融合文本困惑度特征和相似度特征的推特机器人检测方法*

王钟杰 张朝文 丁文琪 付雨濛 单丽莉 刘秉权
哈尔滨工业大学
{wzhongjie07, zhangzhaowen02, dingwenqi2020, cherry1232023} @163.com
{shanlili, liubq} @hit.edu.cn

摘要

推特机器人检测任务的目标是判断一个推特账号是真人账号还是自动化机器人账号。随着自动化账号拟人算法的快速迭代，检测最新类别的自动化账号变得越来越困难。最近，预训练语言模型在自然语言生成任务和其他任务上表现出了出色的水平，当这些预训练语言模型被用于推特文本自动生成时，会为推特机器人检测任务带来很大挑战。本文研究发现，困惑度偏低和相似度偏高的现象始终出现在不同时代自动化账号的历史推文中，且此现象不受预训练语言模型的影响。针对这些发现，本文提出了一种抽取历史推文困惑度特征和相似度特征的方法，并设计了一种特征融合策略，以更好地将这些新特征应用于已有的算法模型。本文方法在选定数据集上的性能超越了已有的基准方法，并在人民网主办、传播内容认知全国重点实验室承办的社交机器人识别大赛上取得了冠军。

关键词： 推特机器人检测 ； 预训练语言模型 ； 文本困惑度分析 ； 文本相似度分析

Twitter robot detection method based on text perplexity feature and similarity feature

ZhongjieWang ZhaowenZhang WenqiDing YumengFu LiliShan BingquanLiu
Harbin Institute of Technology
{wzhongjie07, zhangzhaowen02, dingwenqi2020, cherry1232023} @163.com
{shanlili, liubq} @hit.edu.cn

Abstract

The goal of the Twitter robot detection task is to determine whether a Twitter account is a real person account or an automated robot account. With the rapid iteration of automated account impersonation algorithms, it becomes increasingly difficult to detect the latest categories of automated accounts. Recently, pre-trained language models have shown excellent performance in natural language generation tasks and other tasks. When these pre-trained language models are used for automatic Twitter text generation, they pose significant challenges for Twitter robot detection tasks. This study found that the phenomenon of low perplexity and high similarity has always appeared in historical tweets from automated accounts in different eras, and this phenomenon is not affected by the pre-trained language model. In response to these findings, this paper proposes a method for extracting perplexity and similarity features from historical tweets, and designs a feature fusion strategy to better apply these new features to existing algorithm models. The performance of the method in this paper on the selected dataset exceeded the existing benchmark method, and won the championship in the social robot recognition competition which hosted by People's Daily and undertaken by the National Key Laboratory for Content Awareness Communication.

Keywords: twitter robot detection , pre-trained language model , text perplexity analysis , text similarity analysis

* 基金项目：国家重点研发计划（项目编号：2021YFF0901600）；中央高校基本科研业务费专项资金资助（项目编号：2022FRFK0600XX）

1 引言

推特社交机器人检测的目的是给定一名用户的历史推文和关系网络等信息（如图 1所示）判断其是真人账号还是自动化机器人账号。目前已经有一些推特机器人检测工作讨论了如何利用各种用户信息，如利用元信息进行推特机器人检测(Efthimion et al., 2018; Hayawi et al., 2022)；利用历史推文信息进行推特机器人检测(Derhab et al., 2021; Lundberg et al., 2019; Cresci et al., 2017)；利用关系网络信息进行推特机器人检测(Feng et al., 2022; Feng et al., 2021b)。然而，这些工作在检测最新一代自动化账号上的表现仍有进步的空间(Cresci, 2020)，其中一部分原因是这些工作对于历史推文信息的利用程度不够，导致自动化账号在更新其拟人算法后，这些检测算法原本所利用的信息可能会失效甚至误导检测结果。本文在分析具有代表性的几代自动化账号的基础之上发现，自动化账号的历史推文信息始终拥有较低的困惑度和较高的相似度，并且不随拟人算法的更新而变化。因此，为了更好地检测最新一代的自动化账号，本文提出了一种抽取历史推文困惑度特征和相似度特征的方法，并设计了一种特征融合策略，以更好地将这些新特征应用于已有的算法模型。

推特机器人检测工作的难点在于机器人拟人算法的快速迭代特性。Cresci (2020)总结了过去十年社交媒体机器人的发展情况，研究证据表明这些机器人之间存在着一种进化机制，进化的机器人通过协同完成任务，即一个群组的机器人被同一个人所控制，通过协同发布信息，从而实现更好的伪装。基于这一进化特点，最近的推特机器人检测算法更加注重关系网络信息，通过对这些关系网络信息进行建模，来提升模型检测效果(Feng et al., 2022; Feng et al., 2021b)。然而这些最新的检测工作也带来了新的问题。首先，关系网络信息的建模对收集数据提出了更高的要求，算法要求同时收集用户和相关用户的社交信息才能对关系网络信息完成建模。其次，这些基于关系网络信息的检测算法对关系网络结构十分敏感，如果关系网络过于稀疏或者当自动化账号故意增加与真人账号的互动之后，那么基于关系网络信息的检测算法可能会引入过多噪声，从而造成检测结果的误判。

研究表明，在推特机器人检测算法中，相对于其他信息，历史推文信息拥有更高的贡献权重。从直觉上来看，人们首先会根据一则账号发布的推文内容来怀疑其是否为自动化账号，如果产生怀疑，人们会进一步调查该账号的相关资料以及其与他账号的相关互动，以做出更准确的判断(Feng et al., 2021a)。第一代、第二代的机器人受技术限制，发布的推文内容很容易被人类识别。后来，由于大规模预训练语言模型的出现以及其在文本生成领域的成功应用(Devlin et al., 2018; Brown et al., 2020)，许多基于此项技术的推特机器人生成的推文内容已经很难被人类识别。但是这些生成的推文内容都有一个共同的特点，即相较于真实的人类推文，这些生成推文拥有更低的文本困惑度。这主要是因为早期的生成推文由于技术限制，往往只具有简单的句法结构或语义信息，而基于大规模预训练语言模型的生成推文则是受到其优化目标的影响，导致其更偏向于生成拥有更低文本困惑的推文内容。除此之外，相较于真实的人类推文，这些生成的推文内容之间往往具有更高的相似度，这主要是由推特机器人的任务性质所决定的，这些机器人只需要完成特定的目的，所以它们的推文只集中在某些特定话题下，而真实用户则更关注快速迭代的热点话题。

基于上述分析，本文在设计推特机器人检测算法模型时加入了文本困惑度特征和相似度特征。具体来说，本文利用Radford et al. (2018)提出的模型计算用户历史推文的文本困惑度，为每一个推特用户构建一个困惑度向量，并且统计了这些困惑度向量的五项数值特征，即最大值、最小值、均值、方差、总和，进而通过困惑度向量和其统计特征构建推特用户的困惑度特征。本文还利用Devlin et al. (2018)提出的模型计算这些推文的嵌入向量，再计算这些嵌入向量两两之间的相似度，得到一个相似度矩阵，最后统计相似度矩阵的均值，将其作为推特用户的相似度特征。本文利用一个简单的三层MLP模型验证了特征提取方法的有效性，此外，本文结合Feng et al. (2021b)提出的模型，提出并验证了一种特征融合策略。

本文的主要成果：

（1）分析用户历史推文信息的特点，提出了一种抽取推文文本困惑度和相似度特征的方法，并在数据集上验证了特征抽取方法的有效性。

（2）结合BOTRGCN模型(Feng et al., 2021b)，提出了一种特征融合策略，并在数据集上验证了特征融合策略的有效性。

（3）利用新的特征和特征融合策略，在一定程度上解决了机器人拟人算法快速迭代的问题。

（4）本文方法在选定数据集上的性能超越了已有的基准方法，并且在人民网主办、传播内容认知全国重点实验室承办的社交机器人识别大赛上取得了冠军。

2 相关工作

利用传统机器学习的推特机器人检测方法可以检测出早期的自动化账号。这些工作从数据中手工提取特征，包括元信息手工特征(Efthimion et al., 2018; Hayawi et al., 2022)、历史推文信息手工特征(徐帅帅et al., 2017; Derhab et al., 2021; Lundberg et al., 2019; Cresci et al., 2017)、关系网络信息手工特征(张玄and 李保滨, 2022)等,这些特征会被送入SVM、决策树、无监督聚类等传统机器学习算法中进行推特机器人检测。这些传统机器学习工作在提取特征时消耗了昂贵的资源，但是在检测最新一代的自动化账号时，并没有表现出很好的效果，这主要是因为提取的手工特征对账号信息的挖掘程度不够，当推特机器人更新其拟人算法后，已挖掘的特征会失效，甚至导致结果的误判。

深度学习技术的出现，使得研究者们能够进一步挖掘已有账号信息的特征。当CNN、LSTM以及后续的GPT(Radford et al., 2018)、BERT(Devlin et al., 2018)等深度学习模型出现之后，在传统机器学习中需要手工提取的特征逐渐被深度学习模型输出的嵌入向量所代替。Wei and Nguyen (2019)等人提出了第一个用深度学习模型进行词嵌入来完成推特机器人检测工作的模型，并在Cresci-2017数据集上达到了93%的准确度。Chen et al. (2022)等人则是利用预训练语言模型BERT对用户推文内容进行词嵌入，并论证了BERT得到的历史推文信息嵌入向量对推特机器人检测效果的提升有很大帮助。这些基于深度学习的模型和基于传统机器学习的模型之间的效果对比，说明了如果能够进一步挖掘已有账号信息的特征，就能够进一步提升模型的检测效果。

Cresci (2020)指出目前推特上活跃的自动化账号是第三代自动化账号，它们通过协同工作来实现更好的伪装。Kipf and Welling (2016)首次提出了图卷积的概念，CNN和RNN能够帮助我们从二维和一维的欧式空间数据中提取相应的特征，而图卷积的出现，则使我们能够从不规则的图结构中提取相应的特征。为了能够对最新一代自动化账号的异常协同工作进行识别，一些研究者将图卷积技术应用于用户关系网络信息的建模，他们将用户信息视为节点，将用户之间各种类型的互动关系视为不同类型的边，并用图卷积技术更新关系网络中节点和边的信息(Ali Alhosseini et al., 2019; Feng et al., 2022; Feng et al., 2021b)。然而这种基于图卷积技术的推特机器人检测工作也引入了新的问题，比如，对关系网络信息进行建模需要收集额外的数据，以及最终检测效果过度依赖于用户关系图的结构完整性。从本质上来看，这些对异常协同工作进行建模识别的检测算法相当于引入了新的特征，与充分挖掘已有账号信息的特征属于两种不同的思路。

在充分挖掘已有账号信息方面，还有一些研究者的工作值得关注。如Lei et al. (2022)提出了一种利用多模态技术挖掘历史推文信息中的语义不一致性信息，进而提升推特机器人检测效果的算法模型，该模型不仅关注历史推文信息中的文本信息，还关注推文中的图片和音视频信息。此外，ChatGPT[0]的出现使得由机器生成的文本更加难以和人类真实的文本作区分。Guo et al. (2023)提出了一种用于检测ChatGPT生成文本的工具，其核心思想是相较于人类真实的文本，由文本生成模型生成的文本有更低的文本困惑度。Gehrmann et al. (2019)也在他们的工作中论证了文本生成模型在解码阶段总是选择预测概率较高的词输出，这进一步导致了通过文本生成模型生成的文本拥有更低的文本困惑度。这些检测ChatGPT生成文本的工作，也为推特机器人检测工作带来新的思路。

综上所述，充分挖掘已有账号信息特征和引入新的特征都可以提高推特机器人检测的效果。但考虑到以下现实:

（1）引入新的特征会带来新的问题。

（2）历史推文信息比其余账号信息蕴含更多有助于检测的内容，且历史推文信息还没有被充分地利用。

（3）各种文本生成模型在推文自动生成上被广泛应用。

[0]https://chat.openai.com/

本文提出了一种抽取推文困惑度特征和相似度特征的方法，并设计了一种特征融合策略。新的特征以及特征融合策略兼顾了两种不同的问题解决思路，同时也在一定程度上对以上现实做了回应。实验结果也证明，引入的两个基于历史推文信息的特征比前人所提取的特征更有效，并且融合了这两者特征的检测模型能够在实验数据集上表现出更优秀的结果。

3 融合文本困惑度和相似度特征的推特机器人检测方法

3.1 任务定义

推特机器人检测任务旨在通过用户信息判断一个用户是真人账号还是自动化机器人账号，用户信息包含的内容如下。

个人简介信息：$B = \{b_i\}_{i=1}^{L}$，其中b_i代表一个单词。个人简介信息是一段由用户编写的文本，用来对用户所持有的账号做简单说明，如一些天气预报账号可能会在其个人简介中说明该账号是一个转发天气信息的自动化账号。

历史推文信息：$T = \{t_i\}_{i=1}^{M}$，其中$t_i = \left\{w_1^i, ..., w_{Q_i}^i\right\}$，$w_{Q_i}^i$代表一个单词。历史推文信息是用户在推特上进行活动产生的主要信息之一，用户往往通过发布推文来表达自己的观点。

元信息：$P = \{P^{num}, P^{cat}\}$，其中P^{num}表示数值元信息，P^{cat}表示分类元信息。元信息是用户创建时，推特平台为其分配的一些固有信息。数值元信息是指一些由数字构成的元信息，如：账号已激活时间、昵称长度等；分类元信息是指一些由布尔变量构成的元信息，如：账号是否经过认证；账号是否使用默认头像等。

关系网络信息：$N = \left\{N^f, N^t\right\}$，其中$N^f = \left\{N_1^f, ..., N_u^f\right\}$，表示该用户的$u$个追随者用户，$N^t = \{N_1^t, ..., N_v^t\}$，表示该用户的$v$个关注用户。关系网络信息构成一个社交网络，这些网络表明了哪些用户具有相似的话题爱好。

3.2 模型整体架构介绍

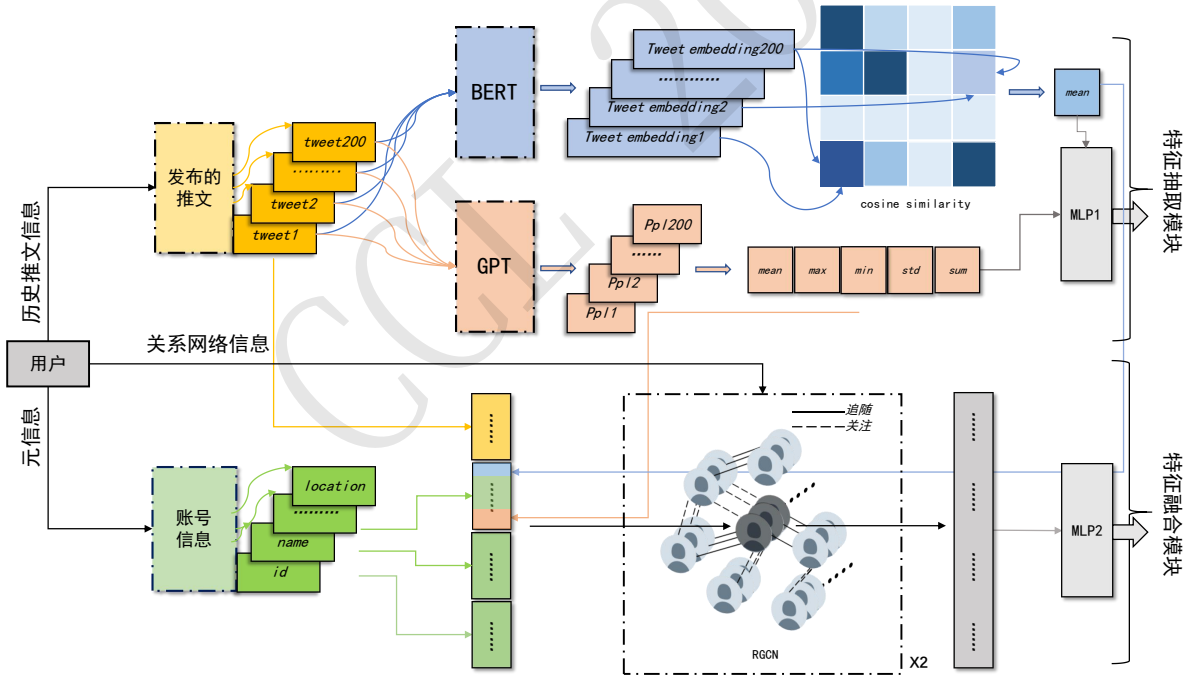

Figure 1: 模型架构图

本文提出了一种融合文本困惑度特征和相似度特征的推特机器人检测模型，如图 1所示。该模型主要由两部分组成：1）文本困惑度特征和相似度特征抽取；2）特征融合。第一部分展示了文本困惑度特征和相似度特征的抽取过程，第二部分则展示了特征融合策略，利用该策略，检测模型将抽取出的特征与BOTRGCN模型(Feng et al., 2021b)进行了融合。

各部分的具体工作流程如图 2所示。主要工作流程由三个阶段组成，分别是：特征抽取阶段、图处理阶段以及结果预测阶段。特征抽取阶段，模型对于输入的每个用户信息进行特征抽取；图处理阶段，模型通过抽取的特征初始化关系图中的每一个节点，通过关系网络信息初始化图中的每一条边，再通过图卷积技术更新图中的每一个节点，最后将每一个节点的特征送入预测网络；结果预测阶段，模型通过图卷积输出的某一节点的特征判断该节点是真人账号还是自动化账号。后续的 3.3章节和 3.4章节将对模型的工作细节进行更详细的阐述。

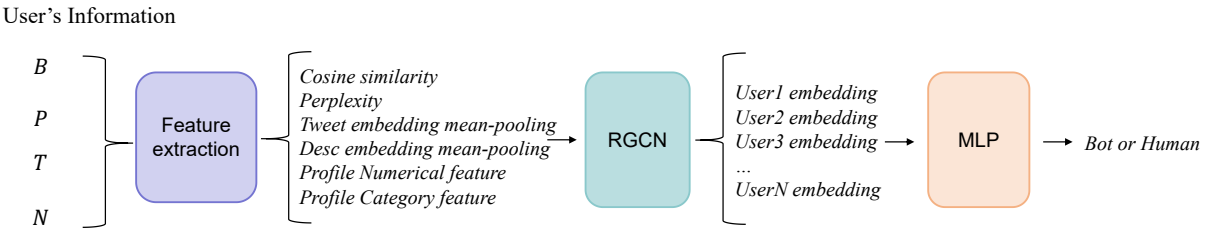

Figure 2: 算法工作流程图

3.3 特征抽取

文本困惑度特征：困惑度（perplexity）常被用来衡量一个预训练语言模型的能力，其计算过程见公式 (1)。显然，一个优秀的预训练语言模型应该在其相应的测试集上拥有更低的困惑度。

$$PP(W) = P(w_1 w_2 ... w_N)^{-\frac{1}{N}} = \sqrt[N]{\frac{1}{P(w_1 w_2 ... w_N)}} \tag{1}$$

本文使用GPT-1预训练语言模型(Radford et al., 2018)计算用户历史推文信息的困惑度，具体的说，检测模型将历史推文信息依次输入至GPT-1中计算其困惑度，计算完某一用户的所有历史推文信息困惑度，可以得到困惑度向量l'，$l' \in R^{1\times 200}$。检测模型还会计算l'的最大值、最小值、方差、均值和总和这五个统计特征，将其分别记做l'_{max}、l'_{min}、l'_{std}、l'_{mean}、$l'_{sum} \in R$，最终检测模型抽取的困惑度特征$l = concat(l'_{max}, l'_{min}, l'_{std}, l'_{mean}, l'_{sum})$，$l \in R^{1\times 5}$。

文本相似度特征：相似度（similarity）常被用来衡量文本内容之间的语义相似性，常用的计算方法有余弦相似度、Jaccard相似度等。本文选用余弦相似度作为计算公式，计算过程见公式 (2)。

$$\cos\theta = \frac{\vec{A}\cdot\vec{B}}{\left|\vec{A}\right|\left|\vec{B}\right|} = \frac{\sum_{i=1}^{n}(A_i \times B_i)}{\sqrt{\sum_{i=1}^{n}(A_i)^2}\sqrt{\sum_{i=1}^{n}(B_i)^2}} \tag{2}$$

本文使用BERT预训练语言模型(Devlin et al., 2018)计算这些用户历史推文信息的文本相似度，具体的说，检测模型将历史推文信息依次输入至BERT，使用BERT输出的[CLS]向量作为一条历史推文信息的嵌入向量表示，再计算这些嵌入向量之间的余弦相似度，进而得到相似度矩阵C'，$C' \in R^{200\times 200}$。检测模型还计算了C'的均值C'_{mean}，$C'_{mean} \in R$，将$C = C'_{mean}$，作为最终的相似度特征。

粗粒度的历史推文信息特征：BOTRGCN对历史推文信息进行了粗粒度的特征提取，具体的说，BOTRGCN将历史推文信息依次输入至预训练语言模型RoBERTa(Liu et al., 2019)，对于每一条推文，通过对RoBERTa输出的[TOKEN]向量进行平均池化得到该条推文的嵌入向量，再将推文的嵌入向量进行平均池化，得到历史推文信息特征f_t，$f_t \in R^{1\times 768}$。

实际上，推特中的推文涉及英语、中文在内的多种语言，除此之外，其语言习惯也受一些网络文化影响。因此，本文还尝试了用TwHIN-BERT(Zhang et al., 2022)对历史推文信息进行粗粒度的特征提取。相较于RoBERTa，TwHIN-BERT由推特上的七十亿条推文数据训练而成，并且具备处理多语种的能力，这使其能够更好地提取历史推文信息特征。$f_{t'}$，$f_{t'} \in R^{1\times 1024}$就是检测模型通过TwHIN-BERT提取的历史推文信息特征。

个人简介信息特征：BOTRGCN将个人简介信息输入至RoBERTa，通过对RoBERTa输出的[TOKEN]向量进行平均池化得到个人简介信息特征f_b，$f_b \in R^{1\times768}$。

元信息特征：对于元信息，BOTRGCN将这些离散的数字或布尔变量输入至一个三层的MLP模型，通过MLP进行编码，得到两个稠密的嵌入向量f_{num}，$f_{num} \in R^{1\times32}$；$f_{cat}$，$f_{cat} \in R^{1\times32}$，其分别表示数值元信息特征和分类元信息特征。

3.4 图操作和结果预测

图初始化操作和特征融合策略：BOTRGCN在对关系网络信息进行建模时，将用户视为节点，将用户之间的关注与被关注关系视为边，从而构建出一张异质图，其中节点特征由 3.3中提取的特征构成。具体的说，针对某一用户节点x_i，其初始化特征$x_i^0 = concat(MLP_1(f_b), MLP_2(f_t), f_{num}, f_{cat})$，$x_i^0 \in R^{1\times128}$，$MLP_1$和$MLP_2$为两个三层的MLP模型，通过它们进行编码，对$f_b$，$f_t$进行降维处理。

在BOTRGCN的工作基础之上，本文还将l'_{max}、l'_{min}、l'_{std}、l'_{mean}、l'_{sum}、C视为六个分类元信息，拼接至原元信息后，从而使得后续得到的f_{cat}能够融入困惑度特征和相似度特征。

图节点更新操作：采用RGCN(Schlichtkrull et al., 2018)更新图中节点特征，更新过程见公式 (3)，其中Θ_{self}和Θ_r为RGCN卷积算子需要学习的参数。

$$x_i^{(l+1)} = \Theta_{self} \cdot x_i^{(l)} + \sum_{r\in N}\sum_{j\in N_r(i)} \frac{1}{|N_r(i)|}\Theta_r \cdot x_j^{(l)}, x_i^{(l+1)} \in R^{1\times128} \tag{3}$$

结果预测：在预测时，BOTRGCN将RGCN的最后一层输出输入至头部网络，得到最终结果$\hat{y}$，$\hat{y} \in R^{1\times2}$（见公式 (4)）。训练时的损失函数采用交叉熵损失函数，并且采用$L2$正则化防止模型过拟合，Y为真实标签集合（见公式 (5)）。

$$\hat{y}_i = softmax\left(W_O \cdot x_i^L + b_O\right) \tag{4}$$

$$L = -\sum_{i\in Y}[y_i log(\hat{y}_i) + (1-y_i)log(1-\hat{y}_i)] + \lambda\sum_{w\in\Theta} w^2 \tag{5}$$

4 实验设计与分析

4.1 数据集

实验使用人民网主办、传播内容认知全国重点实验室承办的社交机器人识别大赛提供的数据作为主要数据集。该数据集收集了推特平台上的10500个账号信息，实验按照8：1：1的比例划分训练集、验证集、测试集，数据集各部分的样本分布见表 1，其中每条数据的格式如表 2所示。

所属集合	机器人样本数	真人样本数	总样本数
训练集	1714	6686	8400
验证集	204	846	1050
测试集	191	859	1050

Table 1: 样本分布

数据域	样例
个人简介信息	Ambition is preiceless
历史推文信息	[@tha..., @STU..., ...]
元信息	{id : ..., name : ..., ...}
关系网络信息	{following : [...] , follower : [266181184, ...]}

Table 2: 数据集格式

需要具体说明的是，每条数据中的历史推文信息被组织成列表的形式，每个列表包含了该账号发布的二百条历史推文；元信息被组织成字典的形式，按照填充值数据类型的不同，元信息可以分为数值元信息（见表 3）和分类元信息（见表 4）；关系网络信息被组织成字典的形式，以$follower$（粉丝）为例，其对应的列表中包含了该账号所有粉丝的id。

数值元信息特征	备注
followers	关注者数量
followings	追随者数量
favorites	点赞数量
statuses	状态数量
active_days	账户激活时间
screen_name_length	网名长度

Table 3: 数值元信息特征

分类元信息特征	备注
protected	账号是否受保护
geo_enabled	是否允许定位
verified	账号是否经过验证
contributors_enabled	是否允许有贡献者
is_translator	是否是翻译人员
is_translation_enabled	账号信息是否允许被翻译
profile_background_tile	背景图片是否平铺
profile_user_background_image	背景图片url
has_extended_profile	是否启用扩展资料
default_profile	是否使用默认个人资料
default_profile_image	是否使用默认个人资料图片

Table 4: 分类元信息特征

本文还统计了每条数据实际收录的$following$和$follower$的平均数量，并将其和数值元信息中记录的$followings$和$followers$作对比。两者实际收录的平均长度为0.58和0.64，而两者在数值元信息中记录的平均长度为3142和101027。这说明对关系网络信息进行建模时，收集一个用户的相关用户信息的代价是昂贵的。

4.2 实验设置

本文使用RoBERTa和TwHIN-BERT作为编码器对历史推文信息进行编码；使用distilRoBERTa作为编码器对个人简介信息进行编码；使用GPT-1和BERT分别提取历史推文信息的困惑度特征和相似度特征。在反向传播时，训练轮数设置为500轮；由于对关系网络建模需要全部训练样本，因此batchsize设置为训练集样本数量大小；学习率设置为0.001；$L-2$正则化系数λ设置为0.005；dropout设置为0.3；最大序列长度设置为512；并使用AdamW优化器进行梯度更新。

本文采用宏F1作为评价指标，计算如公式 (6)至公式 (8)所示，TP、FP、TN、FN分别为真阳性、假阳性、真阴性、假阴性样本。

$$F1_{marco} = \frac{2 \times precision \times recall}{precision + recall} \tag{6}$$

$$precision = \frac{TP}{TP + FP} \tag{7}$$

$$recall = \frac{TP}{TP + FN} \tag{8}$$

4.3 实验结果与分析

本文分别设计实验，验证了困惑度特征和相似度特征相较于前人总结的特征的有效性；以及特征融合策略的有效性。

为了验证困惑度特征和相似度特征相较于前人总结的特征的有效性，本文将这些特征分成四组（如表 5所示）。

分组编号	特征	备注
group1(g1)	数值元信息特征	同表 3
group2(g2)	分类元信息特征	同表 4
group3(g3)	g1+g2	/
group4(g4)	困惑度和相似度特征	/

Table 5: 特征分组

本文试图用最简单的模型研究这些特征的有效性，因此实验直接将这四组特征输入至一个三层MLP模型中，考虑到特征维数的不同，实验只对MLP的输入维数进行了改动，并没有改动MLP的深度和宽度，实验结果如表 6所示。

分组编号	$\mathbf{F1_{macro}}$
g1	0.4420
g2	0.4500
g3	0.4688
g4	**0.4741**

Table 6: 特征有效性实验结果

表 6的实验结果显示，检测模型抽取的困惑度特征和相似度特征在简单模型上显著优于前人总结的特征及其特征的组合。这也证明了用户的历史推文信息比元信息蕴含更多的价值。

为了验证融合策略的有效性，本文在BOTRGCN的工作基础之上融合了文本困惑度特征和相似度特征，实验结果如表 7所示[1]。

算法模型	第一次	第二次	第三次	第四次	第五次	平均$\mathbf{F1_{macro}}$
BOTRGCN	0.8536	0.8539	0.8570	0.8531	0.8538	0.8543
BOTRGCN+g4	0.8445	0.8368	0.8324	0.8426	0.8485	0.8410
BOTGAT+$f_{t'}$	0.8236	0.8238	0.8254	0.8199	0.8196	0.8225
BOTGAT+$f_{t'}$+g4	0.8363	0.8390	0.8267	0.8298	0.8376	0.8339
BOTRGCN+$f_{t'}$	0.8583	0.8556	0.8577	0.8458	0.8594	0.8554
BOTRGCN+$f_{t'}$+g4	**0.8639**	**0.8612**	**0.8595**	**0.8570**	**0.8631**	**0.8610**

Table 7: 融合策略有效性实验结果

表 7的实验结果显示，融合了困惑度和相似度特征的BOTRGCN在实验数据集上取得了更好的表现。除此之外，采用TwHIN-BERT对历史推文信息进行编码也提升了检测模型的性能表现。

为了研究困惑度和相似度特征的鲁棒性，本文还采取BOTGAT(Veličković et al., 2017)作为基础模型架构，并融合困惑度和相似度特征进行实验验证，可以看到困惑度和相似度特征在不同的基础模型架构上也有良好的表现。

此外，本文分析实验结果发现BOTRGCN在初始化图节点时存在特征融合瓶颈。具体的说，BOTRGCN初始化图节点只是简单地拼接不同类型的特征，没有完全发挥它们的潜力。如表 8所示，理论上用户的个人简介信息和历史推文信息是相互独立的，但是在融合f_t和f_b特征

[1]由于PyG库中graph算子的影响，即使固定随机种子，也无法完全固定实验结果，因此本文在同一随机种子上重复实验了五次，后续涉及graph算子的实验同理

时，相较于只使用f_t特征，模型预测效果没有明显提升，这说明只是简单地拼接两个不同类型的特征并没有挖掘出f_b特征的全部潜力，为了更好地提升模型效果，需要研究者们设计出新的特征融合策略。

4.4 消融实验

为了分析个人简介信息特征、数值元信息特征、分类元信息特征、粗粒度历史推文信息特征、文本困惑度特征和文本相似度特征对最终分类结果的影响，实验在用户关系图初始化时，只选用某一个（组）特征初始化节点，从而探究其对最终结果的影响权重，实验结果如表 8所示。

选取特征(组)	第一次	第二次	第三次	第四次	第五次	平均$\mathbf{F1_{macro}}$
f_b	0.7293	0.7223	0.7234	0.7199	0.7264	0.7243
f_{num}	0.5507	0.5617	0.5623	0.5627	0.5534	0.5582
f_{cat}	0.4500	0.4500	0.4500	0.4500	0.4500	0.4500
f_t	0.8445	0.8445	0.8375	0.8438	**0.8438**	0.8428
f_t+f_b	**0.8511**	**0.8531**	**0.8585**	**0.8446**	0.8411	**0.8497**
g3	0.5446	0.5453	0.5598	0.5559	0.5624	0.5536
g4	0.6954	0.6986	0.7402	0.6988	0.7042	0.7074

Table 8: 单一特征实验结果

表 8的实验结果显示，当只采用某一个（组）特征时，f_t、$g4$这些基于历史推文信息的特征对最终结果的正面影响显著大于基于其他用户信息的特征。这验证了本文在前面的论述，即判断一个用户是否是自动化账号时，其发布的推文往往更有价值。除此之外，结合表 6，可以发现与MLP模型相比，结合BOTRGCN模型并不会使f_{cat}特征对检测效果有明显提升，这或许是因为目前的自动化账号更加倾向于盗窃真人账号的分类元信息，而不是通过自动化程序生成，从而使得模型无法捕捉到同一组自动化账号之间分类元信息的相似性。

5 总结与展望

本文为了解决推特机器人拟人算法快速迭代，以及最新一代推特机器人检测算法对数据要求高、对图网络结构敏感的问题，提出了一种融合文本困惑度特征和相似度特征的推特机器人检测算法。本文认为，在进行推特机器人检测时，用户的历史推文信息相较于其他的用户信息，会为检测工作提供更多有价值的内容，因此，本文提出一种抽取推文文本困惑度和相似度特征的方法，旨在充分挖掘历史推文信息的价值。实验结果验证了提取特征的有效性；以及这些特征对最新一代推特机器人检测算法有明显的效果提升。本文分析了数据集中的数据，发现这些数据具有两个特点：多语种和语言习惯网络化。为了更好地捕捉文本特征，本文采用由七十亿条推特数据训练的多语种预训练语言模型TwHIN-BERT代替RoBERTa作为检测模型的编码器。除此之外，通过实验验证，本文还发现目前的算法在对不同类型的特征进行融合时采取的策略过于简单，这会导致特征融合瓶颈现象。虽然本文没有详细讨论，但仍值得注意的一点是，推特机器人种类的多样性(张玄and 李保滨, 2022)也是推特机器人检测算法需要应对的问题，像僵尸粉丝机器人这种类型的自动化账号可能很少发布推文，导致抽取出的历史推文信息特征质量较低，进而影响模型的性能表现。

综上所述，本文认为未来的推特机器人检测工作有如下几个发展方向：

- 提出更好的特征融合策略。目前的工作已经总结出各种类型的用户特征，但是忽视了对于特征融合策略的设计，更好的特征融合策略或许可以更好地利用已有的各种特征。

- 考虑不同种类的推特机器人。大部分推特机器人以发布各种有害推文为主，但是研究工作也不能忽视几乎不发布推文的自动化账号，如僵尸粉丝机器人等。同时，有些自动化账号发布的推文对人们的日常生活有益，如天气预报机器人等，未来的研究工作或许需要考虑更详细的分类标准，将这些有益的自动化账号剔除出自动化账号的检测范围。

- 更充分地挖掘历史推文信息。历史推文信息中还有许多有价值的内容可供挖掘，比如推文中的多模态信息。

参考文献

Seyed Ali Alhosseini, Raad Bin Tareaf, Pejman Najafi, and Christoph Meinel. 2019. Detect me if you can: Spam bot detection using inductive representation learning. In *Companion Proceedings of The 2019 World Wide Web Conference*, pages 148–153.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yeyang Chen, Mondher Bouazizi, and Tomoaki Ohtsuki. 2022. Social robot detection using roberta classifier and random forest regressor with similarity analysis. In *GLOBECOM 2022-2022 IEEE Global Communications Conference*, pages 6433–6438. IEEE.

Stefano Cresci, Roberto Di Pietro, Marinella Petrocchi, Angelo Spognardi, and Maurizio Tesconi. 2017. Social fingerprinting: detection of spambot groups through dna-inspired behavioral modeling. *IEEE Transactions on Dependable and Secure Computing*, 15(4):561–576.

Stefano Cresci. 2020. A decade of social bot detection. *Communications of the ACM*, 63(10):72–83.

Abdelouahid Derhab, Rahaf Alawwad, Khawlah Dehwah, Noshina Tariq, Farrukh Aslam Khan, and Jalal Al-Muhtadi. 2021. Tweet-based bot detection using big data analytics. *IEEE Access*, 9:65988–66005.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Phillip George Efthimion, Scott Payne, and Nicholas Proferes. 2018. Supervised machine learning bot detection techniques to identify social twitter bots. *SMU Data Science Review*, 1(2):5.

Shangbin Feng, Herun Wan, Ningnan Wang, Jundong Li, and Minnan Luo. 2021a. Twibot-20: A comprehensive twitter bot detection benchmark. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*, pages 4485–4494.

Shangbin Feng, Herun Wan, Ningnan Wang, and Minnan Luo. 2021b. Botrgcn: Twitter bot detection with relational graph convolutional networks. In *Proceedings of the 2021 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, pages 236–239.

Shangbin Feng, Zhaoxuan Tan, Rui Li, and Minnan Luo. 2022. Heterogeneity-aware twitter bot detection with relational graph transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 3977–3985.

Sebastian Gehrmann, Hendrik Strobelt, and Alexander M Rush. 2019. Gltr: Statistical detection and visualization of generated text. *arXiv preprint arXiv:1906.04043*.

Biyang Guo, Xin Zhang, Ziyuan Wang, Minqi Jiang, Jinran Nie, Yuxuan Ding, Jianwei Yue, and Yupeng Wu. 2023. How close is chatgpt to human experts? comparison corpus, evaluation, and detection. *arXiv preprint arXiv:2301.07597*.

Kadhim Hayawi, Sujith Mathew, Neethu Venugopal, Mohammad M Masud, and Pin-Han Ho. 2022. Deeprobot: a hybrid deep neural network model for social bot detection based on user profile data. *Social Network Analysis and Mining*, 12(1):43.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

Zhenyu Lei, Herun Wan, Wenqian Zhang, Shangbin Feng, Zilong Chen, Qinghua Zheng, and Minnan Luo. 2022. Bic: Twitter bot detection with text-graph interaction and semantic consistency. *arXiv preprint arXiv:2208.08320*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Jonas Lundberg, Jonas Nordqvist, and Mikko Laitinen. 2019. Towards a language independent twitter bot detector. In *DHN*, pages 308–319.

Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, et al. 2018. Improving language understanding by generative pre-training.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *The Semantic Web: 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3–7, 2018, Proceedings 15*, pages 593–607. Springer.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. 2017. Graph attention networks. *arXiv preprint arXiv:1710.10903*.

Feng Wei and Uyen Trang Nguyen. 2019. Twitter bot detection using bidirectional long short-term memory neural networks and word embeddings. In *2019 First IEEE International conference on trust, privacy and security in intelligent systems and applications (TPS-ISA)*, pages 101–109. IEEE.

Xinyang Zhang, Yury Malkov, Omar Florez, Serim Park, Brian McWilliams, Jiawei Han, and Ahmed El-Kishky. 2022. Twhin-bert: A socially-enriched pre-trained language model for multilingual tweet representations. *arXiv preprint arXiv:2209.07562*.

张玄and 李保滨. 2022. 微博环境中的机器人账户检测综述. 中文信息学报, 36(12):1–15.

徐帅帅, 戴新宇, 黄书剑, and 陈家骏. 2017. 基于无指导学习的微博评论分析方法. 中文信息学报, 31(2):179–186.