Journal of Computer Applications
计算机应用,2021,41(7):1897 - 1901
ISSN 1001-9081
CODEN JYIIDU
2021-07-10
http://www.joca.cn

文章编号:1001-9081(2021)07-1897-05
DOI:10.11772/j.issn.1001-9081.2020101528

基于注意力机制的文本作者识别

张　洋,江铭虎*
(清华大学 人文学院,北京 100084)
(*通信作者电子邮箱jiang.mh@mail.tsinghua.edu.cn)

摘　要:基于神经网络的作者识别在面临较多候选作者时识别准确率会大幅降低。为了提高作者识别精度,提出一种由快速文本分类(fastText)和注意力层构成的神经网络,并将该网络结合连续的词性标签n元组合(POS n-gram)特征进行中文小说的作者识别。与文本卷积神经网络(TextCNN)、文本循环神经网络(TextRNN)、长短期记忆(LSTM)网络和fastText进行对比,实验结果表明,所提出的模型获得了最高的分类准确率,与fastText模型相比,注意力机制的引入使得不同POS n-gram特征对应的准确率平均提高了2.14个百分点;同时,该模型保留了fastText的快速高效,且其所使用的文本特征可以推广到其他语言上。

关键词:作者识别;词性标签n元组合;神经网络;快速文本分类;注意力机制
中图分类号:TP391　　**文献标志码**:A

Authorship identification of text based on attention mechanism

ZHANG Yang, JIANG Minghu*
(*School of Humanities*, *Tsinghua University*, *Beijing* 100084, *China*)

Abstract: The accuracy of authorship identification based on deep neural network decreases significantly when faced with a large number of candidate authors. In order to improve the accuracy of authorship identification, a neural network consisting of fast text classification (fastText) and an attention layer was proposed, and it was combined with the continuous Part-Of-Speech (POS) n-gram features for authorship identification of Chinese novels. Compared with Text Convolutional Neural Network (TextCNN), Text Recurrent Neural Network (TextRNN), Long Short-Term Memory (LSTM) network and fastText, the experimental results show that the proposed model obtains the highest classification accuracy. Compared with the fastText model, the introduction of attention mechanism increases the accuracy corresponding to different POS n-gram features by 2.14 percentage points on average; meanwhile, the model retains the high-speed and efficiency of fastText, and the text features used by it can be applied to other languages.

Key words: authorship identification; Part-Of-Speech (POS) n-gram; neural network; fast text classification (fastText); attention mechanism

0　引言

互联网时代,海量数据涌现,人们在享受信息服务的同时也饱受信息泛滥的困扰。作者识别技术可以准确而及时地识别不良信息,追踪垃圾信息的源头并阻止其传播,对于维护互联网生态健康具有重要的意义。作者识别,又称为作者身份识别(authorship identification)或者作者身份归属(authorship attribution),是自然语言处理(Natural Language Processing, NLP)领域里的一个重要分支。顾名思义,作者识别是识别文本作者的一类研究,它最初源自人们深入阅读的传统。作者识别的主要思路是将文本中隐含的作者无意识的写作习惯通过某些可以量化的特征表现出来,进而凸显作品的文体学特征或写作风格,以此确定匿名文本的作者[1]。从其发展历程来看,最初的研究是确定散轶文献的来源或作者,后面又逐渐发展至确定某一文学作品、法律文档或者电子文本的作者。根据是否使用数学方法量化文本风格,可以将作者识别分为传统作者识别和现代作者识别[2]。传统作者识别多基于文学和语言学的相关知识,依靠专家的经验进行判断;而现代作者识别则基于数学建模,依靠模型的结果确定作者归属。

本文主要研究基于中文文本的现代作者识别,通常可以分为提取文本特征和建立预测作者的数学模型两个步骤。这两个步骤分别被研究者称为作者风格分析(authorship style analysis)和作者身份建模(authorship modeling)。具体来说,作者风格分析是提取能够量化作者写作风格的文本特征的过程,比如字符特征、词汇特征、句法特征、语义特征等。通常需要设计一个特征提取器,生成相应的特征向量,以便于在接下来的步骤中进行建模。而作者身份建模则是根据提取的这些文本特征或者已生成的特征向量建立相应的模型,进而预测文本作者的过程。有时,作者身份建模也指由文本建立预测作者归属模型的过程。通过构建特征集进行作者识别的研究都可以用以上这两个步骤来描述。相比之下,极少数不需要借助特征集识别作者的研究则缺少第一个步骤。此类研究通常直接利用原始文档进行建模,而无需额外的特征提取,比如基于压缩方法的作者识别[3]等。

收稿日期:2020-10-08;**修回日期**:2020-12-15;**录用日期**:2020-12-16。　**基金项目**:国家自然科学基金资助项目(62036001)。
作者简介:张洋(1990—),男,山东济南人,博士研究生,主要研究方向:作者身份识别、文本分类、情感分析;　江铭虎(1962—),男,江苏苏州人,教授,博士,主要研究方向:自然语言处理、神经网络语言处理、模式识别、人工智能。

从大的层面来分，作者身份建模主要分为基于轮廓的建模（profile-based modeling）和基于实例的建模（instance-based modeling）[4]。二者都是由训练文本构建作者归属模型的过程，它们的主要区别在于：在基于轮廓的建模中，每个作者的所有文本都会被累计处理。换句话说，特定作者的所有文本会被整合成一个大文档，根据这个大文档提取相应表示，构建该作者的轮廓。这样，每个测试文本只需跟特定作者的轮廓比较一次就能确定与该作者的相似程度。而在基于实例的建模中，每位作者的所有文本都会被单独处理。换句话说，每个文本都有自己的表示。在这种情况下，每个测试文本需要跟特定作者的所有文本进行比较才能确定与该作者的相似程度。因此，当语料相对比较充足，每个作者都有足够数量的训练文本时，通常采用基于实例的建模；反之，当仅能获得有限数量的训练文本时，常常采用基于轮廓的建模[5]。基于实例的建模通常会与机器学习算法搭配使用，因此研究者一般认为其准确率要高于基于轮廓的建模方法[6]。

1 相关研究

作者识别领域里常见的建模主要有基于概率的建模、基于向量空间的建模和基于相似度的建模等，下面简要叙述这几类建模以及它们通常搭配的分类方法。

1.1 基于概率的建模

基于概率的建模通过引入概率模型来描述不同随机变量之间的数学关系。作者识别领域最早的基于概率的建模是Mosteller等[7]利用贝叶斯方法研究联邦党人文集的作者归属问题。贝叶斯方法是一种建立在条件概率基础上的概率模型。具体来说，贝叶斯方法是在类条件概率密度和先验概率已知的情况下，通过贝叶斯公式比较样本属于两类的后验概率，将类别决策为后验概率大的一类，这样可以使总体错误率最小[8]。

有些研究者利用贝叶斯方法进行相关的研究。Zhao等[9]选择功能词和词性（Part-of-Speech，POS）标签作为特征，使用朴素贝叶斯方法识别作者；Raghavan等[10]为每个作者构建概率上下文无关文法，并使用该文法作为分类的语言模型进行作者归属；Boutwell[11]使用朴素贝叶斯分类器，选择基于字符的n元组合（n-gram）的特征构建作者集统计模型识别短信的作者；Savoy[12]利用隐含狄利克雷分布（Latent Dirichlet Allocation，LDA）把每个文档建模为主题分布的混合，每个主题指定单词的分布，根据文本距离确定相应的作者归属。

1.2 基于向量空间的建模

基于向量空间的建模把对文本内容的处理简化为向量空间中的向量运算，同时以向量空间中向量的相似度衡量文本中语义的相似度，简洁直观。作者识别领域的向量空间模型通常搭配支持向量机（Support Vector Machine，SVM）和神经网络等机器学习方法，本部分着重介绍这两种方法。

1.2.1 支持向量机

SVM是作者识别领域常见的一种方法，它的基本原理是找到一个最优的分类面，使得两类中距离这个分类面最近的点和分类面之间的距离最大[13]。SVM的复杂度与样本维数无关，学习效率和准确率较高，适合应用于高维文体特征数据集，因此受到很多研究者的青睐。Diederich等[14]利用SVM研究德国报纸文本的作者识别；Schwartz等[15]利用SVM研究微小信息在推特语料上的作者识别；Mikros等[16]结合多级n-gram，利用多类SVM研究希腊推文中的作者识别；Posadas-Duran等[17]选择句法关系标签、POS标签以及词根的句法n-gram等特征刻画文本风格，并利用SVM识别相关作者。

1.2.2 神经网络

神经网络是简单处理元件、单元或节点的互连系统，网络的处理能力体现在通过适应或学习一组训练模式的过程中获得的单元间连接强度或权重上[18]。因此，从本质上来说，神经网络是模拟动物脑中神经元网络的简化模型。从理论上来说，神经网络算法能够逼近任意函数，具有很强的非线性映射，以及分布存储、并行处理、自学习、自组织等优点[19]。所以，针对一些实际情况复杂、背景知识不清楚、规则不明确的问题，神经网络算法具有很强的处理能力。

有些研究者利用神经网络研究作者身份识别。Bagnall[20]使用循环神经网络（Recurrent Neural Network，RNN）同时对几个作者的语言进行建模；Ruder等[21]利用卷积神经网络（Convolutional Neural Network，CNN）处理特征级别信号，并对大规模文本进行快速预测；Shrestha等[22]选择字符n-gram作为特征，利用CNN对推文进行作者识别；Jafariakinabad等[23]使用句法循环神经网络从词性标签序列中学习句子的句法表示，同时利用CNN和长短期记忆（Long Short-Term Memory，LSTM）网络研究句中词性标签的短期和长期依赖性。

1.3 基于相似度的建模

基于相似度的建模的主要思想是计算未知文本和所有文本之间的相似性度量，然后根据相似程度估计最可能的作者[24]。这是分类任务中最直观的一种思路，该思路的代表算法是K-最近邻（K-Nearest Neighbor，KNN）算法。KNN的基本原理为：根据某个距离度量找出训练样本中与测试样本最接近的k个样本，再根据它们中的大多数样本标签进行预测。因此，衡量样本相似程度的距离度量是KNN或者其他基于相似度的分类方法的关键。KNN不需要使用训练数据来执行分类，可以在测试阶段使用训练数据[25]。

有些研究者利用基于相似度的模型研究作者识别。Jankowska等[26]选择通用n-gram相异性度量作为距离度量参与网络测评竞赛，获得了较优的结果；Burrows[27]提出了Delta方法，该方法通过计算未知文本与语料库的Z分数和Delta值，把文本分配给具有最低Delta值的作者；Eder[28]使用基于KNN的Delta方法研究文本尺寸对作者归属的影响。

2 注意力机制

近年来，注意力机制（attention mechanism）被广泛应用在图像识别、机器翻译、语音识别等各种深度学习任务中。顾名思义，注意力机制是模仿人识别物体时的注意力焦点的数学模型。人在识别物体时，先通过视觉系统获得物体的图像信息，而后由大脑对这些信息进行加工和整理，最终分辨物体的类别。大脑在对这些信息进行分析时，会格外关注一些局部信息，而忽略或者部分忽略其他信息。这种机制就被称为注意力机制。深度学习中的注意力机制利用Encoder和Decoder模型有效地赋予不同模块不同的权重，从而使得整个模型具有更强的分辨能力。

本文采用基于注意力机制的深度神经网络进行作者识别，整个作者识别流程如图1所示。原始文本经过降噪、分词、词性标注后提取其词性标签n元组合（POS n-gram）得到特征序列，特征序列经过Embedding层转化成相应的向量，然后在池化层取平均，再经过Attention层被赋予不同的权重，最后经过输出层得到分类结果。其中，Embedding层、池化层、Attention层和输出层构成了深度神经网络。

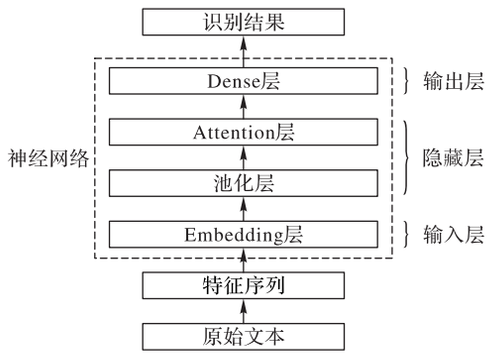

图1　作者识别流程

Fig. 1　Flowchart of authorship identification

2.1　Embedding层

神经网络的第一层是Embedding层,也叫输入层。它的输入是*batch_size*个POS *n*-gram序列,这些序列以数字编号(索引)的形式呈现,并且每个序列含有*seq_length*个索引。Embedding层将每个索引映射成*emb_dim*维的向量,以便于刻画不同特征之间的相互关系。

2.2　池化层

神经网络的第二层是池化层,主要用来对样本特征进行叠加平均。由于一篇文档被转化为*seq_length*个索引,每个索引又被映射成*emb_dim*维的向量,较多的特征数量会不可避免地引入很多噪声。鉴于此,可以利用池化操作对样本特征进行分组平均,通过设置*pool_size*的值可以控制分组大小。假设一个句子平均有N个词,设置*pool_size*的值为N,意味着每N个词进行一次叠加平均。这样,池化操作在降低噪声的同时赋予神经网络快速阅读的能力——从逐词阅读变为逐句阅读。

2.3　Attention层

神经网络的第三层是Attention层。池化层进行了特征的平均,这样能在很大程度上减小噪声的影响,避免过拟合,从而提高神经网络的准确率;然而处于不同位置的向量组合对分类的贡献不同,池化操作对此无能为力。因此本文引入注意力机制来给不同位置的向量组合分配不同的权重。注意力机制的示意图如图2所示。

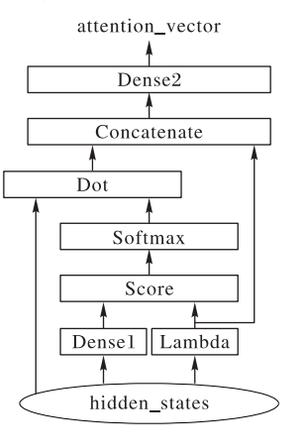

图2　注意力机制示意图

Fig. 2　Schematic diagram of attention mechanism

池化层的输出构成了Attention层的输入,即图2的隐藏层状态$\overline{\boldsymbol{h}_s}$。隐藏层状态$\overline{\boldsymbol{h}_s}$经过Dense1得到$\boldsymbol{W}\overline{\boldsymbol{h}_s}$。Dense1是无激活函数的全连接神经网络,它的作用是对隐藏层状态$\overline{\boldsymbol{h}_s}$进行线性变换。Lambda为自定义的切片函数,隐藏层状态$\overline{\boldsymbol{h}_s}$经过Lambda函数得到$\boldsymbol{h}_t$。$\overline{\boldsymbol{h}_s}$和$\boldsymbol{h}_t$的区别在于前者是源序列的隐藏层状态,而后者是目标序列的隐藏层状态,并且$\boldsymbol{h}_t = \overline{\boldsymbol{h}_s}[:,-1,:]$。

Score函数用于计算每个输入向量和查询向量之间的相关性。常见的Score函数有以下几种形式:

$$Score\left(\boldsymbol{h}_t,\overline{\boldsymbol{h}_s}\right)=\begin{cases}\boldsymbol{h}_t^{\mathrm{T}}\overline{\boldsymbol{h}_s}\\ \boldsymbol{h}_t^{\mathrm{T}}\boldsymbol{W}\overline{\boldsymbol{h}_s}\\ \boldsymbol{v}_a^{\mathrm{T}}\tanh\left(\boldsymbol{W}_1\boldsymbol{h}_t+\boldsymbol{W}_2\overline{\boldsymbol{h}_s}\right)\end{cases}\tag{1}$$

本文采用的是第二种形式,即$\overline{\boldsymbol{h}_s}$经过线性变换后再与$\boldsymbol{h}_t$进行点乘。接下来是注意力权重(attention weights)的计算。注意力权重由Score函数经过softmax函数得到,计算公式为:

$$\alpha_{ts}=\frac{\exp\left(Score\left(\boldsymbol{h}_t,\overline{\boldsymbol{h}_s}\right)\right)}{\sum_{s'=1}^{S}\exp\left(Score\left(\boldsymbol{h}_t,\overline{\boldsymbol{h}_{s'}}\right)\right)}\tag{2}$$

根据注意力权重可以计算原序列状态的权重均值,它等于注意力权重α_{ts}与隐藏层状态$\overline{\boldsymbol{h}_s}$点乘后求和。原序列状态的权重均值也被称为上下文向量(context vector),计算公式为:

$$\boldsymbol{c}_t=\sum_{s'}\alpha_{ts}\overline{\boldsymbol{h}_{s'}}\tag{3}$$

最终的注意力向量(attention vector)需要将上下文向量$\boldsymbol{c}_t$和目标序列的隐藏层状态$\boldsymbol{h}_t$连接后生成。Dense2是激活函数为tanh的全连接神经网络,用来对拼接后的向量进行tanh变换。注意力向量的计算公式为:

$$\boldsymbol{a}_t=f\left(\boldsymbol{c}_t,\boldsymbol{h}_t\right)=\tanh\left(\boldsymbol{W}_c\left[\boldsymbol{c}_t;\ \boldsymbol{h}_t\right]\right)\tag{4}$$

2.4　输出层

神经网络的第四层是输出层,用于最终的分类。本文直接采用激活函数为softmax的全连接神经网络完成分类。输出层的输出是样本属于不同类别的概率:

$$\boldsymbol{result}=\mathrm{softmax}\left(\boldsymbol{W}_a\boldsymbol{a}_t\right)\tag{5}$$

本文没有采用快速文本分类(fastText)[29]中的层次softmax,因为对于具有10位候选作者的作者识别任务,普通的softmax即可完成快速而高效的分类。此外,式(5)也可以写成:

$$\boldsymbol{result}=-\frac{1}{N}\sum_{n=1}^{N}y_n\left(\mathrm{softmax}\left(\boldsymbol{CBA}\boldsymbol{x}_n\right)\right)\tag{6}$$

其中:N表示样本的个数;$\boldsymbol{x}_n$表示第n个样本归一化后的特征向量,或者也可以理解为第n个样本的特征序列经过Embedding层生成的向量;y_n为第n个样本对应的类别标签;权重矩阵$\boldsymbol{A}$、$\boldsymbol{B}$和$\boldsymbol{C}$分别表示池化层对应的分组平均的权重矩阵、Attention层对应的分配权重的权重矩阵以及输出层对应的使用已学习的表示正确预测标签的权重矩阵。

3　实验及分析

本文选取莫言、路遥、贾平凹等10位作家的多部小说作品(共48.7 MB)作为语料进行研究。不同作者的语料规模如表1所示。

首先把同一位作家的多部作品进行合并,然后按照每个文档1 000字的长度进行分割。每位作家抽取1 000个文本进行实验,其中实验集、验证集和测试集的比例分别为:54%、6%和40%。作者的写作风格主要反映在其遣词造句的方式

上，换句话说，作者排列词语、组织句子的方式在很大程度上决定了其写作风格。因此，本文选择POS *n*-gram来进行作者识别。POS *n*-gram在很大程度上反映了作者词语选用和搭配的方式，进而体现作者的写作风格。之前关于*n*-gram的研究大多基于离散的特征，采用统计和机器学习模型相结合的分类方法，这些方法没有考虑特征之间的相互关系。本文采用连续*n*-gram特征，构建示例：

1）POS标签序列：pnvnfxdvrnux。

2）连续2-gram特征：pn、nv、vn、nf、fx、xd、dv、vr、rn、nu、ux。

3）连续3-gram特征：pnv、nvn、vnf、nfx、fxd、xdv、dvr、vrn、rnu、nux。

每一篇文档都会被转换成这样一串POS *n*-gram序列，然后又被转换成相应的数字序列，从而得到特征序列。特征序列会通过Embedding层转化成相应的向量，然后参与训练和分类过程。由于POS *n*-gram被转化成向量后，向量之间的距离可以反映这些POS组合的相近程度，因此这样的*n*-gram特征被称为连续特征[30]。普通*n*-gram特征可以表征作者词性搭配的频繁程度，但却无法表征语序信息；而连续*n*-gram不仅可以充分表征语序信息，还能通过向量之间的距离体现不同词性搭配之间的关系。换句话说，本文所采用的连续*n*-gram特征同时结合了*n*-gram特征和连续表示的优点，它既可以反映作者遣词造句的方式，又能够捕捉到不同词性搭配的差别。

表1 作者语料规模表

Tab. 1 Scale table of author corpus

作者名	文本大小/MB	作者名	文本大小/MB
莫言	3.6	刘斯奋	3.4
路遥	3.1	姚雪垠	8.8
贾平凹	4.0	刘震云	4.6
王旭烽	3.2	熊召政	3.4
张炜	10.3	王火	4.3

本文实验以文档为单位进行训练，文档中的词汇和标点符号均用POS标签的形式进行呈现。为了更好地训练模型，使用网格搜索来确定初始化向量维度、小批量大小、周期数、学习率等参数的最优组合。最终设置初始化向量维度为100，小批量大小为30，周期数为20，学习率为0.001。为了确定何种POS *n*-gram更能体现作者的写作风格，令*n*取1~5。特别地，当*n*=1时，意味着以单独的词性标签作为分类特征。分别使用文本卷积神经网络（Text Convolutional Neural Network，TextCNN）、文本循环神经网络（Text Recurrent Neural Network，TextRNN）、LSTM、fastText和本文模型对连续的POS *n*-gram特征进行训练和分类。表2展示了5种分类模型在不同*n*值下的实验结果，使用准确率进行评估。

表2 不同模型的准确率对比 单位：%

Tab. 2 Comparison of accuracy of different models unit: %

模型	1-gram	2-gram	3-gram	4-gram	5-gram
TextCNN	87.03	86.78	88.63	86.95	87.95
TextRNN	85.73	86.80	84.50	85.15	85.75
LSTM	83.60	83.98	85.60	84.05	83.88
fastText	91.53	90.88	91.40	91.30	91.48
本文模型	94.00	93.05	93.70	92.65	93.90

从表2中可以看出，不同模型的预测准确率从低到高排序依次为：LSTM、TextRNN、TextCNN、fastText和本文模型。换句话说，针对不同的POS *n*-gram特征，本文提出的基于注意力机制的识别模型均获得了最高的准确率。对于同一模型，不同*n*值对应的准确率差别不大，这说明对于采用连续POS *n*-gram特征进行中文小说作者识别的研究而言，*n*值的贡献远小于分类器对作者风格的捕捉。可以换一个角度进行理解，连续POS *n*-gram特征已经涵盖了文档的全部序列信息和词语搭配信息，此时*n*在1~5变化，并没有带来信息结构的变化。在这种情况下，分类器捕捉特征关联的能力主要取决于模型的训练，而并非初始特征组合的建构。

本文的基于注意力机制的神经网络分为输入层、池化层、Attention层和输出层四层结构。如果进行消融实验，去掉Attention层，则模型只剩下输入层、池化层和输出层三层结构。这是一个类似fastText的网络结构，区别在于该网络的输出层使用了普通的softmax，而fastText的输出层采用的是层次softmax。这个差异仅会对程序的运行时间产生影响，而不会影响作者识别的准确率，因此可以直接借用fastText的实验数据进行对比分析。与fastText的结果进行对比可以发现，加入Attention层后，不同POS *n*-gram特征对应的准确率平均提高了2.14个百分点。这是由于Attention机制给处于不同位置的向量组合分配了不同的权重，进而使得整个网络能够捕捉文档不同部分所反映出的作者风格，从而提高了准确率。因此可以得出这样的结论：注意力机制能够有效提高中文小说作者识别的准确率。

4 结语

本文提出一种基于注意力机制的神经网络，该网络通过在fastText的池化层和输出层之间添加Attention层得到。借助注意力机制，该网络能够捕捉文档不同部分所体现的作者风格，同时又保留了fastText快速而高效的特点。在结合连续POS *n*-gram特征进行的10位作者识别实验中，本文模型的准确率超过了TextCNN、TextRNN、LSTM和fastText这四种常见模型，比不添加注意力机制的fastText平均高出2.14个百分点。实验结果表明，神经网络中引入注意力机制能够有效提高中文小说作者识别的准确率。以后可以在以下几个方面继续改进原有工作：

1）尝试利用其他文本特征搭配注意力机制神经网络进行研究；

2）分析影响注意力网络的因素，例如文档长度、嵌入维度等；

3）改进注意力网络模型，比如由基于词语的Attention改为基于句子的Attention。

参考文献（References）

[1] 祁瑞华. 文本作者身份识别——基于机器学习与计算语言学[M]. 北京：清华大学出版社，2017：1.（QI R H. Text Authorship Identification - based on Machine Learning and Computational Linguistics[M]. Beijing: Tsinghua University Press, 2017: 1.）

[2] STAMATATOS E. A survey of modern authorship attribution methods [J]. Journal of the American Society for Information Science and Technology, 2009, 60(3): 538-556.

[3] CERRA D, DATCU M, REINARTZ P. Authorship analysis based on data compression[J]. Pattern Recognition Letters, 2014, 42: 79-84.

[4] POTHA N, STAMATATOS E. A profile-based method for authorship verification [C]// Proceedings of the 8th Hellenic

Conference on Artificial Intelligence, LNCS 8445. Cham: Springer, 2014: 313-326.

[5] MA J, XUE B, ZHANG M. A profile-based authorship attribution approach to forensic identification in Chinese online messages[C]// Proceedings of the 11th Pacific-Asia Workshop on Intelligence and Security Informatics, LNCS 9650. Cham: Springer, 2016: 33-52.

[6] KOPPEL M, SCHLER J, ARGAMON S. Authorship attribution in the wild[J]. Language Resources and Evaluation, 2011, 45(1): 83-94.

[7] MOSTELLER F, WALLACE D L. Inference and Disputed Authorship: the Federalist [M]. Wokingham: Addison-Wesley Publishing Company, 1964.

[8] 张学工. 模式识别[M]. 3版. 北京:清华大学出版社, 2010: 71-73. (ZHANG X G. Pattern Recognition[M]. 3rd ed. Beijing: Tsinghua University Press, 2010: 71-73.)

[9] ZHAO Y, ZOBEL J. Searching with style: authorship attribution in classic literature [C]// Proceedings of the 30th Australasian Conference on Computer Science. Sydney: Australian Computer Society, Inc., 2007: 59-68.

[10] RAGHAVAN S, KOVASHKA A, MOONEY R. Authorship attribution using probabilistic context-free grammars [C]// Proceedings of the ACL 2010 Conference Short Papers. Stroudsburg, PA: Association for Computational Linguistics, 2010: 38-42.

[11] BOUTWELL S R. Authorship attribution of short messages using multimodal features [D]. Monterey, CA: Naval Postgraduate School, 2011: 39-55.

[12] SAVOY J. Authorship attribution based on a probabilistic topic model [J]. Information Processing and Management, 2013, 49(1): 341-354.

[13] SCHÖLKOPF B, SMOLA A J. Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond[M]. Cambridge: MIT Press, 2018: 15-16.

[14] DIEDERICH J, KINDERMANN J, LEOPOLD E, et al. Authorship attribution with support vector machines[J]. Applied Intelligence, 2003, 19(1/2): 109-123.

[15] SCHWARTZ R, TSUR O, RAPPOPORT A, et al. Authorship attribution of micro-messages [C]// Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA: Association for Computational Linguistics, 2013: 1880-1891.

[16] MIKROS G K, PERIFANOS K. Authorship attribution in Greek tweets using author's multilevel n-gram profiles[C]// Proceedings of the 2013 AAAI Spring Symposium. Palo Alto, CA: AAAI Press, 2013: 17-23.

[17] POSADAS-DURAN J P, SIDOROV G, BATYRSHIN I. Complete syntactic n-grams as style markers for authorship attribution[C]// Proceedings of the 13th Mexican International Conference on Artificial Intelligence, LNCS 8856. Cham: Springer, 2014: 9-17.

[18] GURNEY K. An Introduction to Neural Networks [M]. Boca Raton: CRC Press, 1997: 1-3.

[19] GRAUPE D. Principles of Artificial Neural Networks[M]. 2nd ed. Singapore: World Scientific Publishing, 2007: 2-4.

[20] BAGNALL D. Author identification using multi-headed recurrent neural networks [EB/OL]. [2020-06-16]. https://arxiv.org/ftp/arxiv/papers/1506/1506.04891.pdf.

[21] RUDER S, GHAFFARI P, BRESLIN J G. Character-level and multi-channel convolutional neural networks for large-scale authorship attribution[EB/OL]. [2020-09-21]. https://arxiv.org/pdf/1609.06686.pdf.

[22] SHRESTHA P, SIERRA S, GONZÁLEZ F, et al. Convolutional neural networks for authorship attribution of short texts [C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2017: 669-674.

[23] JAFARIAKINABAD F, TARNPRADAB S, HUA K A. Syntactic recurrent neural network for authorship attribution [EB/OL]. [2020-02-26]. https://arxiv.org/pdf/1902.09723.pdf.

[24] QIAN T Y, LIU B, LI Q, et al. Review authorship attribution in a similarity space[J]. Journal of Computer Science and Technology, 2015, 30(1): 200-213.

[25] TRSTENJAK B, MIKAC S, DONKO D. KNN with TF-IDF based framework for text categorization [J]. Procedia Engineering, 2014, 69: 1356-1364.

[26] JANKOWSKA M, MILIOS E, KEŠELJ V. Author verification using common n-gram profiles of text documents[C]// Proceedings of the 25th International Conference on Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2014: 387-397.

[27] BURROWS J. 'Delta': a measure of stylistic difference and a guide to likely authorship[J]. Literary and Linguistic Computing, 2002, 17(3): 267-287.

[28] EDER M. Does size matter? Authorship attribution, small samples, big problem[J]. Digital Scholarship in the Humanities, 2015, 30(2): 167-182.

[29] JOULIN A, GRAVE E, BOJANOWSKI P, et al. Bag of tricks for efficient text classification [C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2017: 427-431.

[30] SARI Y, VLACHOS A, STEVENSON M. Continuous n-gram representations for authorship attribution[C]// Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2017: 267-273.

This work is partially supported by the National Natural Science Foundation of China (62036001).

ZHANG Yang, born in 1990, Ph. D. candidate. His research interests include authorship identification, text categorization, sentiment analysis.

JIANG Minghu, born in 1962, Ph. D., professor. His research interests include natural language processing, neural network language processing, pattern recognition, artificial intelligence.