

卷积神经网络的多视角句子相似度建模

Multi-Perspective Sentence Similarity Modeling
with Convolutional Neural Networks

汇报人：黑桃

目录

- 一 相关介绍
- 二 动机
- 三 模型介绍
- 四 实验结果及分析



相关介绍



相关介绍

文本相似度：

两个文本（文章）之间的相似度

应用：

在搜索引擎、推荐系统、论文鉴定、机器翻译、自动应答、命名实体识别、拼写纠错等领域有广泛的应用。

文本距离：

简单共有词、欧几里德距离、曼哈顿距离、切比雪夫距离、余弦距离、汉明距离、杰卡德相似度、编辑距离



动机



动机

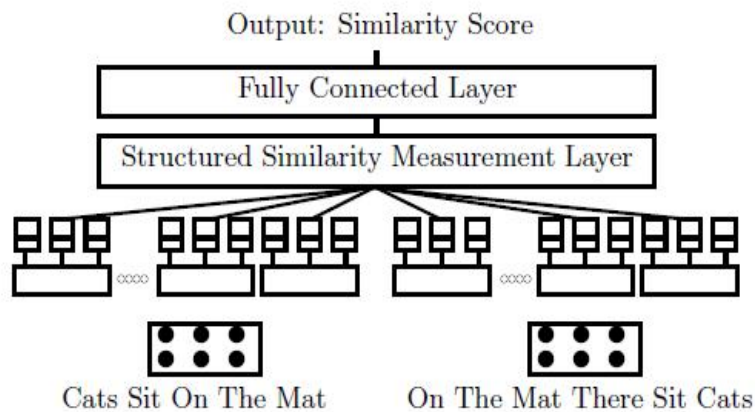
语言表达的模糊性和可变性使句子相似性建模变得复杂。



模型思路



模型介绍



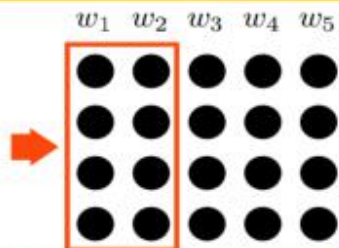
1. 两个输入句子（在底部）由相同的神经网络并行处理，输出句子表示。
2. 句子表示由结构化相似性度量层进行比较。
3. 将相似性特征传递到完全连接的层，以计算相似性得分（顶部）。



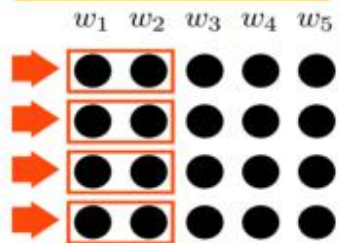
模型介绍——句子表示层

two types of convolution filters

Building Block A:
holistic (all dimensions)

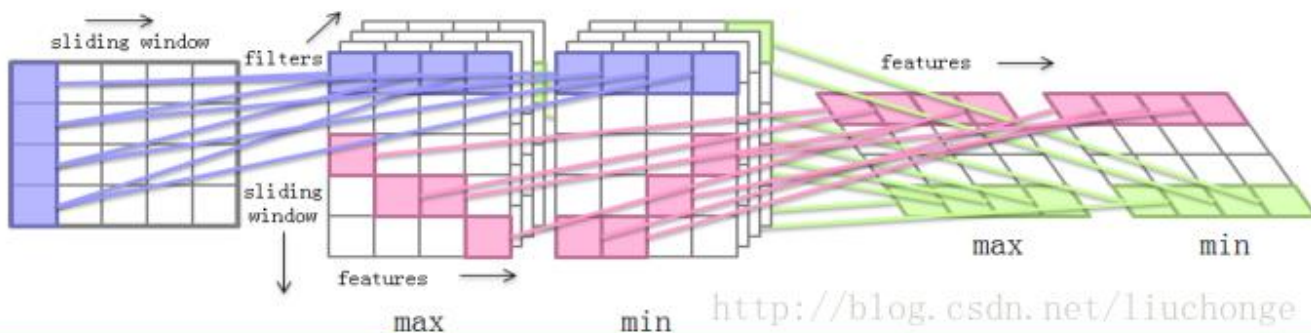
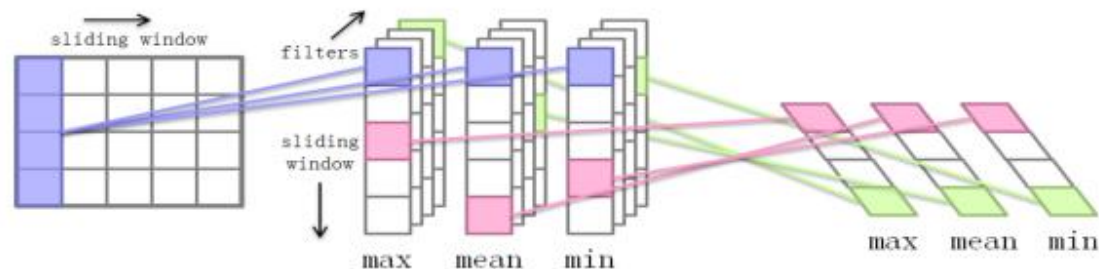


Building Block B:
per-dimension



three types of pooling: max/min/mean

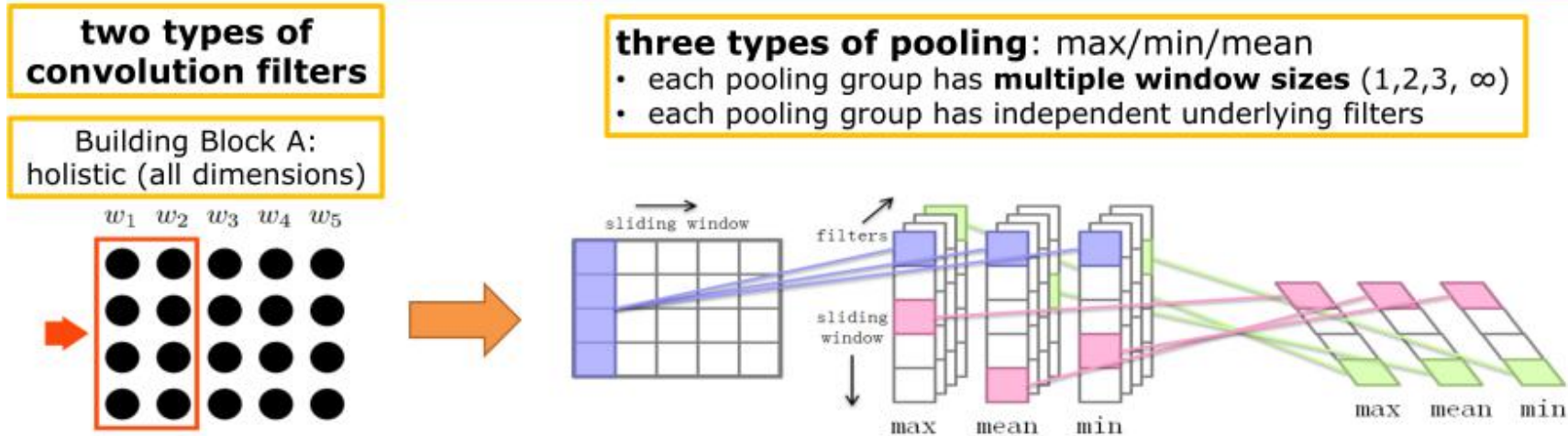
- each pooling group has **multiple window sizes** (1, 2, 3, ∞)
- each pooling group has independent underlying filters



<http://blog.csdn.net/liuchonge>



模型介绍——句子表示层



整体卷积核

$$out_F[i] = h_F(w_F \cdot sent_{i:i+ws-1} + b_F) \quad (1)$$

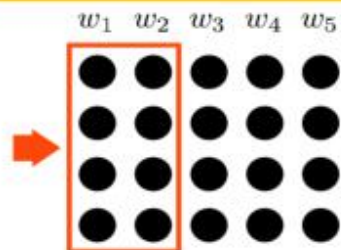
输入是一个单词流，把句子可以理解为时间序列，邻近的单词具有关联性，那么 $sent \in R^{len \times dim}$ 是长度为len，单词维度为dim维的一个句子，其中 $sent_i \in R^{Dim}$ ，表示第i个单词的embedding，而 $sent_{i,j}$ 表示从i到j包括j的连接embedding， $sent_i^{[k]}$ 表示第i个单词向量的第k维度



模型介绍——句子表示层

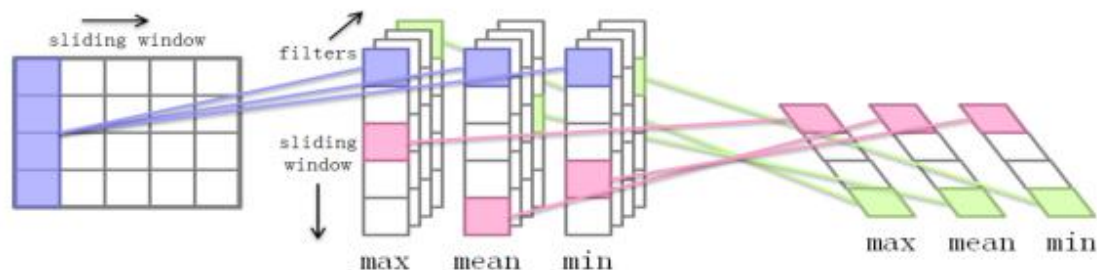
two types of convolution filters

Building Block A:
holistic (all dimensions)



three types of pooling: max/min/mean

- each pooling group has **multiple window sizes** (1, 2, 3, ∞)
- each pooling group has independent underlying filters



输入: [batch_size, seq_len, embed_size]

卷积核: [ws, embed_size, num_filters_A]

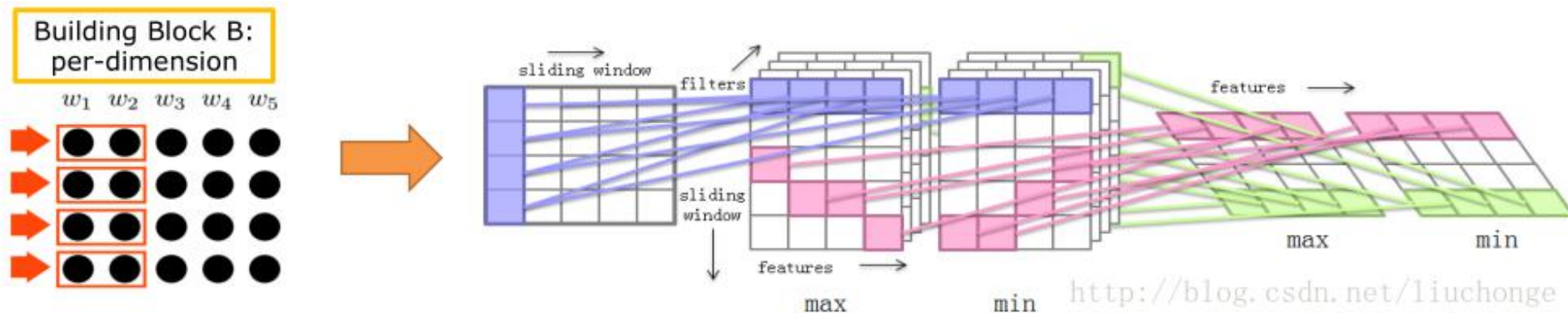
输出: [batch_size, 1, seq_len-ws+1, num_filters_A]

ws: (1, 2, ∞)待调参 图中为1

num_filters_A:待调参 图中为4



模型介绍——句子表示层



单维卷积核

$$out_{F[k]}[i] = h_{F[k]}(w_{F[k]} \cdot sent_{i:i+ws-1}^{[k]} + b_{F[k]})$$

输入: [batch_size, seq_len, embed_size, 1]

卷积核: [ws, embed_size, 1, num_filters_A](多个卷积核)

输出: [batch_size, seq_len-ws+1, 1, num_filters_B]

ws: (1, 2)待调参 图中为1

num_filters_A:待调参 图中为4



模型介绍——句子表示层

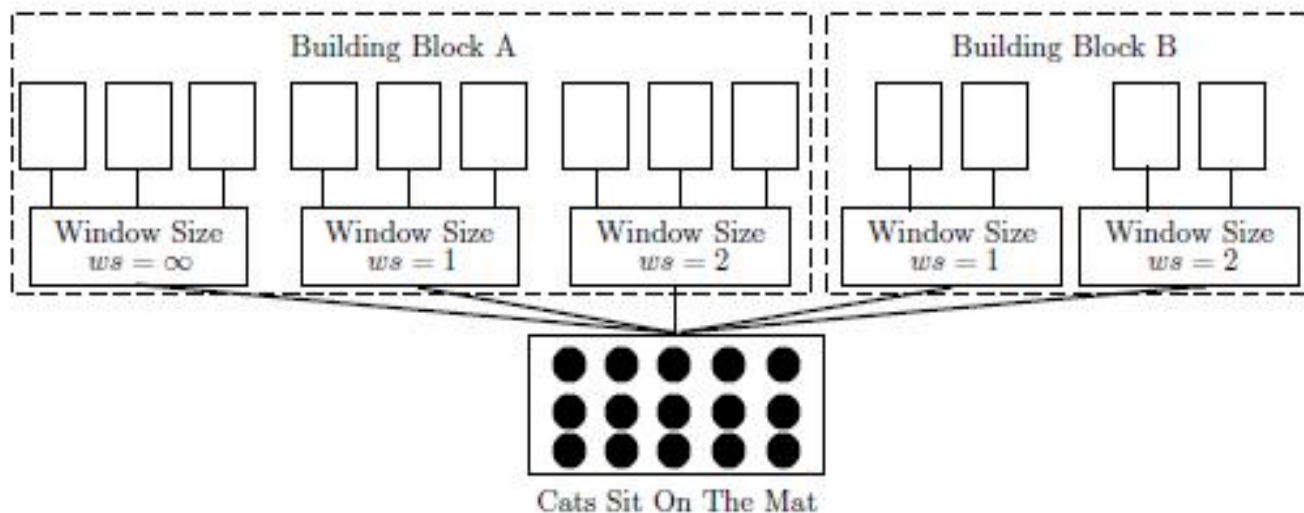


图4：单个句子的示例神经网络架构，包含3个blockA实例（具有3种类型的池）和2个blockB实例（具有2种类型）在变化的窗口尺寸 $ws = 1, 2$ 和 $ws = co$ 上； blockA对整个字向量进行操作，而blockb包含对各个维度进行独立操作的过滤器。



模型介绍——相似性计算层

三种计算距离的方式:余弦距离, L1距离, L2距离

组合成两种距离计算函数:

$$comU_1(\vec{x}, \vec{y}) = \{\cos(\vec{x}, \vec{y}), L_2Euclid(\vec{x}, \vec{y}), |\vec{x} - \vec{y}|\} \quad (3)$$

$$comU_2(\vec{x}, \vec{y}) = \{\cos(\vec{x}, \vec{y}), L_2Euclid(\vec{x}, \vec{y})\} \quad (4)$$

在输入经过不同的卷积层, 池化层之后, 会得到数据的结果, 我们不能简单的把所有的结果展开并拼接在一起, 组成一个大的向量, 然后计算相似度. 我们要考虑结果来源的相似程度, 具体来说, 从以下四个角度判断:

- 结果是否来自同一个block, 即同一个输入, 同一种卷积核长度, 区别只在于池化层不同
- 结果是否来自同一个卷积核长度
- 结果是否来自同一个池化层
- 结果是否来自相同的通道, 可以是不同卷积核

以上四种衡量标准对于两种卷积核是分开的, 即相互之间不比较. 而且计算相似度时独立.



模型介绍——相似性计算层

Algorithm 1 Horizontal Comparison

```
1: for each pooling  $p = \text{max, min, mean}$  do
2:   for each width  $ws_1 = 1 \dots n, \infty$  do
3:      $regM_1[*][ws_1] = group_A(ws_1, p, S_1)$ 
4:      $regM_2[*][ws_1] = group_A(ws_1, p, S_2)$ 
5:   end for
6:   for each  $i = 1 \dots numFilter_A$  do
7:      $fea_h = comU_2(regM_1[i], regM_2[i])$ 
8:     accumulate  $fea_h$  for final layer
9:   end for
10: end for
```

6-9行就是遍历输出的每一行，计算两个句子的相似性（即max到max），然后对每个池化方式做相同处理即可。绿色框也不止仅代表第一行，应该下面每一行都有。



模型介绍——相似性计算层

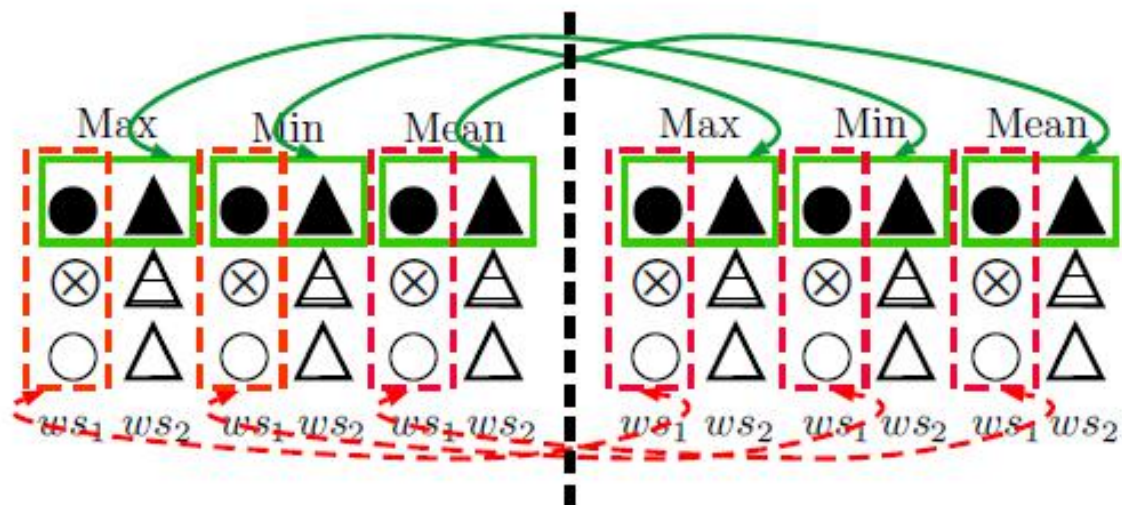
Algorithm 2 Vertical Comparison

```
1: for each pooling  $p = \max, \min, \text{mean}$  do
2:   for each width  $ws_1 = 1 \dots n, \infty$  do
3:      $oG_{1A} = \text{group}_A(ws_1, p, S_1)$ 
4:     for each width  $ws_2 = 1 \dots n, \infty$  do
5:        $oG_{2A} = \text{group}_A(ws_2, p, S_2)$ 
6:        $fea_a = \text{com}U_1(oG_{1A}, oG_{2A})$ 
7:       accumulate  $fea_a$  for final layer
8:     end for
9:   end for
10:  for each width  $ws_1 = 1 \dots n$  do
11:     $oG_{1B} = \text{group}_B(ws_1, p, S_1)$ 
12:     $oG_{2B} = \text{group}_B(ws_1, p, S_2)$ 
13:    for each  $i = 1 \dots \text{numFilter}_B$  do
14:       $fea_b = \text{com}U_1(oG_{1B}[*][i], oG_{2B}[*][i])$ 
15:      accumulate  $fea_b$  for final layer
16:    end for
17:  end for
18: end for
```

2-9行。红色框也没有花全，应该对于同一种池化（max）每个句子的 ws_1 ， ws_2 都互相比对，而不是像图中 ws_1 仅和 ws_1 比较。



模型介绍——相似性计算层





模型介绍——全连接层

相似性计算层算法计算得到的相似度向量在接上一个Dense层，最后接Output层，就得到了完整的模型结构.



实验结果及分析



实验结果及分析

Model	Acc.	F1
Hu et al. (2014) ARC-I	69.6%	80.3%
Hu et al. (2014) ARC-II	69.9%	80.9%
Blacoe and Lapata (2012)	73.0%	82.3%
Fern and Stevenson (2008)	74.1%	82.4%
Finch (2005)	75.0%	82.7%
Das and Smith (2009)	76.1%	82.7%
Wan et al. (2006)	75.6%	83.0%
Socher et al. (2011)	76.8%	83.6%
Madnani et al. (2012)	77.4%	84.1%
Ji and Eisenstein (2013)	80.41%	85.96%
Yin and Schütze (2015) (without pretraining)	72.5%	81.4%
Yin and Schütze (2015) (with pretraining)	78.1%	84.4%
Yin and Schütze (2015) (pretraining+sparse features)	78.4%	84.6%
This work	78.60%	84.73%

Model	r	ρ	MSE
Socher et al. (2014) DT-RNN	0.7863	0.7305	0.3983
Socher et al. (2014) SDT-RNN	0.7886	0.7280	0.3859
Lai and Hockenmaier (2014)	0.7993	0.7538	0.3692
Jimenez et al. (2014)	0.8070	0.7489	0.3550
Bjerva et al. (2014)	0.8268	0.7721	0.3224
Zhao et al. (2014)	0.8414	-	-
LSTM	0.8477	0.7921	0.2949
Bi-LSTM	0.8522	0.7952	0.2850
2-layer LSTM	0.8411	0.7849	0.2980
2-layer Bidirectional LSTM	0.8488	0.7926	0.2893
Tai et al. (2015) Const. LSTM	0.8491	0.7873	0.2852
Tai et al. (2015) Dep. LSTM	0.8676	0.8083	0.2532
This work	0.8686	0.8047	0.2606

关于SICK数据的测试集结果

MSRP上用于语义识别的测试集
结果灰色行是基于神经网络的方法



实验结果及分析

Model	Pearson's r
Rios et al. (2012)	0.7060
Wang and Cer (2012)	0.8037
Beltagy et al. (2014)	0.8300
Bär et al. (2012)	0.8730
Šarić et al. (2012)	0.8803
This work	0.9090

关于MSRVID数据的测试集结果

敬请各位大佬批评指正
