euclidean distance欧几里得距离, cosine distance余弦距离，the element-wise multiplication operator点乘，逐个元素乘积之和，benchmark 一种规则，baseline基线 参照，比如当前f1-score是0.90，那么以这个为基准进行优化提升