

# Deep Learning on Improved Word Embedding Model for Topic Classification

Yingying Zhou, Lei Fan

School of Information Security Engineering, Shanghai Jiao Tong University, Shanghai  
Email: celestialsir@sjtu.edu.cn, fanlei@sjtu.edu.cn

Received: Oct. 18<sup>th</sup>, 2016; accepted: Nov. 6<sup>th</sup>, 2016; published: Nov. 9<sup>th</sup>, 2016

Copyright © 2016 by authors and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

---

## Abstract

Topic classification has wide applications in content searching and information filtering. It can be divided into two core parts: text embedding and classification modeling. In recent years, methods have brought out significant results using distributed word embedding as input and convolutional neural network (CNN) as classifiers. This paper discusses the impact of different word embedding for CNN classifiers, proposes topic2vec, a new word embedding specifically suitable for Chinese corpora, and conducts an experiment on Zhihu, a representative content-oriented internet community. The experiment turns out that CNN with topic2vec gains an accuracy of 98.06% for long content texts, 93.27% for short title texts and an improvement comparing with other word embedding models.

## Keywords

Topic Classification, Deep Learning, Convolutional Neural Network, Word Embedding

---

# 基于改进词向量模型的深度学习文本主题分类

周盈盈, 范磊

上海交通大学信息安全工程学院, 上海  
Email: celestialsir@sjtu.edu.cn, fanlei@sjtu.edu.cn

收稿日期: 2016年10月18日; 录用日期: 2016年11月6日; 发布日期: 2016年11月9日

## 摘要

主题分类在内容检索和信息筛选中应用广泛,其核心问题可分为两部分:文本表示和分类模型。近年来,基于分布式词向量对文本进行表示,使用卷积神经网络作为分类器的文本主题分类方法取得了较好的分类效果。本文研究了不同词向量对卷积神经网络分类效果的影响,提出针对中文语料的topic2vec词向量模型。本文利用该模型,对具有代表性的互联网内容生成社区“知乎”进行了实验与分析。实验结果表明,利用topic2vec词向量的卷积神经网络,在长内容文本和短标题文本的分类问题中分别取得了98.06%,93.27%的准确率,较已知词向量模型均有显著提高。

## 关键词

主题分类, 深度学习, 卷积神经网络, 词向量

## 1. 引言

文本主题分类将文本按照潜在的主题进行分类,在内容检索和信息筛选中应用广泛,涉及语言学、自然语言处理、机器学习等领域。文本主题分类的核心问题包括文本表示、分类模型等。

传统的文本表示方法主要基于词袋模型,该方法认为文档是无序词项的集合。词袋模型存在数据稀疏问题,并且丢弃了词序、语法等文档结构信息,对分类效果的提升存在瓶颈。传统的文本分类模型以词袋为输入特征,并基于一定的模型假设。其基本思想是,假设数据分类服从某种概率分布,利用贝叶斯理论获得最佳的分类器。由于对于数据附加了额外的假设,因此,一旦假设不成立则会严重影响模型准确率。

近年来,对文本分类问题的改进研究主要集中于深度学习。针对深度学习的分类模型,文本表示可使用 Mikolov 等人提出的分布式词向量[1]方法。该方法针对词袋模型中语义信息丢失的问题,利用语言神经网络将字典空间映射为词向量空间。分析表明,对于空间距离相近的词项,其语义或词性也具有强关联性。在分类模型方面,以分布式词向量作为模型的输入特征,循环神经网络、递归神经网络、卷积神经网络等深度学习模型都可应用于文本分类。其中, Kim 等人于 2014 年提出的卷积神经网络(CNN) [2] [3] [4] [5]在文本主题分类中具有较高的准确率,在 MR 数据集[6]中的分类准确率达 81.5%。

本文主要研究基于卷积神经网络的文本分类方法,并通过改进词向量空间模型提高分类的准确性。针对中文语料的特点,提出了改进的词向量模型 topic2vec,并通过对比实验,研究不同词向量作为卷积神经网络的输入特征,对分类准确率的影响,其中,利用 topic2vec 的卷积神经网络在长短语料中分别取得 98.06%, 93.27%的准确率,较其他词向量如 word2vec [1]、GloVe [7]和随机词向量都有提高。实验表明,对于中文语料而言,利用卷积神经网络进行文本主题分类是有效的,且将 topic2vec 作为原始特征的模型分类效果优于其他词向量。

本文的结构如下:第二章介绍文本分类的相关工作。第三章介绍 topic2vec 模型的原理及文本分类模型的结构。第四章介绍实验设定和实验结果,并对实验结果进行讨论。第五章为总结。

## 2. 相关工作

深度学习是文本分类的主流方法,取得比较好的效果。其核心是卷积神经网络作为分类器,同时使用词向量模型作为文本表示。

## 2.1. 卷积神经网络

基于卷积神经网络的文本分类模型是一个有监督的三层网络模型, 由输入层、隐层和输出层构成, 通过后向传播算法[8]进行参数优化。输入层为一组大小相等的文档矩阵, 矩阵第  $n$  行对应文档第  $n$  个词项的向量表示; 隐层通过一组不同大小的卷积核(filter)对文本进行自上而下的卷积运算, 提取一组特征向量, 并从每个特征向量中选取最大项(max-pooling [9]), 组成特征地图; 输出层通过 Softmax 分类器[10]计算文档落在各个主题下的概率并输出最终分类结果。模型结构如图 1。

模型的训练使用极大似然估计, 即求出一组参数  $\theta$ , 使得目标函数最大化

$$\theta \mapsto \sum_{(x,y) \in D_T} \log(p(y|x,\theta))$$

$$\log(p(y|x,\theta)) = y - \log\left(\sum_{j=1}^K \exp(y_j)\right)$$

其中  $D_T$  为训练集,  $K$  为主题数,  $(x, y)$  为训练文档及其对应的正确主题分类标号。模型采用随机梯度下降法(SGD)优化上述目标。令学习率为  $\alpha$ , 在每一轮迭代中, 随机选取一组样本  $(\hat{x}, \hat{y})$ , 并用下式进行一轮梯度迭代

$$\theta \leftarrow \theta + \alpha \frac{\partial \log(p(\hat{y}|\hat{x},\theta))}{\partial \theta}$$

Zhang 等人[3]分别利用 word2vec、GloVe、word2vec+GloVe 作为 CNN 模型的输入特征, 并在不同英文数据集下进行实验。实验结果表明, 对于不同数据集, word2vec 和 GloVe 的分类效果各不相同, 因此, 词向量的选择直接影响模型的性能。

## 2.2. 分布式词向量

词向量是深度学习模型的输入特征, 根据向量之间的相关性, 可分为随机词向量和分布式词向量。随机词向量将词项随机表示为两两正交的单位向量, 由 1 个 1 和  $W-1$  个 0 组成, 其中  $W$  为字典长度, 其构造方式简单, 但词项之间的关联关系则被完全割裂, 并可能造成维数灾难。分布式词向量则将词项表示为一个几十到几百维的稠密型实值向量, 并可通过向量空间距离考察词项之间的语义相似度, 其中, 最具代表性的是 word2vec。对于长度为  $N$  的文档, word2vec 考察每个词项  $w_i$  的  $c$  个上下文, 其目标函数为极大对数似然

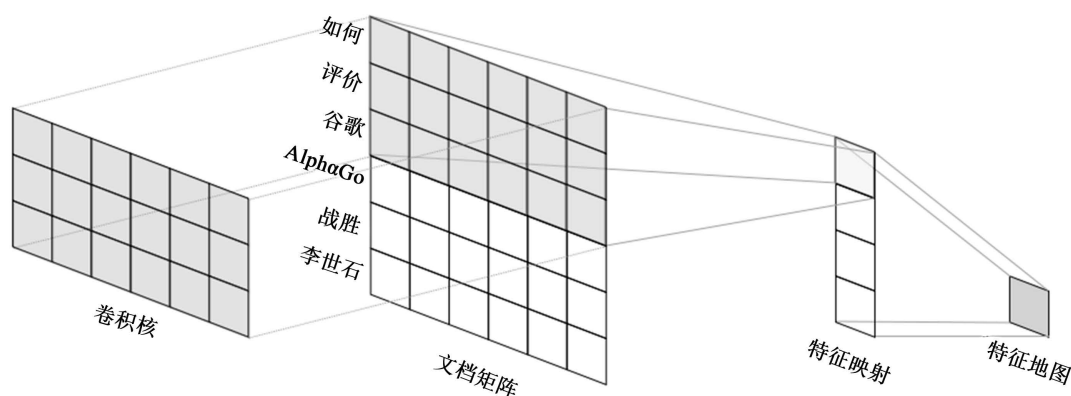


Figure 1. Text classification model based on convolutional neural network  
图 1. 基于卷积神经网络的文本分类模型

$$\frac{1}{N} \sum_{t=1}^N \sum_{-c \leq j \leq c, j \neq 0} \log(p(w_{t+j} | w_t))$$

通过随机梯度下降法求解目标函数可得词向量。其中, 词项上下文的条件概率  $p(w_{t+j} | w_t)$  由 Softmax 函数给出

$$p(w_o | w_t) = \frac{\exp(v_{w_o}^T v_{w_t})}{\sum_{w=1}^W \exp(v_w^T v_{w_t})}$$

由于 word2vec 仅考察目标词  $w_t$  的局部上下文, 因此损失了语料中全文性的关联信息, 主题模型可以体现语料的全文关联性信息。

### 2.3. 主题模型

主题模型[11]是无监督的文本聚类模型。模型由文档、词项、主题三个维度构成, 通过模拟文档的生成过程, 对文档在各主题下的概率分布矩阵  $\theta$  和各个词语在各主题下的概率分布矩阵  $\beta$  进行参数估计。假设  $\theta$  服从先验分布  $p(\theta | \alpha)$ , 接着随机为文档分配一个主题  $z \sim p(z | \theta)$ , 并随机生成一个单词  $w \sim p(w | z, \beta)$ 。假设语料库  $D$  大小为  $M$ , 第  $d$  篇文档长度为  $N_d$ , 则语料库的目标函数为

$$p(D | \alpha, \beta) = \prod_{d=1}^M \int p(\theta_d | \alpha) \left( \prod_{n=1}^{N_d} \sum_{z_{d_n}} p(z_{d_n} | \theta_d) p(w_{d_n} | z_{d_n}, \beta) \right) d\theta_d$$

将主题数设置为词向量的维度, 得词项的稀疏主题分布向量, 并可作为字典在向量空间的映射, 通过向量空间距离考察词项的在全局语料范围内的主题相似度。

## 3. 基于深度学习的文本分类模型

现有的分布式词向量, 都直接沿用了比较上下文对单词建模的方法, 缺少了在全局文档中对主题特征的刻画, 针对这一局限性, 本章提出 topic2vec 词向量模型, 并将其作为卷积神经网络的输入特征。

### 3.1. topic2vec 词向量表示

word2vec 等分布式词向量使用固定大小的滑动窗口对文档中每个词项的上下文进行统计, 并给出词在向量空间中的映射。此类模型仅从局部文档考察词项的语义, 并不统计词在全局文档中的主题信息。一般而言, 词向量中保留越多的语义信息, 深度学习模型对语料的特征学习效果更好。

topic2vec 词向量模型主要基于这个猜想, 设分词后的中文语料库为  $D$ , 取语料库中所有独立词项  $w_i$ , 建立字典  $W = \{w_1, w_2, \dots, w_n\}$ , 其中  $n$  为不同词项的总数。为了在词向量中加入适量的主题信息, 引入主题分布向量  $\beta$  对词项的主题属性进行预训练。

**算法 1** 通过伪代码描述了主题分布向量的生成过程。假设分词的主题分布均匀, 使用 Dirichlet 分布[12]作为词项的先验主题分布, 即  $\beta \sim \text{Dirichlet}(\delta)$ 。遍历语料库  $D$  中每个文档  $d_i$ , 以及  $d_i$  中的每个词项  $w_j$ , 第一步, 考察当前文档  $d_i$  中除去当前词  $w_j$  后的其他所有词项, 根据它们的主题分布进行投票表决, 生成文档  $d_i$  在词项  $w_j$  下的主题  $z_{i,j}$ ; 第二步, 根据当前主题  $z_{i,j}$ , 更新文档  $d_i$  的主题分布  $\theta$  以及当前词项  $w_j$  的主题分布  $\beta$ 。Blei 等人[11]证明, 重复以上遍历, 当循环次数  $I$  足够大时, 模型收敛。最后对两个统计矩阵进行归一化, 得到词项的主题分布概率矩阵  $\beta = \{b_1, b_2, \dots, b_N\}$ 。

**Algorithm 1.** Topic model**算法 1.** 主题模型

Input: $D, I$
Generalize topic distribution of terms $\beta \sim \text{Dirichlet}(\delta)$
for $i = 1$ to $I$ :
for each $d_i$ in $D$ :
for each $w_j$ in doc:
Choose a topic for $d_i$ : $z_{i,j} \sim p(z   d_i - w_j, \beta)$
Update topic distribution of documents $\theta$
Update topic distribution of terms $\beta$
Output $E(\theta), E(\beta)$

由模型可知, 对任意词项  $w_k$ , 对应主题分布向量  $b_k = \{b_k^1, b_k^2, \dots, b_k^d\}$ , 其中  $b_k^i \in [0, 1]$ , 且  $\sum_i b_k^i = 1$ , 因此, 当  $N$  足够大且主题分布均匀时,  $|b_k| = \sqrt{\sum_{i=1}^d |b_k^i|^2} \rightarrow 0$ , 词项的主题分布矩阵  $\beta$  是稀疏的。

**算法 2** 通过伪代码描述了词向量 `topic2vec` 生成的具体过程。第一步, 对词项的邻近上下文进行考察, 基于滑动窗口内的语义信息构造维数为  $d$  的 `word2vec` 词向量  $V = \{v_1, v_2, \dots, v_N\}$ ; 其中  $N$  为字典空间的维度  $\dim(W)$ ; 第二步, 通过主题模型获取词项在全局文档范围内的  $d$  维稀疏主题分布向量  $\beta = \{b_1, b_2, \dots, b_N\}$ ; 第三步, 将两组词向量进行线性加权求和, 即  $t_i \leftarrow v_i + \mathcal{G}b_i$ , 其中  $b_i$  的共享权重  $\mathcal{G}$  为词向量空间  $V$  的平均 L2 范数, 通过对向量空间中的词项进行微小位移, 同一主题下不同词性的词项经过调整后距离更近, 并生成新的词向量空间  $T = \{t_1, t_2, \dots, t_N\}$ 。

由上述算法可知,  $\mathcal{G}$  是词向量在主题维度的偏移程度。对任意词项  $w_k$  和  $w_l$ , 用余弦距离表征它们的语义相似度

$$\begin{aligned} \cos(t_k, t_l) &= \frac{\langle t_k, t_l \rangle}{|t_k| |t_l|} = \frac{\langle v_k + \mathcal{G}b_k, v_l + \mathcal{G}b_l \rangle}{|v_k + \mathcal{G}b_k| |v_l + \mathcal{G}b_l|} \\ &= \frac{|v_k| |v_l| \cos(v_k, v_l) + \mathcal{G} \langle v_k, b_l \rangle + \mathcal{G} \langle v_l, b_k \rangle + \mathcal{G}^2 \langle b_k, b_l \rangle}{\sqrt{(|v_k|^2 + 2\mathcal{G} \langle v_k, b_k \rangle + |b_k|^2)} \sqrt{(|v_l|^2 + 2\mathcal{G} \langle v_l, b_l \rangle + |b_l|^2)}} \\ &= \alpha \cos(v_k, v_l) + \mathcal{G}\varepsilon \end{aligned}$$

其中  $\alpha = \frac{|v_k| |v_l|}{\sqrt{(|v_k|^2 + 2\mathcal{G}x_{kk})} \sqrt{(|v_l|^2 + 2\mathcal{G}x_{ll})}}$ ,  $\varepsilon = \frac{x_{kl} + x_{lk}}{\sqrt{(|v_k|^2 + 2\mathcal{G}x_{kk})} \sqrt{(|v_l|^2 + 2\mathcal{G}x_{ll})}}$ ,  $x_{ij} = \langle v_i, b_j \rangle$ 。为了保留词向量

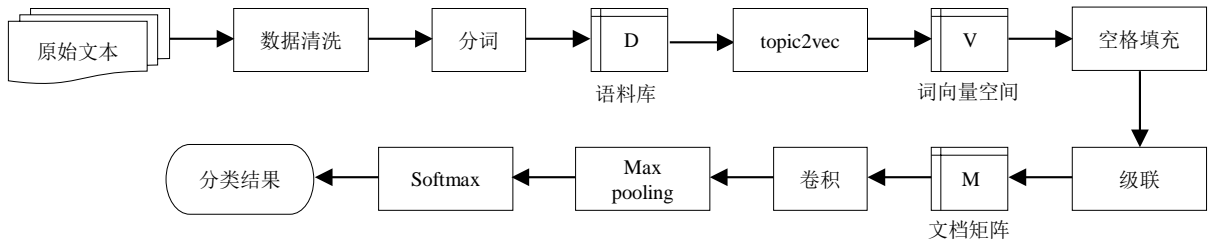
在局部上下文中的语法信息, 同时引入全局文档范围内语义信息, 取  $\mathcal{G} = \frac{1}{N} \sum_{i=1}^N |v_i|$ , 即词向量空间  $V$  的

平均 L2 范数。此时,  $\alpha, \varepsilon \in (0, 1)$ ,  $E(\alpha) = \frac{\mathcal{G}^2}{\mathcal{G}^2 + 2\mathcal{G}x}$ ,  $E(\varepsilon) = \frac{2x}{\mathcal{G}^2 + 2\mathcal{G}x}$ ,  $x = E(x_{ij})$ 。

**3.2. 文本分类模型**

利用 `topic2vec` 作为卷积神经网络的输入特征, 得到改进的文本分类模型, 模型结构如图 2。

模型首先对文本进行数据清洗、分词、构建字典并完成词项到 `topic2vec` 的映射, 接着将原始文本表示为对应词向量的级联, 并在文末进行空格填充, 以保证每个文档的输入矩阵高度相等



**Figure 2.** Text classification model based on topic2vec  
**图 2.** 基于 topic2vec 的文本分类模型

**Algorithm 2.** topic2vec  
**算法 2.** topic2vec

---

Input:  $D$ , dimension of word vector  $d$

---

```

 $W \leftarrow \text{BuildDictionary}(D), N \leftarrow \text{dim}(W)$ 
 $\theta, \beta \leftarrow \text{TopicModel}(D, d)$ 
 $V \leftarrow \text{skipgram}(D, d)$ 
 $\mathcal{G} \leftarrow E(|v_i|)$ 
for each  $w_i$  in  $W$ :
    find correlated word vector  $v_i \in V, b_i \in \beta$ 
     $t_i \leftarrow v_i + \mathcal{G}b_i$ 

```

Output  $T = \{t_1, t_2, \dots, t_N\}$

---

$$M = t_1 \oplus t_2 \oplus \dots \oplus t_N$$

对于高度为  $N$ , 维度为  $d$  的文档矩阵  $M$ , 卷积层用宽度恒定为  $d$ , 高度  $h$  可变 ( $0 \leq h \leq s$ ) 的滑动窗口  $W$  对文档矩阵  $M$  进行自上而下的扫描,  $W$  每次可扫过的  $h$  个相邻的词向量, 记为  $M[i:i+h-1] \in \mathbb{R}^{hd}$ , 通过自上而下对窗口进行滑动,  $i$  可取  $1, 2, \dots, s-h+1$ . 定义卷积核  $w \in \mathbb{R}^{hd}$  为一个长度为  $hd$  的权值向量. 将  $w$  与  $W$  中的局部文档矩阵进行卷积, 得到一个特征响应

$$c_i = \text{ReLU}(w * M[i:i+h-1] + b)$$

其中  $b \in \mathbb{R}$  为共享偏置,  $\text{ReLU}$  为修正线性单元. 对文档完成一周窗口滑动可得一组特征映射向量  $C = \{c_1, c_2, \dots, c_{s-h+1}\}$ . 通过改变滑动窗口的高度  $h$  和卷积核  $w$ , 同一文档可以通过卷积层生成一组特征映射向量  $\{C_1, C_2, \dots, C_k\}$ .

由于滑动窗口的高度不尽相同, 卷积得到的特征映射向量也有各自不同的维度, 因此, 为了对特征映射向量进行正则化, 模型通过表决层(max pooling)从每个特征映射向量中挑选出最大项  $\hat{c}_i = \max(C_i)$ , 并将其余项丢弃, 生成最终的特征地图  $F = \{\hat{c}_1, \hat{c}_2, \dots, \hat{c}_k\}$ . 此时, 特征地图的维度  $k$  由滑动窗口个数和每个窗口对应的卷积核个数共同决定, 皆为用户指定的模型超参数.

模型输出层的节点个数  $K$  即为主题个数, 通过 Softmax 分类器[6]对每个文档的特征地图进行归一化, 并给出文档落在第  $i$  个主题下的概率

$$p(i | x, \theta) = \frac{\exp(y_i)}{\sum_{j=1}^K \exp(y_j)}$$

其中  $x$  为输入文档,  $\theta$  为模型参数, 包括词向量和网络中的参数。

## 4. 实验分析

本文实验主要以具有代表性的中文网络文本最为研究对象。知乎作为国内知名的社会化问答社区, 截至 2014 年 8 月有活跃用户数有 350 万, 具有较为完备的标签与分类机制, 是理想的文本主题分类语料来源。

### 4.1. 实验设置

实验数据集采集自知乎, 包括 8 个主题 45063 个精华提问及高票回答, 话题涉及计算机、体育、经济、美食、教育、医疗、人际传播和艺术。以分词为文本长度的最小单位, 统计各个文本的长度分布, 如图 3 所示。

在数据预处理中, 本文采用 Stanford Core NLP 3.5.0 [13] 进行分词, 并过滤掉文本中出现的超链接、标点符号、日期及停用词等无效信息。经过预处理后的实验语料平均长度为 359, 字典长度为 116432, 取词向量维度为 300。表 1 给出了分词后的实例。

实验使用两个语料, “短语料”为数据集中所有标题信息, “长语料”为标题、问题详情、和高票回答的组合信息。“短语料”提供了话题的精简信息, 词项少但涵盖关键词的比例较高, “长语料”提供了话题的全部信息, 内容丰富但干扰项多。图 4 给出了两个语料的长度分布。

使用完整长语料作为词向量的训练集, 并分别在两组语料上进行主题分类实验, 考察分类模型在短文本和长文本中的分类效果。实验在配备 Intel Core i7 处理器, 内存 8GB 的 OS X El Capitan 操作系统下进行, 卷积神经网络的训练使用 Python 的深度学习框架 theano [14]。

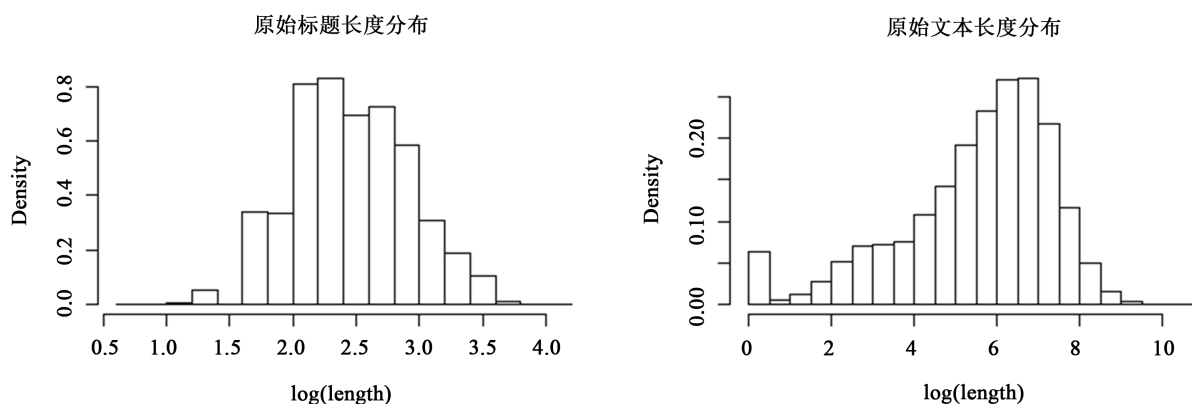


Figure 3. Length distribution of Zhihu raw dataset

图 3. 知乎原始数据集文本长度分布

Table 1. Example of data preprocessing

表 1. 数据预处理示例

原始文本	预处理后
图灵机器人( <a href="http://www.tuling123.com/">http://www.tuling123.com/</a> )这款智能自动问答应用在业界处于什么水平? 应用前景如何	图灵机器人智能自动问答应用业界处于什么水平应用前景如何
假如你带领一个开发、一个美工、一个产品、一个人资去开一家老北京炸酱面, 你会如何各尽其能地让生意越来越红火?	假如带领开发美工产品资去北京炸酱面如何各尽其能生意越来越红火
2014 年 10 月 19 日, 在 PM2.5 值超过 400 的情况下参加北京马拉松是什么体验?	PM2.5 超过情况参加北京马拉松什么体验

## 4.2. 实验结果

对词向量的评价主要基于两个方面的考量：词向量的语言学特征和词向量作为分类模型的初始特征对模型效率的影响。

在评价词向量的语言学特征方面，本章选取了 5 个词项，考察它们在向量空间中的最邻近词项及语义相关性。表 2 比较了 word2vec 和 topic2vec 得到的最邻近词项：

由表 2 可知，利用 word2vec 训练得到的邻近词项具有词性相同、实体种类相近的特点，例如“苹果”与“香蕉”等；而 topic2vec 则能够找到同一主题下的强关联词，例如“乔布斯”与“苹果”，topic2vec 可以更好地体现人类对词语的理解。

在将词向量作为特征提高模型性能方面，本章对长语料和短语料分别进行分类实验，并以不同词向量作为输入特征进行对比，这些词向量包括：随机词向量(rand)、word2vec、GloVe 和 topic2vec。将实验数据随机分成 10 组进行交叉验证。实验结果如下：

由表 3 可知，较于主题模型算法，CNN 具有准确度高、鲁棒性好等优势。此外，影响 CNN 分类结果的主要因素包括文本长度和词向量的选择。对于长文本，topic2vec 的平均准确率为 0.98，word2vec 的平均准确率为 0.96，随机正交词向量的准确率为 0.90；对于短语料，topic2vec 和 word2vec 的平均准确率较为接近，为 93.27% 和 91.54%，GloVe 的准确率为 69.72%。正交词向量则为 67.17%。对于长语料，topic2vec 的平均准确率为 98.06%，word2vec、GloVe 和随机正交词向量则分别为 95.71%、92.4%、89.95%。由此可见，通过训练词向量，将词语之间的词性和语义关系表达为词向量之间的空间距离，这在 CNN 的分类过程中具有一定的积极影响。

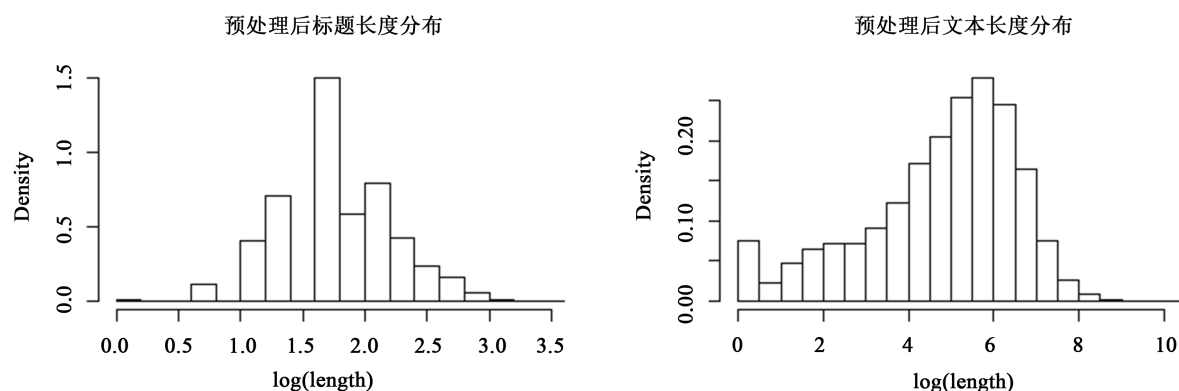


Figure 4. Length distribution of Zhihu preprocessed dataset

图 4. 知乎预处理后数据集文本长度分布

Table 2. Nearest terms of 2 types of word embedding

表 2. 两种词向量得到的最邻近词项

词项	Word2vec	Topic2vec
苹果	Apple, 黑莓, 香蕉, 橘子, 橙子	Apple, 黑莓, 乔布斯, 小米, 库克
锅包肉	锅包, 咸口, 包肉, 茄盒, 米豆腐	正宗, 好吃, 地道, 吃到, 小吃
足球	篮球, 排球, 五人制, 橄榄球, 曲棍球	篮球, 打球, 运动员, 踢球, 踢足球
大学	入读, 早稻田, 宾夕法尼亚, 学校, CCER	入读, 早稻田, 学校, 就读, 匹兹堡
禽流感	H7N9, H1N1, 流感, 高致病性, H10N8	流感, 高致病性, 甲型, 病毒, 感染



**Table 3.** Classification accuracy of CNN under different word embeddings**表 3.** 不同词向量下的 CNN 文本分类准确率

词向量	短语料(%)	长语料(%)
Rand	67.17	89.95
GloVe	69.72	92.40
word2vec	91.54	95.71
topic2vec	93.27	98.06

## 5. 总结

本文研究了基于深度学习的文本分类, 以及不同词向量输入对神经网络性能的影响, 提出了一种改进的词向量表示方法 Top2vec。通过实验数据表明, 该方法对长短语料分类准确率均有提高。

## 基金项目

上海市科委基础研究重点项目(13JC1403501)。

## 参考文献 (References)

- [1] Mikolov, T., Sutskever, I., Chen, K., Corrado, G., *et al.* (2013) Distributed Representations of Words and Phrases and Their Compositionality. *Advances in Neural Information Processing Systems*, 3111-3119.
- [2] Kim, Y. (2014) Convolutional Neural Networks for Sentence Classification.
- [3] Kalchbrenner, N., Grefenstette, E. and Blunsom, P. (2014) A Convolutional Neural Network for Modelling Sentences. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, 655-665.
- [4] Johnson, R. and Zhang, T. (2015) Semi-Supervised Convolutional Neural Networks for Text Categorization via Region Embedding. *Advances in Neural Information Processing Systems*, 919-927.
- [5] Zhang, Y. and Wallace, B. (2015) A Sensitivity Analysis of (and Practitioners' Guide to) Convolutional Neural Networks for Sentence Classification.
- [6] Pang, B. and Lee, L. (2005) Seeing Stars: Exploiting Class Relationships for Sentiment Categorization with Respect to Rating Scales. *Proceedings of the ACL*. <http://dx.doi.org/10.3115/1219840.1219855>
- [7] Pennington, J., Socher, R. and Manning, C.D. (2014) Glove: Global Vectors for Word Representation. *Proceedings of the Empirical Methods in Natural Language Processing*. <http://dx.doi.org/10.3115/v1/d14-1162>
- [8] Rumelhart, D.E., Hinton, G.E. and Williams, R. (1988) Learning Representations by Back-Propagating Errors. *Cognitive Modeling*, 5, 3.
- [9] Boureau, Y.L., Ponce, J. and Le Cun, Y. (2010) A Theoretical Analysis of Feature Pooling in Visual Recognition. *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, 111-118.
- [10] Hinton, G.E. and Salakhutdinov, R.R. (2009) Replicated Softmax: An Undirected Topic Model. *Advances in Neural Information Processing Systems*, 1607-1614.
- [11] Hoffman, M., Bach, F.R. and Blei, D.M. (2010) Online Learning for Latent Dirichlet Allocation. *Advances in Neural Information Processing Systems*, 856-864.
- [12] Blei, D.M., Ng, A.Y. and Jordan, M.I. (2003) Latent Dirichlet Allocation. *Journal of Machine Learning Research*, 3, 993-1022.
- [13] Manning, C.D., Surdeanu, M., Bauer, J., *et al.* (2014) The Stanford CoreNLP Natural Language Processing Toolkit. *ACL (System Demonstrations)*, 55-60.
- [14] Bergstra, J., Breuleux, O., Bastien, F., *et al.* (2010) Theano: A CPU and GPU Math Compiler in Python. *Proceedings of the 9th Python in Science Conference*, 1-7.

**期刊投稿者将享受如下服务：**

1. 投稿前咨询服务 (QQ、微信、邮箱皆可)
2. 为您匹配最合适的期刊
3. 24 小时以内解答您的所有疑问
4. 友好的在线投稿界面
5. 专业的同行评审
6. 知网检索
7. 全网络覆盖式推广您的研究

投稿请点击：<http://www.hanspub.org/Submission.aspx>

期刊邮箱：[csa@hanspub.org](mailto:csa@hanspub.org)