

GM-FastText 多通道词向量短文本分类模型^①



白子诚, 周艳玲, 张 龔

(湖北大学 计算机与信息工程学院, 武汉 430062)
通信作者: 周艳玲, E-mail: sunnyzhou@hubu.edu.cn

摘 要: 针对短文本分类中文本特征稀疏难以提取、用词不规范导致 OOV (out of vocabulary) 等问题, 提出了基于 FastText 模型多通道嵌入词向量, 和 GRU (gate recurrent unit) 与多层感知机 (multi-layer perceptron, MLP) 混合网络结构 (GRU-MLP hybrid network architecture, GM) 的短文本分类模型 GM-FastText. 该模型使用 FastText 模型以 N-gram 方式分别产生不同的嵌入词向量送入 GRU 层和 MLP 层获取短文本特征, 通过 GRU 对文本的特征提取和 MLP 层混合提取不同通道的文本特征, 最后映射到各个分类中. 多组对比实验结果表明: 与 TextCNN、TextRNN 方法对比, GM-FastText 模型 *F1* 指标提升 0.021 和 0.023, 准确率提升 1.96 和 2.08 个百分点; 与 FastText、FastText-CNN, FastText-RNN 等对比, GM-FastText 模型 *F1* 指标提升 0.006、0.014 和 0.016, 准确率提升 0.42、1.06 和 1.41 个百分点. 通过对比发现, 在 FastText 多通道词向量和 GM 混合结构网络的作用下, 多通道词向量在短文本分类中有更好的词向量表达且 GM 网络结构对多参数特征提取有更好的性能.

关键词: 短文本分类; 快速文本; 词向量; 多层感知机; 多特征

引用格式: 白子诚,周艳玲,张龔.GM-FastText 多通道词向量短文本分类模型.计算机系统应用. <http://www.c-s-a.org.cn/1003-3254/8648.html>

Short Text Classification Model of GM-FastText Multi-channel Word Vector

BAI Zi-Cheng, ZHOU Yan-Ling, ZHANG Yan

(School of Computer Science and Information Engineering, Hubei University, Wuhan 430062, China)

Abstract: To tackle the problems in short text classification, such as difficult extraction of sparse text features and out of vocabulary (OOV) caused by non-standard words, this study proposes a short text classification model GM-FastText based on the FastText multi-channel embedded word vector and the GRU-MLP hybrid network architecture (GM) built by a gated recurrent unit (GRU) and multi-layer perceptron (MLP). This model uses the FastText model to generate different embedded word vectors in the N-gram mode and feeds them into the GRU layer and MLP layer to obtain short text features. After the extraction of text features by GRU and the hybrid extraction of the text features in different channels in the MLP layer, they are finally mapped to each classification. The experimental results show that compared with TextCNN and TextRNN, the GM-FastText model has an *F1* index increased by 0.021 and 0.023 and accuracy by 1.96 and 2.08 percentage points. Moreover, compared with FastText, FastText-CNN and FastText-RNN, the GM-FastText has an *F1* index improved by 0.006, 0.014 and 0.016 and accuracy by 0.42, 1.06 and 1.41 percentage points. In short, under the action of FastText multi-channel word vector and GM hybrid structure network, the multi-channel word vector has better word vector expression in short text classification and the GM network structure has better performance for multi-parameter feature extraction.

Key words: short text classification; FastText; word vector; multi-layer perceptron (MLP); multi-feature

^① 基金项目: 国家自然科学基金面上项目 (61977021)

收稿时间: 2021-11-23; 修改时间: 2021-12-20; 采用时间: 2021-12-28; csa 在线出版时间: 2022-05-30

1 引言

随着互联网的普及,中国大数据产业领跑全球,每时每刻都产生大量的短文本数据信息,如新闻标题、应用评论、短信息等^[1].在大数据时代背景下,大量短文本信息的筛选与管理成为人们亟待解决的需求.文本分类作为自然语言处理(natural language processing, NLP)的一个子任务,是将指定文本归纳到预定义标签的过程,广泛应用于新闻标题分类、情感分析、主题标签和对话系统等,对于特定的信息筛选,有着极大的便利.与长文本分类相比,短文本分类数据具有特征稀疏,用词不规范,数据海量等问题^[2].

目前处理短文本分类思路主要有两个方向^[3]:一是通过大规模的预训练语言模型,生成“动态”的嵌入词向量,通过引入大量的外部信息提高分类效果;二是通过构建优良的模型结构,更深层次的挖掘信息从而提高分类准确度.本文同时考虑这两个角度改进提高文本分类效果.为了获取更多短文本的特征,采用FastText方法^[4]代替传统Word2Vec方法产生嵌入词向量,这种方式不单单只是利用字向量级别的嵌入词向量,同时产生含有N-gram级别的嵌入词向量,形成多通道的嵌入词向量输入;另外在多通道的嵌入词输入下,采取了GRU(gate recurrent unit)和多层感知机(multi-layer perceptron, MLP)混合网络结构(GRU-MLP hybrid network architecture, GM)提取并结合各通道词向量特征.提出新的文本分类结构GM-FastText,并在多个数据集上通过对比分析其性能.

2 相关工作

在数据较小情况下,可以采用传统的机器学习方式,如朴素贝叶斯、K-近邻、支持向量机等,这些方法通过对数据集的预定义学习从而预测结果.然而,特征工程的建立往往需要大量工作量.随着数据量的增长,传统机器学习方式已不再适用,卷积神经网络(convolutional neural networks, CNN)、循环神经网络(recurrent neural networks, RNN)等基于神经网络可以自主提取文本的特征,减少大量人工标注, Kim^[5]提出TextCNN模型,应用了多个不同卷积核提取文本特征实现文本分类.

实现自主提取特征的关键是字词向量化.在NLP中广泛应用的独热编码(one-hot)将词表示为长度为词表大小的长维度向量,其中仅一个纬度值为1及表示

这个词.这样的编码方式易于理解,但是难以联系上下文关系,在实际应用过程中易造成维数灾难等问题.然而,分布式词向量表示则在很大程度上解决了以上问题,训练良好的分布式词向量可以通过计算向量之间欧几里得距离衡量相似度.现阶段主流的分布式词向量训练模型有Mikolov^[6]在2013年提出的Word2Vec模型,其训练方式有两种模式:CBOW和Skip-gram,并通过层次Softmax和负采样两种优化方式训练,可以得到较为准确的词向量表示. Facebook研究团队提出的FastText模型是一个开源词向量计算和文本分类工具,其效果可以媲美深度神经网络,运行速度快并且在CPU上一分钟能实现10万数据级别的分类任务,同样是在Word2Vec的基础上进行了改进,可以得到除了词向量以外的N-gram向量信息.张焱博等人^[7]将预训练词向量分别通过CNN-Bi-LSTM和FastText提取浅层语义并做拼接然后直接映射到分类,得到了较好的结果并加速了训练过程.汪家成等人^[8]为解决FastText准确率低问题,在输入阶段使用TextRank和TF-IDF技术使输入特征表示信息量更高.范昊等人^[9]则利用Bi-GRU直接处理FastText词向量.但是这些操作仅是对FastText不同N-gram词向量简单的叠加操作,而丢失了各个N-gram词向量独立特征.不同N-gram词向量这一特性对于短文本分类处理可以一定程度上解决“简写”导致超出词表问题.然而FastText产生的向量组比主流的词向量多了很多数据参数,如何建立高效网络结构成为一个挑战,针对此问题采取了GRU和MLP混合网络结构.

3 GM-FastText 模型

为了解决短文本分类中存在的特征稀疏、用词不规范等问题,本文根据FastText模型能产生多通道的词向量等特点,结合GRU和MLP等模型在特征提取上的优势提出了GM-FastText模型. GM-FastText模型流程图如图1所示,利用FastText模型产生3种不同的嵌入词向量编码,One-Emb、Two-Emb、Thr-Emb分别代表由1、2、3个字表示的向量.将One-Emb输入到GRU网络提取One-Emb信息,然后再输入到MLP-Layer层. Two-Emb和Thr-Emb则直接输入到MLP-Layer层.通过MLP层联系3组嵌入词向量,平均池化后连接全连接层得到分类结果.

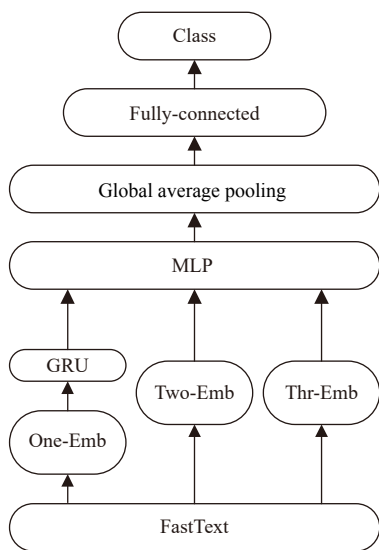


图1 GM-FastText 模型结构图

3.1 FastText

FastText 模型架构如图 2 所示, 只有输入层、隐藏层、输出层 3 层构成. 虽然其结构与 Word2Vec 模型的 CBOW 类似, 但是模型任务不同. 前者通过上下文预测中间词, 后者通过全部特征预测文本标签.

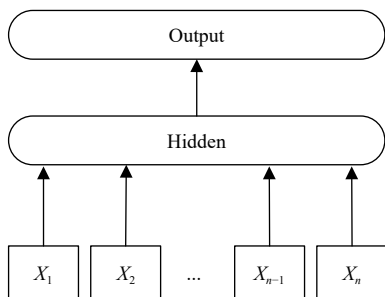


图2 FastText 模型结构

模型中 $\{X_1, X_2, \dots, X_{n-1}, X_n\}$ 表示文本中的特征词向量, 通过隐藏层将多个词向量叠加经过激活函数得到输出层的输入:

$$H_{doc} = \frac{1}{n} \sum_{i=1}^n X_i \tag{1}$$

$$Y = Sigmoid(W_0 H_{doc}) \tag{2}$$

其中, H_{doc} 表示平均文本特征, W_0 表示隐藏层权重矩阵, Y 表示输出层输入.

对于大量类别的数据集, 在做类别概率归一化时会占用非常长的时间, 因此在模型输入层中引入了分层 Softmax^[10], 其思想是根据类别的频率构造哈曼树

来代替标准 Softmax, 只需要计算一条路径上所有节点的概率值, 不需要在意其他节点, 可以将计算时间复杂度从 $O(N)$ 降低到 $O(\log N)$.

FastText 通过词袋模型获取文本特征的同时, 还融入了 N-gram 信息. 其基本思想是按指定的步长进行大小为 N 的窗口滑动, 最终得到片段长度 N 的序列. 这样对于一些罕见单词可以提供更好的词向量, 对于一些超出词表的单词可以通过字符级 N-gram 进行组合, 同时也面临着内存压力. FastText 模型随着语料库增加, 使用 Hash 的方式被分配到不同的桶中缓解内存压力.

3.2 GUR

为采用 GRU 模型处理字符集别的嵌入词向量, 这个模型由 Chung^[11] 提出, 是 RNN 的变种与 LSTM (long short-term memory) 结构相似. 然而 GUR 只有两个门控, 分别是更新门和重置门, 简化了模型结构计算效率更高, 同时也能解决梯度消失和梯度爆炸等文本信息丢失问题.

GRU 模型结构由图 3 所示, Z_t 为更新门, 是由隐藏状态 H_{t-1} 和当前输入 X_t 控制, 其计算过程如式 (3) 所示, W_z 表示为权重, 通过激活函数 Sigmoid 将结果映射到 0-1 之间, 结果越大表示存储下来的信息多, 反之越少, 有助于获取长序列依赖关系. R_t 为重置门决定了上一时刻的信息的获取程度, 如式 (4) 所示, 有助于获取短序列的依赖关系. H_t 表示当前序列的隐藏状态, 是由重置门和当前输入控制, 使用 tanh 激活函数结果映射在 -1~1 之间如式 (5) 所示. H_t 表示传递到下个序列信息, 通过更新门实现记忆更新, 其计算如式 (6).

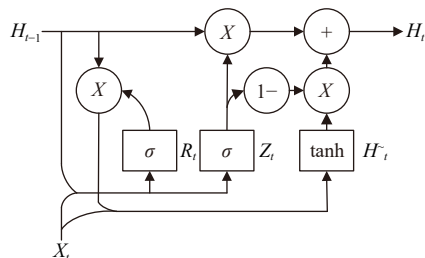


图3 GRU 模型结构图

$$Z_t = \sigma(W_Z \cdot [H_{t-1}, X_t]) \tag{3}$$

$$R_t = \sigma(W_R \cdot [H_{t-1}, X_t]) \tag{4}$$

$$H_t^{\sim} = \tanh(W_{H^{\sim}} \cdot [R_t * H_{t-1}, X_t]) \tag{5}$$

$$H_t = (1 - Z_t) * H_{t-1} + Z_t * \tilde{H}_t \quad (6)$$

3.3 MLP-Layer

为了使 3 种不同特征通道之间参数能有交互, 设计了 MLP-Layer 层^[12], 是由层归一化层 (LayerNorm)、全联接层和一个激活层构成, 其结构如图 4 所示. 层归一化是为了突出特征同时保持特征稳定的分布结构, 然后再通过全联接层和激活层输入到下一层. 其计算流程可以如式 (7), 其中 X 表示输入, Y 表示输出, W 表示训练权重, b 表示偏置, σ 表示激活函数.

$$Y = \sigma(W \cdot \text{LayerNorm}(X) + b) \quad (7)$$

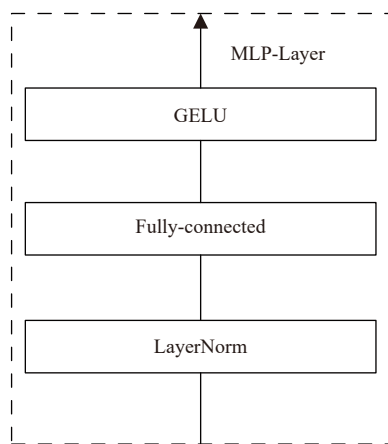


图 4 MLP-Layer 结构

4 实验结果分析

该实验将采取多个数据集与传统的基线模型做对比, 分析模型的性能, 实验使用 Python 3.8 版本, 采用 NVIDIA 1070 显卡作为计算平台.

4.1 数据集

该实验采用短新闻标题去验证模型的效果, 由于数据库过大, 限于计算资源, 分别从 THUCNews、Sogo 新闻库中选取 10 个类别按每个类别 2 万条, 然后每个类别抽取 2 000 条以 1:1 分为测试集和验证集. 由于头条新闻库单个类别数量少, 选取了 10 个类别每个类别只抽取 12 000 条再分别抽取 2 000 条以 1:1 分为测试集和验证集. 数据集的详细情况如表 1 所示.

表 1 数据集信息表

Data set	Train	Test	Validation	category
THUCNews	180000	10000	10000	10
SogoNews	180000	10000	10000	10
头条新闻库	100000	10000	10000	10

4.2 实验评估指标

实验将词向量维度设置为 300, 采用 Adam 优化器进行参数更新, 学习率设置为 0.001, 批处理设置为 128, 为防止过拟合随机失活率设置为 0.5. 文本分类常用指标包括精确率 P 、召回率 R 、 $F1$ 以及准确率 Accuracy, 由于 $F1$ 指标中已经包含了精确率 P 和召回率 R , 所以选择 $F1$ 和 Accuracy 作为评估指标.

为了验证 GM-FastText 模型的有效性, 主要从两个角度进行了对比. 首先使用相同的运行参数, 且都使用随机初始化嵌入词向量对比分析了 TextCNN、TextRNN、DPCNN^[13]、RCNN^[14]、Transformer、FastText 模型的文本分类性能. 同时为了分析 GM 结构的特征提取优势, 将 FastText 分别与 RNN、CNN 拼接后的模型 FastText-CNN、FastText-RNN 与 GM-FastText 模型也进行对比实验分析.

4.3 实验结果分析

各模型在 3 个数据集上的 $F1$ 值如表 2-表 4 所示. 从表 2 中可以看出, GM-FastText 模型在 THUNews 数据集中各分类的 $F1$ 值最好. 与传统文本分类模型 TextCNN、TextRNN 相比, GM-FastText 在该数据集上 $F1$ 值提升最大类别为 Sport 和 Stock 分别为 0.04 和 0.05, 10 个类别 $F1$ 值分别平均提升 0.02 和 0.019; 采用 FastText 词向量, GM-FastText 与 FastText, FastText-CNN 和 FastText-RNN 相比在该数据集上 $F1$ 值提升最大类别为 Stock、Entertainment 和 Stock 分别为 0.02、0.03 和 0.03, 10 个类别 $F1$ 值平均提升 0.005, 0.015, 0.011.

从表 3 中可以看出, 在头条新闻库数据集中除 Entertainment、Education 和 Travel 3 个类别外有 7 个类别达到了最优值. 与传统文本分类模型 TextCNN、TextRNN 相比, GM-FastText 在该数据集上 $F1$ 值提升最大类别为 Science 和 Word 分别为 0.03 和 0.04, 10 个类别 $F1$ 值分别平均提升 0.013 和 0.015; 采用 FastText 词向量, GM-FastText 与 FastText, FastText-CNN 和 FastText-RNN 相比在该数据集上 $F1$ 值提升最大类别为 Financial、Science 和 Word 分别为 0.03、0.03 和 0.03, 10 个类别 $F1$ 值平均提升 0.008, 0.008, 0.013.

从表 4 可以看出, 在 Sogo 新闻库数据中除 Travel 类别外其他 9 个分类能达到最好的效果. 与传统文本分类模型 TextCNN、TextRNN 相比, GM-FastText 在

该数据集上 $F1$ 值提升最大类别都为 Culture 分别为 0.07 和 0.06, 10 个类别 $F1$ 值分别平均提升 0.032 和 0.035; 采用 FastText 词向量, GM-FastText 与 FastText, FastText-CNN 和 FastText-RNN 相比, 在该数据集上 $F1$ 值提升最大类别为 Culture、Education 和 Science 分别为 0.02、0.05 和 0.05, 10 个类别 $F1$ 值平均提升

0.005, 0.019, 0.025.

根据以上分析, GM-FastText 与传统文本分类模型 TextCNN、TextRNN 相比, 在 3 个数据集上 $F1$ 平均提升 0.021 和 0.023; GM-FastText 相比于 FastText, FastText-CNN 和 FastText-RNN 在 3 个数据集上 10 个类别 $F1$ 平均提升 0.006, 0.014 和 0.016.

表 2 THUNews 数据集 10 类别 $F1$ 值

Model	Financial	Realty	Stocks	Education	Science	Society	Politics	Sports	Game	Entertainment
TextCNN	0.90	0.91	0.85	0.94	0.85	0.88	0.87	0.93	0.90	0.90
TextRNN	0.88	0.91	0.82	0.94	0.83	0.89	0.87	0.96	0.92	0.92
RCNN	0.89	0.91	0.84	0.95	0.85	0.90	0.87	0.96	0.92	0.92
DPCNN	0.89	0.91	0.83	0.94	0.85	0.90	0.87	0.96	0.92	0.92
FastText	0.89	0.92	0.85	0.95	0.85	0.90	0.89	0.97	0.93	0.93
Transformer	0.86	0.90	0.80	0.92	0.80	0.88	0.86	0.93	0.90	0.89
FastText-CNN	0.89	0.91	0.84	0.94	0.86	0.89	0.88	0.95	0.92	0.91
FastText-RNN	0.89	0.92	0.84	0.94	0.85	0.89	0.88	0.96	0.93	0.92
GM-FastText	0.90	0.92	0.87	0.95	0.86	0.90	0.89	0.97	0.93	0.94

表 3 头条新闻库数据集 10 个类别 $F1$ 值

Model	Culture	Entertainment	Sports	Financial	Car	Education	Science	Military	Travel	Word
TextCNN	0.87	0.88	0.92	0.83	0.91	0.90	0.84	0.87	0.88	0.82
TextRNN	0.87	0.88	0.91	0.84	0.90	0.88	0.85	0.87	0.86	0.81
RCNN	0.88	0.88	0.92	0.84	0.90	0.89	0.85	0.88	0.88	0.83
DPCNN	0.82	0.87	0.92	0.83	0.89	0.86	0.84	0.89	0.86	0.82
FastText	0.87	0.89	0.92	0.82	0.91	0.90	0.86	0.89	0.88	0.84
Transformer	0.82	0.82	0.88	0.78	0.87	0.85	0.81	0.85	0.79	0.77
FastText-CNN	0.89	0.89	0.93	0.83	0.91	0.90	0.84	0.89	0.87	0.83
FastText-RNN	0.87	0.87	0.91	0.84	0.91	0.90	0.86	0.87	0.86	0.82
GM-FastText	0.89	0.88	0.93	0.85	0.91	0.89	0.87	0.89	0.86	0.85

表 4 Sogo News 库数据集 10 个类别 $F1$ 值

Model	Culture	Entertainment	Sports	Financial	Car	Education	Science	Military	Travel	Word
TextCNN	0.83	0.80	0.93	0.96	0.85	0.89	0.85	0.96	0.92	0.93
TextRNN	0.84	0.80	0.93	0.96	0.85	0.88	0.84	0.97	0.92	0.92
RCNN	0.88	0.83	0.95	0.97	0.88	0.89	0.87	0.97	0.95	0.92
DPCNN	0.85	0.81	0.93	0.96	0.87	0.88	0.86	0.97	0.93	0.90
FastText	0.88	0.83	0.95	0.98	0.89	0.91	0.88	0.98	0.96	0.95
Transformer	0.81	0.79	0.88	0.96	0.83	0.86	0.83	0.96	0.89	0.89
FastText-CNN	0.88	0.84	0.94	0.96	0.87	0.87	0.86	0.97	0.93	0.94
FastText-RNN	0.85	0.82	0.93	0.96	0.87	0.88	0.84	0.97	0.95	0.93
GM-FastText	0.90	0.84	0.95	0.98	0.89	0.92	0.89	0.98	0.95	0.95

各模型在 3 个数据集上的准确率值如表 5 所示. 从表 5 中可以看出, GM-FastText 在不同的数据集上都达到了最高的值. 与传统文本分类模型 TextCNN、TextRNN 相比, GM-FastText 分别在 3 种数据集上准确率提升了 1.74、0.95、3.2 和 1.64、1.58、3.03 个百分点; 采用 FastText 词向量, GM-FastText 与 FastText、

FastText-CNN、FastText-RNN 相比在 3 个不同的数据集上的准确率分别提升, 0.56、0.28、0.43, 1.15、0.41、1.62 和 1.01、1.09、2.21 个百分点.

由以上数据可得, GM-FastText 与传统文本分类模型 TextCNN、TextRNN 相比在 3 个数据集上准确率平均提升 1.96、2.08 个百分点; GM-FastText 相比于

FastText, FastText-CNN 和 FastText-RNN 在 3 个数据集上准确率平均提升 0.42、1.06、1.41 个百分点。

表 5 各模型在不同数据集的准确率 (%)

Model	THUNews	头条新闻库	SogoNews
TextCNN	89.95	87.72	89.59
TextRNN	90.05	87.09	89.76
RCNN	90.61	88.04	91.65
DPCNN	90.03	86.28	90.13
FasTText	91.13	88.39	92.36
Transformer	87.88	82.73	87.49
FastText-CNN	90.50	88.26	91.17
FastText-RNN	90.68	87.58	90.67
GM-FastText	91.69	88.67	92.79

同时从表 2–表 5 可以看出 DPCNN 和 Transformer 两个神经网络模型, 相比其他网络模型准确率上缺乏优势, 说明仅依靠增加网络深度对于短文本处理没有明显的提升效果。

总之, 针对 3 个数据集表 2–表 5 中的 $F1$ 值和准确率的性能改善可知, FastText 多通道词向量在短文本分类中有更好的词向量表达和更加准确的特征表示; 且 GM 网络结构相对于传统的 CNN、RNN 模型在应对多通道大量数据时有更好的特征提取和整合能力。

5 结论与展望

通过词向量表征和模型结构两个切入点, 构建了一个多通道嵌入词的简易网络短文本分类模型 GM-FastText。通过 FastText 生成 3 种不同 N-gram 嵌入词向量, 以多通道的形式输入到 GM 结构中, 突出文本特征然后通过全联接层得到结果。GM-FastText 模型利用 N-gram 特殊的滑窗结构构建特殊的字词向量, 对于短新闻中一些极简词也有对应向量解决大多数的 OOV 问题, 提高了部分类别新闻分类准确率。尽管 MLP 提取整合多特征向量有着良好性能, 但是特征向量经过 1 次 MLP-Layer 层特征识别有限, 在接下来的工作中应着重优化 MLP-Layer 尝试多次经过 MLP-Layer 层多次提取词向量特征。

参考文献

- 杨朝强, 邵党国, 杨志豪, 等. 多特征融合的中文短文本分类模型. 小型微型计算机系统, 2020, 41(7): 1421–1426. [doi: 10.3969/j.issn.1000-1220.2020.07.013]
- 段丹丹, 唐加山, 温勇, 等. 基于 BERT 模型的中文短文本分类算法. 计算机工程, 2021, 47(1): 79–86.
- 杨世刚, 刘勇国. 融合语料库特征与图注意力网络的短文本分类方法. 计算机应用. <http://kns.cnki.net/kcms/detail/>

51.1307.TP.20210628.1334.015.html. (2021-06-28)[2021-07-30].

- Joulin A, Grave E, Bojanowski P, *et al.* Bag of tricks for efficient text classification. Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers. Valencia: Association for Computational Linguistics, 2017. 427–431.
- Kim Y. Convolutional neural networks for sentence classification. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing. Doha: Association for Computational Linguistics, 2014. 1746–1751.
- Mikolov T, Chen K, Corrado G, *et al.* Efficient estimation of word representations in vector space. arXiv: 1301.3781, 2013.
- 张焱博, 郭凯. 基于 FastText 和多融合特征的文本分类模型. 计算机仿真, 2021, 38(7): 461–466. [doi: 10.3969/j.issn.1006-9348.2021.07.098]
- 汪家成, 薛涛. 基于 FastText 和关键句提取的中文长文本分类. 计算机系统应用, 2021, 30(8): 213–218. [doi: 10.15888/j.cnki.csa.008007]
- 范昊, 李鹏飞. 基于 FastText 字向量与双向 GRU 循环神经网络的短文本情感分析研究——以微博评论文本为例. 情报科学, 2021, 39(4): 15–22. [doi: 10.13833/j.issn.1007-7634.2021.04.003]
- 吴震, 冉晓燕, 苗权, 等. 基于 fastText 算法的行业分类技术. 北京航空航天大学学报, 2022, 48(2): 193–198. [doi: 10.13700/j.bh.1001-5965.2020.0402.]
- Chung J, Gulcehre C, Cho KH, *et al.* Empirical evaluation of gated recurrent neural networks on sequence modeling. arXiv: 1412.3555, 2014.
- Tolstikhin IO, Houlsby N, Kolesnikov A, *et al.* MLP-mixer: An all-MLP architecture for vision. arXiv: 2105.01601, 2021.
- Johnson R, Zhang T. Deep pyramid convolutional neural networks for text categorization. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Vancouver: Association for Computational Linguistics, 2017. 562–570.
- Lai SW, Xu LH, Liu K, *et al.* Recurrent convolutional neural networks for text classification. Proceedings of the 29th AAAI Conference on Artificial Intelligence. Austin: AAAI, 2015. 2267–2273.

(校对责编: 孙君艳)