重要 Tor 暗网站点的验证码快速识别和数据采集

龙 军 王轶骏 薛 质
(上海交通大学电子信息与电气工程学院 上海 200240)

摘 要 针对面向 Tor 暗网的数据采集和信息监控的任务,为了解决爬取重要 Web 站点中所遇到的验证码自动识别这个技术难点,设计一套结合 CNN 网络、GRU 网络和 ctc loss 的快速识别模型,并将其应用到 Tor 暗网站点的数据采集系统中去。一段时间的实际运行结果充分证明了该 Tor 暗网数据采集系统能够快速、准确地识别重要 Tor 暗网站点的验证码,自动绕过检验机制后爬取并存储站点的数据信息,从而有力支撑了暗网数据提炼、分析和挖掘的后续工作。

关键词 Tor 暗网 CNN 网络 GRU 网络 ctc loss 算法 Scrapy 爬虫

中图分类号 TP3 **文献标志码** A DOI:10.3969/j.issn.1000-386x.2022.07.048

QUICK IDENTIFICATION OF VERIFICATION CODES AND DATA COLLECTION ON KEY TOR DARK WEB SITES

Long Jun Wang Yijun Xue Zhi
(*School of Electronic Information and Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China*)

Abstract At present, in the dark web space, more than 80% of dark web sites constructed using the principle of Tor technology are used to publish important information such as trading markets, black forums and publicity. Therefore, crawling data information of important Tor dark web sites in real time and accurately is a great significance for monitoring the resources of the dark web, obtaining threat intelligence information and perceiving the security posture of the network. This paper was oriented to the tasks of data collection and information monitoring of the Tor dark web, in order to solve the technical difficulty of automatic identification of verification codes encountered in crawling important web sites, a fast identification model combining CNN network, GRU network and ctc loss was designed and then applied to the data acquisition system of Tor dark web sites. A period of actual operation results fully proved that the Tor dark web data acquisition system could quickly and accurately identify the verification codes of important Tor dark web sites, automatically bypass the inspection mechanism the crawl and store the site's data information, thus it could provide strong support to the follow-up work of refining, analysis and mining of dark web data.

Keywords Tor dark web CNN network GRU network Ctc loss algorithm Scrapy crawler

0 引 言

洋葱路由(The onion route,Tor)通过采用不定数量节点、不定路由建立通信链路,并且在通信过程对通信数据进行层层加密,使得每一个洋葱节点只知道自己的前一个节点和后一个节点,从而保证数据通信过程的隐蔽、匿名和防溯源等特性[1]。由于 Tor 暗网的这些特性,里面充斥着许多违法犯罪行为。例如,2019 年 1 月 13 日,暗网中文论坛 Deepmix 上一名 ID 为 Itaiwanses 的卖家在出售近 1.3 GB 的中国航空客户信息数据[2],其中包含了用户的各种敏感信息。文献[3]指出,2018 年是数据泄露的灰色之年,而暗网(主要是 Tor 暗网)成为了贩卖泄露数据的主要渠道。

收稿日期:2020-04-01。国家重点研发计划项目"网络空间安全"重点专项(2016QY01W0202)。**龙军**,硕士生,主研领域:网络空间安全。**王轶骏**,讲师。**薛质**,教授。

于浩佳等[4]提出了一种通过 Scrapy 框架接入 Polipo 服务器,再结合 tor 浏览器进入到 Tor 暗网中的方法,对 Tor 暗网网页进行爬取。汤艳君等[5]利用 Selenium 工具及相关代理进入到 Tor 暗网,对 Tor 暗网网页进行爬取。但是以上方法爬取的站点都是简单的网页,没有验证码机制,不需要绕过人机交互,而目前的重要 Tor 暗网站点基本都采用了验证码机制。因此,上述方法都不适合对现有的重要 Tor 暗网站点进行自动化爬取。

研究表明,目前大型的 Tor 暗网站点都采用了验证码机制进行人机交互来抵抗分布式拒绝服务(DDoS)攻击和防止爬虫,特别是交易市场和黑色论坛[6]。针对目前重要 Tor 暗网站点的特性,本文通过对神经网络进行研究和实验来实现对相关验证码进行快速有效的识别,并且将其应用到 Tor 暗网站点的数据采集系统中去,实现自动化绕过验证码检验机制后爬取和存储站点的数据信息,从而能够有力地支撑暗网数据提炼、分析和挖掘的后续工作,对于监控暗网空间资源、获取威胁情报信息、感知网络安全态势具有重大意义。

对于一个特定的 Tor 暗网站点,验证码识别工作包括样本收集、样本标记和神经网络的设计与训练等工作,最后得到一个有效的模型。但是鉴于 Tor 暗网站点的变化是不定的,它的验证码样式和存在状态是随时可能发生变化,所以每一个 Tor 暗网站点的验证码识别工作花费的时间不能太长。针对于 Tor 暗网站点的这个特性,本文的最主要研究内容为设计实现一个能够使用少量验证码样本来训练的神经网络模型,并且该模型训练完成后能够快速有效地对验证码进行识别。设计实现 Tor 暗网站点的数据采集系统来实时、准确地采集相应站点的数据信息是另一项研究内容。最后将验证码快速识别模型应用到 Tor 暗网站点的数据采集系统,实现一套自动化的 Tor 暗网站点数据采集系统。

1 神经网络模型

实际调查表明,基本所有的重要 Tor 暗网站点所采取的验证码机制是由数字或者英文字母或者两者混合组成的文本验证码。程舟航[7]提出了一种端到端的文本验证码识别方法,结合 CNN、RNN 和 Attention 机制设计实现一个神经网络模型,通过使用大量标记好的样本进行训练得到了一个识别准确率可观的验证码识别模型。文献[8]提出了一种基于生成式对抗网络(GAN)的验证码识别模型,通过使用一定量的样本训练一个能够生成与样本非常相似的验证码的 GAN 模型,然后通过 GAN 模型自动生成大量样本来训练一个 CNN 网络,再用一定量的原始验证码进一步训练来修正这个 CNN 网络,最后能够得到一个识别准确率可观的验证码识别模型。但是上述第一种研究方法需要大量的样本和一定的机器资源,第二种方法需要一定量的样本、足够的机器资源及多次的实验,因此都需要花费大量的时间,不适合 Tor 暗网站点数据采集中的验证码识别工作。

结合文献阅读和相关实验,本文提出了一种结合 CNN 网络、门控循环单元(GRU)网络及 ctc(Connectionist Temporal Classification)loss 的神经网络模型,实验结果表明该模型能够在使用少量验证码样本进行训练的情况下达到可观的识别准确率。该网络模型的结构如图 1 所示。

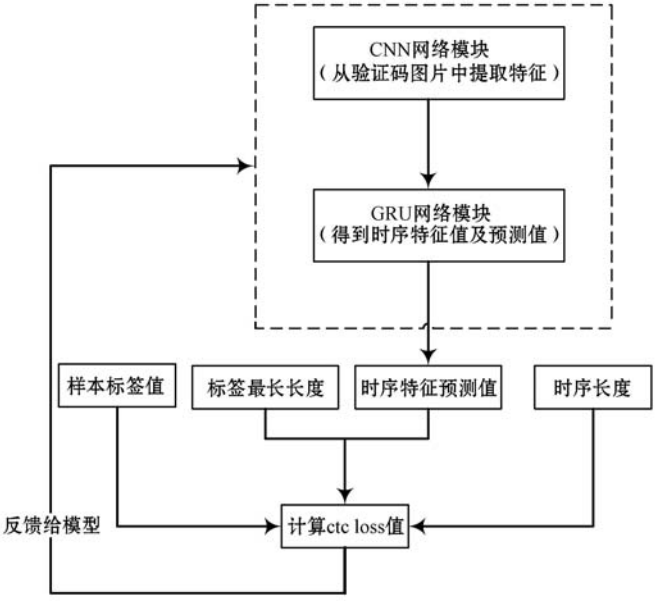


图 1 网络模型整体结构图

1.1 CNN 网络模块

在整个神经网络模型中,CNN 网络模块的作用是从验证码图片中提取特征,将每一幅验证码图片转化为相应维度的特征值。

本文在设计 CNN 网络模块时借鉴了 VGG[9]分类模型,整个 CNN 网络模块由五层子模块串联而成,每一层子模块包含了两层卷积层和一层池化层,卷积层中包括了卷积操作、批量正则化操作,并且使用了 ReLU 函数作为激活函数;池化层的参数需要根据验证码图片的宽度来设计,要使得经过 CNN 网络模块提取的特征再经过 GRU 网络模块后得到的时序特征长度不

能太长。每层子模块的网络结构如图 2 所示。

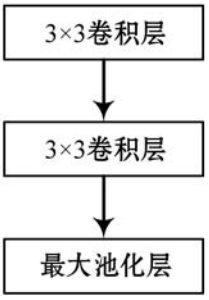


图 2　CNN 网络模块的子模块结构图

1.2　GRU 网络模块

GRU 网络模块首先将 CNN 网络模块提取的特征转化为时序特征,再对得到的时序特征进行预测,计算出一个时序特征预测值,这个值的长度要大于样本标签值的最长长度。

本文设计的 GRU 网络模块由置换层、时序平滑层、两层 GRU 层及 Dense 层组成,其中置换层可以改变特征维度的顺序,本文的 GRU 网络模块中通过置换层将第一维的特征与第二维的特征交换顺序;时序平滑层将特征转化为时序特征;GRU 层是一种特殊的 RNN 层,在原本的 RNN 层的基础上引入重置门和更新门,从而能够有效避免训练过程中参数消失和参数爆炸的情况;Dense 层对经过 GRU 层得到的特征进行分类计算出一个长度为时序特征长度的预测值,这个长度大于样本标签值的最长长度。GRU 网络模块的结构如图 3 所示。

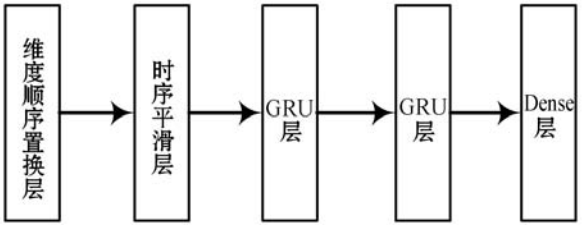


图 3　GRU 网络模块结构图

1.3　损失函数 ctc loss

假设样本标签的取值空间为 $Z=\{z_1,z_2,\cdots,z_n\}$,样本标签的最大长度为 m。经过上述的 CNN 网络模块和 GRU 网络模块(Dense 层之前),一幅验证码图片就变成了 $X=[x_1,x_2,\cdots,x_T]$ 的时序特征,其中 T 表示时序长度,$x_i(0<i<T+1)$ 表示 i 时刻的特征值,特别地,这里 T 要大于 m。文献[10]指出,ctc loss 是一种不需要输入值和输出值对齐就能够计算出一个可导损失值的损失函数,适用于本文的神经网络模型。

ctc loss 在原来的样本标签取值空间中加入了空符号,即变为 $Z=\{z_1,z_2,\cdots,z_n,\_\}$,其中_表示空符号。因为 ctc loss 的输入值长度大于输出值,所以会有多个输入值映射到一个输出值,映射的规则为连续的重复符号只保留一个,空符号去掉,这里称每一个映射为一条路径。对于 t 时刻,预测值为 $a_t(a_t\in Z)$ 的概率为 $P(a_t\mid x_t)$,所以时序特征的预测值为 $A=[a_1,a_2,\cdots,a_T]$ 的概率,如式(1)所示。

$$P(A\mid X)=\prod_{t=1}^{T}P(a_t\mid x_t) \tag{1}$$

令 A_y 表示所有能够映射到样本标签值的时序特征预测值的集合,则预测后映射能够得到正确的样本标签 Y 的概率值如式(2)所示。

$$P(Y\mid X)=\sum_{A\in A_y}\prod_{t=1}^{T}P(a_t\mid x_t) \tag{2}$$

训练过程中 ctc loss 的损失函数则为式(2)所示的概率值的负对数。但是随着时序长度 T 的增加,能够映射到样本标签值的路径个数呈指数增长,ctc loss 在计算过程中没有使用穷举法计算所有路径,而是借鉴了隐马尔可夫模型(HMM)中的前向后向动态规划算法[11],动态地计算式(2)中的值。假设样本标签值为 AUUW,计算时首先在这个标签的前后及两两字符之间插入空字符变为_A_U_U_W_,令 $r_t(s)$ 表示 t 时刻预测的符号为扩展序列(_A_U_U_W_)的第 s 个符号,且到 t 时刻为止预测路径是正确路径的前缀的概率,则可以动态地计算 $r_t(s)$。如图 4 所示,通过前向概率动态计算 $r_t(s)$ 分为三种情况(通过后向概率计算类似)。

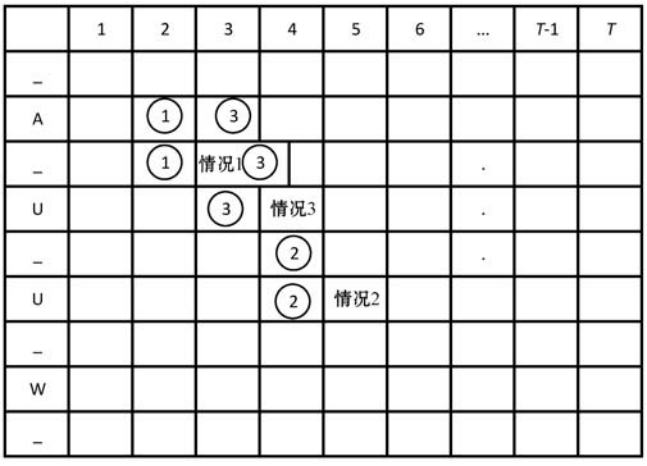


图 4　ctc loss 动态计算图

第一种情况,第 s 个符号为空白符号,如图 4 中的情况 1 所示,此时 $t=3,s=3$,则 $t=2$ 时只能是图 4 中圆圈 1 所标记的两种可能,所以 $r_3(3)=(r_2(3)+r_2(2))\times P\_(3)$,其中 $P\_(3)$ 表示 $t=3$ 时预测为空白符号的概率。

第二种情况,第 s 个符号等于第 $s-2$ 个符号,如图 4 中的情况 2 所示,此时 $t=5,s=6$,则 $t=4$ 时只能是图

4 中圆圈 2 所标记的两种可能，所以 $r_5(6)=(r_4(6)+r_4(5))\times P_U(5)$，其中 $P_U(5)$ 表示 $t=5$ 时刻预测为符号 U 的概率。

第三种情况为其他情况，即不属于上述两种情况，如图 4 中情况 3 所示，此时 $t=4$，$s=4$，则 $t=3$ 时只有图 4 中圆圈 3 标记的三种可能，所以可得 $r_4(4)=(r_3(4)+r_3(3)+r_3(2))\times P_U(4)$，其中 $P_U(4)$ 表示 $t=4$ 时刻预测为符号 U 的概率。

所以可以利用动态规划算法计算 $r_t(s)$，其中 seq 表示扩展后的序列（示例为_A_U_U_W_），seq(s)表示 seq 序列中的第 s 个字符。

（1）初始条件 $r_1(1)=P_{\_}(1)$，$r_1(2)=P_{\mathrm{seq}(2)}(1)$，$r_1(s)=0$，其中 $s>2$。

（2）seq(s)为空字符或者 seq(s)=seq($s-2$)，如式(3)所示。

$$r_t(s)=(r_{t-1}(s)+r_{t-1}(s-1))\times P_{\mathrm{seq}(s)}(t) \tag{3}$$

（3）其他情况如式(4)所示。

$$r_t(s)=(r_{t-1}(s)+r_{t-1}(s-1)+r_{t-1}(s-2))\times P_{\mathrm{seq}(s)}(t) \tag{4}$$

则式(2)中的 $P(Y\mid X)=r_T(L)$，其中 L 表示 seq 的长度。

2 测 试

对于上述的神经网络模型，本文在实际过程中对 18 个重要 Tor 暗网站点进行测试，其中大部分为售卖违法商品的 Tor 暗网交易市场，还有几个 Tor 暗网黑色论坛。所测试的 Tor 暗网站点有部分目前已经被关掉了，这也说明了自动化的 Tor 暗网数据采集工作中快速识别验证码的重要性。

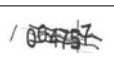
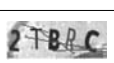
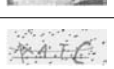
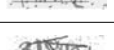
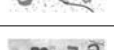
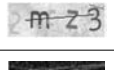
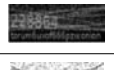
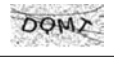
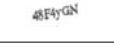
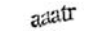
2.1 测试结果

本文设计的所有验证码识别模型在训练过程中所使用的机器配置为 Intel Core i5 2.3 GHz 四核，8 GB 内存，macOS Mojave 系统。

测试过程中，对所有测试了的 Tor 暗网站点收集的样本量均为 1 000 到 4 000 之间，所收集的验证码样本数量与该 Tor 暗网站点的验证码复杂性有关。在模型训练之前，把对每一个 Tor 暗网站点收集的验证码样本分成三部分，一部分作为训练集，一部分作为训练过程的验证集，一部分作为测试训练完成后的模型的试验集来计算该模型的识别准确率。

所有进行测试的 Tor 暗网站点的模型测试结果及相关测试信息如表 1 所示。

表 1 Tor 暗网站点的模型测试结果

网站名称	验证码	训练集数	验证集数	试验集数	试验集准确率/%	三次登录成功概率/%
deepmix		1 500	500	500	92.4	99.9
apollon		3 000	500	500	78.6	98.9
empire		1 500	400	150	72.6	97.8
darkbay		3 200	500	269	62.5	94.5
square		2 200	500	300	88.3	99.8
darkmarket		1 500	400	100	94.0	99.9
dreamalt		1 200	200	100	75.0	98.4
agartha		2 200	500	300	93.3	99.9
avaris		2 250	600	150	83.3	99.5
avior		1 300	500	200	81.0	99.3
torum		800	150	150	80.0	99.2
grey		1 200	200	100	80.0	99.2
genesis		800	150	50	92.2	99.9
berlusconi		1 100	300	200	88.0	99.8
cryptonia		1 500	500	200	98.0	99.9
nightmare		2 200	500	300	98.0	99.9
wallstreet		1 300	400	300	73.0	99.5
yellowbrick		1 200	200	100	99.0	99.9

2.2 结果分析

如表 1 所示，本文所设计的神经网络模型在少量验证码样本的训练下能够获得较高的识别准确率，全部站点都在 60% 以上，大部分站点在 80% 以上，甚至有的达到了 90% 以上。实际上，在 Tor 暗网的数据采集过程中，数据采集插件是可以进行多次的验证码识别和登录尝试，而其中只要成功一次即可。因此本文也针对每个重要 Tor 暗网站点，分别统计了在三次尝试以内至少能够成功一次的概率，结果表明绝大部分站点都在 98% 以上，而若干验证码样式比较复杂的 Tor 暗网站点也均在 90% 以上。因此，测试结果表明，将本文所提出的神经网络模型应用在重要 Tor 暗网站点的数据采集工作中，就能够进行自动的数据采集，而不是像以前需要人为干预。由于目前业界尚没有和本文类似的研究工作公开发表，基本上都还只是针对无

验证码机制的简单 Tor 暗网页面进行的数据采集,因此本文没有也无须进行相关的功能和性能比较。这同时也表明本文在暗网空间资源监测领域获得了突破性的进展,具有较重大的意义。

该神经网络模型通过 CNN 网络模块和 GRU 网络模型得到一个比样本标签值更长的时序特征预测值,该预测值中能够包含空白符号,这相当于在特征层面对原有的验证码进行了切割,从而对每一个切割的部分进行识别,再通过 ctc loss 计算整体预测值的损失函数反馈给模型进行训练,这是该神经网络模型在少量验证码样本的训练下就能够得到一个识别准确率较高的识别模型的重要原因。

3　Tor 暗网数据采集系统

Tor 暗网数据采集系统主要由三部分组成,分别是 Tor 内核传输模块、数据采集模块和数据存储模块,其中数据采集模块是整个系统的核心模块,结合使用了本文实现的神经网络模型来进行自动登录采集数据,整个系统的结构如图 5 所示。

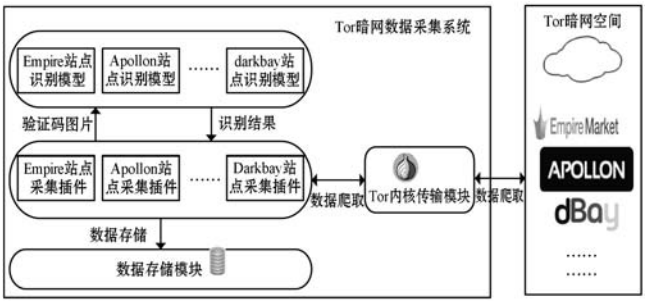


图 5　Tor 暗网数据采集系统结构图

3.1　Tor 内核传输模块

Tor 内核传输模块使用 Tor 官方提供的 tor 软件,该软件可以提供一个能够进入到 Tor 暗网空间的 socks 代理。但是本文设计的数据采集插件对于 socks 代理的支持不稳定,因此本文使用了 polipo 服务器将 socks 代理转化为 https 代理,更稳定地支持数据采集插件的使用。本文参考文献[12]中 Tor 暗网空间资源探测的工作中的 Tor 提速手段,修改 tor 软件配置文件中的节点为平均节点速度高的国家的节点,提高数据采集模块的采集速度。

3.2　数据采集模块

数据采集模块将数据采集插件与本文实现的验证码快速识别模型结合起来,从而解决数据采集插件遇到验证码无法自动识别后进行登录的技术难题,实现自动化的数据采集。

3.2.1　数据采集插件

数据采集插件使用 Scrapy 爬虫框架设计实现,实现采集插件中最关键的两点是抵御相应 Tor 暗网站点的反爬措施及数据的去重增量爬取。本文通过控制数据爬取的速率与频率来抵御相应 Tor 暗网网站的反爬措施,并且设计为被反爬措施检测到进行人机交互挑战后数据采集插件从之前停止的地方继续重新爬取。对于去重增量爬取,数据量大的 Tor 暗网站点采用 Bloomfilter 去重,数据量较小的 Tor 暗网站点则直接在数据库中查询进行去重。

文献[13]指出,Bloomfilter 是一种使用很长的比特串记录目标是否出现过的技术,其中比特位为 1 表示记录为出现过,0 表示记录为未出现。最初的比特串全为 0,判断过程中使用多个 Hash 函数将目标内容映射为多个数字用来表示在比特串中的索引,如果这些索引处全为 1 则表示该目标出现过,否则目标未出现过则处理目标并且将这些索引处全置为 1。因此,使用 Bloomfilter 技术在爬取时进行去重能够极大地减少内存的使用,节省资源。

3.2.2　验证码识别模型应用

在数据采集模块中,本文通过使用 Python 的 Flask 框架将所有训练好的验证码识别模型整合成一个 Web 系统,为每一个 Tor 暗网站点提供一个特定的 Web API 来自动识别验证码并且返回结果。爬取过程中识别相应验证码然后进行自动登录的流程如图 6 所示。

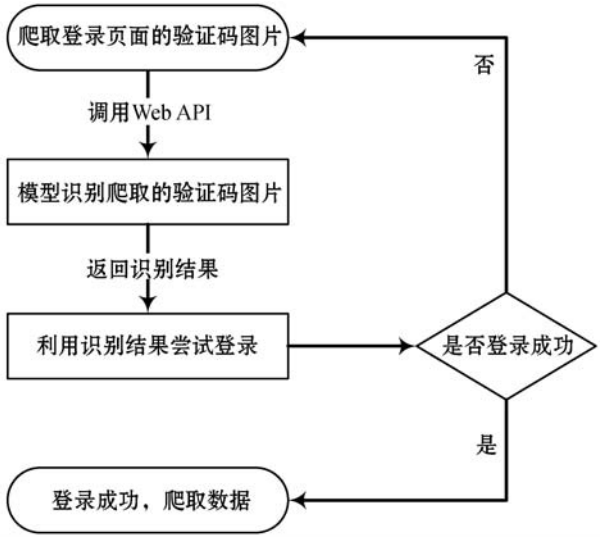


图 6　Tor 暗网站点自动登录流程图

当数据采集插件遇到登录页面时,首先将登录页面的验证码图片爬取下来,通过相应的 Web API 调用相应模型识别该验证码图片并且将识别结果返回给数据采集插件,插件利用识别结果尝试登录,再根据登录后的页面判断是否登录成功,如果登录成功则进一步爬取数据,登录失败则重新爬取登录页面的验证码。

重复进行上述的工作直至登录成功。在实际数据采集工作中,结果表明所有采集的重要 Tor 暗网站点基本尝试三次以内的登录流程后就能够登录成功。另外,测试结果表明登录过程调用 Web API 识别验证码每次所消耗的时间都在 5 秒以内。

3.3 数据存储模块

本文使用 MongoDB 作为数据存储的数据库,MongoDB 数据库存储数据时不会固定于一种数据格式,因此能够使得数据爬取过程中更具有伸缩性。另外,本文通过搭建一个 MongoDB 集群而不是使用单一的 MongoDB 节点进行数据存储,MongoDB 集群由多个 MongoDB 节点组成,其中一个为主节点,剩余的节点全为从节点,存储过程中先将数据存储在主节点然后备份到各个从节点,当主节点出现不可预估的故障时,MongoDB 集群剩余的从节点会通过选举方式选出一个节点替代原先的主节点继续数据存储工作,能够提高数据存储模块的鲁棒性。

3.4 系统运行结果与分析

Tor 暗网数据采集系统使用机器的配置为 Intel Xeon Bronze 1.7 GHz 12 核,64 GB 内存,4 TB 硬盘,Centos7 系统,带宽 100 MB。

到目前为止,该系统已经运行长达近一年,实际结果证明该系统能够自动、准确、快速地采集重要 Tor 暗网站点的数据信息。爬取的 Tor 暗网重要站点分为 Tor 暗网交易市场和 Tor 暗网黑色论坛,系统采集的交易市场实际数据如表 2 所示,采集的黑色论坛的实际数据如表 3 所示。

表 2 暗网交易市场数据采集结果表

市场名称	站点状态	商品数量	卖家数量
empire	开放且更新	123 047	4 054
agartha	开放且更新	90 229	490
darkbay	开放且更新	77 370	394
darkmarket	开放且更新	57 544	1 193
dreamalt	开放且更新	25 400	308
avairs	开放且更新	11 136	374
yellowbrick	开放且更新	2 385	503
square	开放且更新	4 691	167
apollon	关闭	71 449	1 915
berlusconi	关闭	145 306	4 282
nightmare	关闭	86 868	1 965
wallstreet	关闭	29 551	2 966

续表 2

市场名称	站点状态	商品数量	卖家数量
cryptonia	关闭	25 807	1 093
grey	关闭	15 239	300
genesis	关闭	8 450	368

表 3 Tor 暗网黑色论坛数据采集结果表

论坛名称	站点状态	发表数量	用户数量
torum	开放且更新	52 944	32 426
deepmix	开放且更新	9 257	无用户信息
deepbbs	关闭	224	无用户信息

本文的研究重点放在 Tor 暗网交易市场的数据采集上,因此表 2 中的市场数据比表 3 的论坛数据要大。此外,表 2 市场中的商品数量这一数据比这些市场上实时发布的数据量要大,这是因为 Tor 暗网交易市场中的商品存在更新和下架,系统采集的数据长期积累使得该数据量大于市场实时发布的数据量。系统实际运行的结果充分证明本文所设计的 Tor 暗网数据采集系统能够自动、准确、快速地采集 Tor 暗网重要站点的数据信息。

4 结 语

随着数字货币的出现和快速发展,暗网上的各种违法行为愈来愈多,Tor 暗网上违法泄露数据交易事件也逐年增多。为了及时采集 Tor 暗网站点的数据信息,从而进一步监控暗网空间资源,本文提出了数据采集模块通过 Tor 内核传输模块进入到 Tor 暗网空间实时采集数据信息的方法。但是随着 Tor 暗网站点的发展,特别是为了抵御 DDoS 攻击和防止爬虫,重要 Tor 暗网站点都使用了验证码机制来进行人机交互。学者们目前普遍使用 CNN、RNN 网络来对验证码进行端到端的识别,但是这种方法需要大量标记好的验证码样本来训练模型,这对于 Tor 暗网站点地数据采集工作是不合适的。Tor 暗网的变化是频繁和不定的,需要能够快速简便地得到一个 Tor 暗网站点的验证码识别模型,同时需要的训练样本要尽可能少。因此本文设计实现了一个神经网络模型,通过 CNN 网络模块提取验证码图片的特征,再通过 GRU 网络模块将提取到的特征转化为时序特征并且得到一个时序特征预测值,网络模型使用 ctc loss 对时序特征预测值和样本标签值进行比较计算得到损失函数反馈给模型,不断训练得到一个有效的验证码识别模型。这个神经网络模型在

使用少量验证码样本训练的情况下能够达到一个可观的验证码识别准确率,这是因为通过利用 GRU 网络得到时序特征,并且在训练过程中使用 ctc loss 进行计算,相当于在特征层面对验证码进行了切割,每一个时刻的特征像相当于切割的一部分,从而相当于对每个单字符进行识别,而单字符的识别所需要的样本是少量的。最后将该神经网络模型应用到 Tor 暗网站点的数据采集系统中,结果表明该系统能够自动、准确、快速地采集 Tor 暗网站点的数据信息,对于监控暗网空间资源、获取威胁情报信息、感知网络安全态势具有重大意义。

参 考 文 献

[1] torproject. Tor overview[EB/OL]. (2019)[2020 - 03 - 20]. https://www.torproject.org/about/overview.html.en.

[2] 安全讯息平台 NOSEC. 暗网出现大量疑似中国大陆航空客户数据售卖[EB/OL]. (2019 - 02 - 11)[2020 - 03 - 20]. https://nosec.org/home/detail/2234.html.

[3] 腾讯安全. 信息泄露:2018 企业信息安全头号威胁回顾[EB/OL]. (2019/01/07)[2020 - 02 - 20]. https://s.tencent.com/research/report/626.html.

[4] 于浩佳,陈波,刘蓉. 匿名网站信息爬取技术研究[J]. 信息安全研究,2017,3(10):922 - 931.

[5] 汤艳君,安俊霖. 基于 Tor 的暗网数据爬虫设计与实现[J]. 信息安全研究,2019,5(9):798 - 804.

[6] 曹哲超,王轶骏,薛质. 基于页面标签和文本特征的暗网重要站点识别[J]. 通信技术,2019,52(12):3021 - 3026.

[7] 程舟航. 端到端文本验证码识别[D]. 西安:西安电子科技大学,2019.

[8] Ye G X, Tang Z Y, Fang D Y, et al. Yet another text captcha solver: A generative adversarial network based approach[C]//Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security. ACM,2018: 332 - 348.

[9] Simonyan K, Zisserman A. Very deep convolutional networks for large scale image recognition[C]//3rd International Conference on Learning Representations, 2015.

[10] Hannun. Sequence Modeling with CTC[EB/OL]. (2017)[2020 - 03 - 20]. https://distill.pub/2017/ctc/.

[11] Rabiner L R. A tutorial on hidden Markov models and selected applications in speech recognition[J]. Proceedings of the IEEE,1989,77(2):257 - 286.

[12] 杨溢,郭晗,王轶骏,等. 基于 Tor 的暗网空间资源探测[J]. 通信技术,2017,50(10):2304 - 2309.

[13] Cao P. Bloom Filters—The math[EB/OL]. (2018 - 05 - 27)[2020 - 32 - 20]. http://pages.cs.wisc.edu/~cao/papers/summary-cache/node8.html.

(上接第 268 页)

[12] Frigui H, Nasraoui O. Unsupervised learning of prototypes and attribute weights[J]. Pattern Recognition,2004,37(3): 567 - 581.

[13] Jing L, Ng M K, Xu J, et al. Subspace clustering of text documents with feature weighting k-means algorithm[C]// Pacific-Asia Conference on Knowledge Discovery and Data Mining, 2005: 802 - 812.

[14] Gan G, Wu J. A convergence theorem for the fuzzy subspace clustering(FSC) algorithm[J]. Pattern Recognition,2008, 41(6):1939 - 1947.

[15] Jing L, Ng M K, Huang J Z. An entropy weighting k-means algorithm for subspace clustering of high-dimensional sparse data[J]. IEEE Transactions on Knowledge and Data Engineering, 2007,19(8):1026 - 1041.

[16] Domeniconi C, Gunopulos D, Ma S, et al. Locally adaptive metrics for clustering high dimensional data[J]. Data Mining and Knowledge Discovery, 2007,14(1):63 - 97.

[17] Deng Z, Choi K S, Chung F L, et al. Enhanced soft subspace clustering integrating within-cluster and between-cluster information[J]. Pattern Recognition,2010,43(3):767 - 781.

[18] 封文清,巩敦卫. 基于在线感知 Pareto 前沿划分目标空间的多目标进化优化[J]. 自动化学报,2020,46(8):1628 - 1643.

[19] Cheng J, Yen G, Zhang G. A grid-based adaptive multi-objective differential evolution algorithm[J]. Information Sciences,2016,367:890 - 908.

[20] 张曦,赵嘉,李沛武,等. 改进萤火虫优化的软子空间聚类算法[J]. 南昌工程学院学报,2018,37(4):61 - 67.

[21] Monti S, Tamayo P, Mesirov J, et al. Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data[J]. Machine Learning, 2003, 52(1/2):91 - 118.

[22] Hoshida Y, Brunet J P, Tamayo P, et al. Subclass mapping: Identifying common subtypes in independent disease data sets[J]. Plos One, 2007, 2(11):e1195.

[23] Rand W M. Objective criteria for the evaluation of clustering methods[J]. Journal of the American Statistical Association, 1971,66(336):846 - 850.

[24] Strehl A, Ghosh J. Cluster ensembles—knowledge reuse framework for combining multiple partitions[J]. Journal of Machine Learning Research,2002,3(3):583 - 617.

[25] Yang A, Li W, Yang X. Short-term electricity load forecasting based on feature selection and least squares support vector machines[J]. Knowledge-Based Systems,2019,163:159 - 173.

[26] 陈鸿琳. 基于相似日和智能算法的短期负荷组合预测[D]. 长沙:湖南大学,2016.