·安全技术·　　文章编号：1000-3428(2019)09-0161-08　　文献标志码：A　　中图分类号：TP391

隐马尔可夫模型在恶意域名检测中的应用

白玲玲[1]，宁振虎[1]，薛　菲[2]，杨永丽[1]

（1. 北京工业大学 信息学部，北京 100124；2. 北京物资学院 信息学院，北京 101149）

摘　要： 提出一种基于隐马尔可夫模型（HMM）的恶意域名检测方法。分析善恶域名在 DNS 通信中的各类特征，利用 Spark 大数据处理平台的高效计算能力对属性特征进行统计，在此基础上，通过 HMM 中的 Baum-Welch 算法和 Viterbi 算法对恶意域名进行准确分类。实验结果表明，与随机森林模型相比，HMM 对恶意域名分类的准确率与召回率均较高。

关键词： 恶意域名；隐马尔可夫模型；Baum-Welch 算法；Viterbi 算法；Spark 大数据处理平台

中文引用格式： 白玲玲，宁振虎，薛菲，等. 隐马尔可夫模型在恶意域名检测中的应用[J]. 计算机工程，2019，45(9)：161-168.

英文引用格式： BAI Lingling，NING Zhenhu，XUE Fei，et al. Application of hidden Markov model in malicious domain name detection[J]. Computer Engineering，2019，45(9)：161-168.

Application of Hidden Markov Model in Malicious Domain Name Detection

BAI Lingling[1]，NING Zhenhu[1]，XUE Fei[2]，YANG Yongli[1]

（1. Faculty of Information Technology，Beijing University of Technology，Beijing 100124，China；
2. School of Information，Beijing Wuzi University，Beijing 101149，China）

【Abstract】 A malicious domain name detection method based on Hidden Markov Model（HMM） is proposed. The characteristics of good and evil domain name in DNS communication are analyzed，and the attribute characteristics are counted by using the efficient computing power of Spark big data processing platform. On this basis，malicious domain name are accurately classified by Baum-Welch algorithm and Viterbi algorithm in HMM. Experimental results show that compared with the Random Forest（RF） model，the accuracy and recall rate of HMM for malicious domain name classification are both higher.

【Key words】 malicious domain name；Hidden Markov Model（HMM）；Baum-Welch algorithm；Viterbi algorithm；Spark big data processing platform

DOI： 10.19678/j.issn.1000-3428.0051486

0　概述

当前，以信息技术为代表的新一轮科技与产业革命正在兴起，随着互联网模式的不断创新以及线上线下服务融合的快速发展，截止 2017 年 12 月，我国网民规模达 7.72 亿，普及率为 55.8%，并继续保持平稳增长趋势。与此同时，以网络钓鱼和僵尸网络为代表的恶意域名攻击以灵活多变的形式不断出现，中国互联网络信息中心（CNNIC）对各行业域名安全状况进行分析，结果表明，我国 80% 以上的域名解析服务存在安全风险。随着全球域名体系注册量、查询量以及系统部署规模的持续增长，针对域名系统的 DDOS 攻击规模和攻击技术复杂度也在显著提升[1]。因此，对域名安全检测方面进行研究成为信息安全领域广泛关注的热点之一。

近年来，已有较多学者通过机器学习方法对恶意域名进行有效检测，主要包括支持向量机（Support Vector Machine，SVM）、随机森林（Random Forest，RF）和聚类等方法[2]。机器学习可通过分析善恶域名在网络行为特性与自身特性方面存在的差异，利用各种高效算法来实现域名分类[3-5]。但是，随着当今信息量和访问量的快速增长，域名在 DNS 通信中的日志记录量也在持续提升，需要通过大数据的高效统计分析技术来提升海量通信记录的分析速度，这将导致传统检测算法在数据计算与响应速度 2 个方面受到极大限制[6]。此外，在利用机器学习检测恶意域名时，分类模型大都选择 RF 分类器，尽管 RF 分类器不易发生过拟合现象，且在每次划分时只考

基金项目： 北京市博士后工作经费项目（2017-22-030）；CCF-启明星辰“鸿雁”科研计划（CCF-VenustechRP2017008）。

作者简介： 白玲玲（1991—），女，硕士研究生，主研方向为信息安全；宁振虎（通信作者）、薛　菲，讲师、博士；杨永丽，硕士研究生。

收稿日期： 2018-05-08　　**修回日期：** 2018-07-28　　**E-mail：** nzh41034@163.com

虑很少的属性,速度比 Bagging 和 Boosting 分类器更快,但其难以处理某些噪声较大的分类和回归问题。同时,根据实际需要对多个不同级别的属性进行划分时,会使 RF 模型的建模结果产生较大的误差,导致属性权值不具有可信度[7]。

恶意域名通常通过域名解析系统来实现攻击,因此,当黑客们通过恶意域名进行破坏活动时,相应的善恶域名数据及其特征行为模式会被记录到 DNS 服务器的日志中。本文通过分析 DNS 服务器中海量的域名日志解析记录及其相应特征,利用隐马尔可夫模型(Hidden Markov Model,HMM)及 Spark 大数据处理平台[8-10],结合域名的自身特性、时间性(包括 TTL 平均值、域名生存期等)、对应 IP 的特征以及相关域名集,得到恶意域名特定的行为方式,在此基础上,对其进行检测与分类。

1 相关工作

HMM 在语音识别、机器学习等领域得到广泛应用[11],其是马尔可夫模型的延伸与扩展。HMM 在任一时刻的状态不可见,观察者无法通过一个状态序列推测转移概率等相关参数,但可通过 HMM 在每个时刻的观测值来对隐含状态进行预测,且该时刻的观测值仅与其隐含状态有关[12-14]。

1.1 隐马尔可夫模型概述

在通常情况下,HMM 的形式可表示为[15]:

$\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$

各元素定义为:

1)π 表示初始状态的概率分布,$\pi=\{\pi_1,\pi_2,\cdots,\pi_N\}$,$\pi_i=P(Q_1=S_i)(1\leqslant i\leqslant N)$ 表示模型在初始状态 Q_1 时取 S_i 状态的概率。

2)$\boldsymbol{A}$ 表示状态之间的转移矩阵,$\boldsymbol{A}=[a_{ij}]_{N\times N}$,$a_{ij}=P(Q_{t+1}=S_j\mid Q_t=S_i)(1\leqslant i,j\leqslant N)$ 表示在时刻 t、状态 S_i 的条件下时刻 $t+1$ 转移到状态 S_j 的概率。

3)$\boldsymbol{B}$ 表示观测概率矩阵,$\boldsymbol{B}=[b_{jk}]_{M\times N}$,$b_{jk}=P(O_t=V_k\mid Q_t=S_j)(1\leqslant j\leqslant N,1\leqslant k\leqslant M)$ 表示在状态 S_j 出现时观测值 V_k 的概率。

在 HMM 中,初始概率和状态转移概率决定状态序列,观测概率矩阵决定模型的观测序列,因此,HMM 可简记为 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$。

1.2 隐马尔可夫模型中的基本问题

HMM 涉及 3 个基本问题[15]:

1)评估问题:已知观测序列 $O=\{O_1,O_2,\cdots,O_T\}$ 及模型 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$,计算 $P(O\mid\lambda)$,即给定一个模型的观测序列,计算某个特定的输出序列的概率。解决评估问题主要采用前向算法和后向算法。

(1)前向算法

输入 HMM 模型 λ,观测序列 O

输出 观测序列概率 $P(O\mid\lambda)$

1. 初始化

$\alpha_1(i)=\pi_i b_i(o_1),i=1,2,\cdots,N$

2. 对 $t=1,2,\cdots,T-1$,有:

$$\alpha_{t+1}(i)=\left[\sum_{j=1}^{N}\alpha_t(j)\alpha_{ji}\right]b_i(o_{t+1}),i=1,2,\cdots,N$$

3. 终止

$$P(O\mid\lambda)=\sum_{i=1}^{N}\alpha_T(i)$$

前向算法的步骤 1 初始化前向概率,包括初始时刻的状态 $i_1=q_i$ 和观测 o_1 的联合概率;步骤 2 是前向概率的递推公式,计算到 $t+1$ 时刻观测序列为 $o_1,o_2,\cdots,o_t,o_{t+1}$ 且在 $t+1$ 时刻模型处于状态 q_i 的前向概率;步骤 3 给出 $P(O\mid\lambda)$ 的计算公式。

(2)后向算法

输入 HMM 模型 λ,观测序列 O

输出 观测序列概率 $P(O\mid\lambda)$

1. $\beta_t(i)=1,i=1,2,\cdots,N$

2. 对 $t=T-1,T-2,\cdots,1$,有:

$$\beta_t(i)=\sum_{j=1}^{N}\alpha_{ij}b_j(o_{t+1})\beta_{t+1}(j),i=1,2,\cdots,N$$

3. $$P(O\mid\lambda)=\sum_{i=1}^{N}\pi_i b_i(o_1)\beta_1(i)$$

步骤 1 初始化后向概率,即对模型最终时刻的所有状态 q_i 规定 $\beta_T(i)=1$;步骤 2 对后向概率的计算公式进行递推,即在 t 时刻状态 q_i 条件下计算 $t+1$ 之后模型观测序列为 $o_{t+1},o_{t+2},\cdots,o_T$ 的后向概率 $\beta_t(i)$,只需计算 $t+1$ 时刻所有可能的 N 个状态 q_j 的转移概率和在此状态下的 o_{t+1} 的概率,再考虑状态 q_j 之后的观测序列的后向概率;步骤 3 求 $P(O\mid\lambda)$,其思路与步骤 2 一致,只是用初始概率 π_i 代替转移概率。

2)解码问题,即在给定观测序列 $O=\{O_1,O_2,\cdots,O_T\}$ 和模型 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$ 的情况下,选择一个对应的状态 $S=\{S_1,S_2,\cdots,S_N\}$,使得由 S 生成观测序列 O 的概率最大。

概率统计模型中最常用的标准是最大似然标准,即在给定模型和某个特定输出序列的情况下,选择概率最大的一个输出序列,此时常用算法为 Viterbi 算法,其是应用较广的动态规划算法[16-17],可利用动态规划的特性来求解相关概率的最大路径问题。其中,一条路径对应模型中的一个状态序列。Viterbi 算法的思路为:从 $t=1$ 时刻开始,不断向后递推,每次递推下一步时保留前一步所有选择的最大概率,在递推完成之后,利用回溯法从终点逐步倒退到起始点,即可得到所求模型的最佳状态路径。Viterbi 算法描述如下:

输入　模型参数 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$,观测序列 $O=(o_1,o_2,\cdots,o_T)$

输出　最优隐状态路径 $I^*=(i_1^*,i_2^*,\cdots,i_T^*)$

1. 初始化

$\delta_1(i)=\pi_i b_i(o_1),i=1,2,\cdots,N$

$\psi_1(i)=0,i=1,2,\cdots,N$

2. 递推

对 $t=2,3,\cdots,T$:

$\delta_t(i)=\max\limits_{1\leqslant j\leqslant N}[\delta_{t-1}(j)a_{ji}]b_i(o_t),i=1,2,\cdots,N$//记录从初//始时刻递推到时刻t时,概率最大路径(即最优路径)经过//的所有节点的联合概率

$\psi_t(i)=\arg\max\limits_{1\leqslant j\leqslant N}[\delta_{t-1}(j)a_{ji}],i=1,2,\cdots,N$//记录最终到//达的 t 时刻的隐状态

3. 终止

$P^*=\max\limits_{1\leqslant i\leqslant N}[\delta_T(i)]$

$i_T^*=\arg\max\limits_{1\leqslant j\leqslant N}[\delta_T(i)]$

4. 回溯最优路径

对 $t=T-1,T-2,\cdots,1$:

$i_t^*=\psi_{t+1}(i_{t+1}^*)$

求得最优路径 $I^*=(i_1^*,i_2^*,\cdots,i_T^*)$

3)学习问题。对给定的观测序列 $O=\{O_1,O_2,\cdots,O_T\}$,估计 HMM 模型的参数 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$,使得 $P(O|\lambda)$最大。该问题通过不断地输入观测序列,应用迭代法对模型参数进行调整与改进,继而使模型达到期望效果,常用算法为 Baum-Welch 算法。Baum-Welch 是典型无监督学习 HMM 的方法,其利用期望极大(Expectation Maximization,EM)算法思想[18-19],通过反复迭代达到局部极大值,最后得出 HMM 的估计参数 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$。

Baum-Welch 算法步骤如下:

(1)随机设置初始模型 M_0。

(2)基于 M_0 以及观测序列 O,训练新模型 M^*。

(3)若满足设定条件(如 $\|M^*-M_0\|<\varepsilon$),表明训练已达到预期效果,算法结束。

(4)否则,令 $M_0=M^*$,继续执行第 2 步。

给定模型 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$,生成观测序列 O 的概率如下:

$$P(O|\lambda)=\sum_S P(O|S,\lambda)P(S|\lambda)$$

Baum-Welch 算法的目标是找到使得观测序列 O 概率最大的模型:

$$\lambda^*=\arg\max_{\lambda} P(O|\lambda)$$

针对上述最大化问题,得到全局最优解比较困难,但可通过 EM 算法得到局部最优解。EM 算法步骤如下:

(1)确定由观测序列 O 和状态数据 S 所构成的对数似然函数:

$$\lg P(O,S|\lambda)$$

(2)E 步,求 Q 函数:

$$Q(\lambda,\lambda^*)=\sum_S P(S|O,\lambda)\lg P(O,S|\lambda^*)$$

(3)M 步,最大化 Q 函数以求解最新的模型参数:

$$\bar{\lambda}=\arg\max_{\lambda^*} Q(\lambda,\lambda^*)$$

先对 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$设置初值 $\lambda_0=(\pi_0,\boldsymbol{A}_0,\boldsymbol{B}_0)$,通过求解 $\max Q(\lambda_0,\lambda^*)$,得到参数新的估计值 λ_1,再利用 λ_1 求解 $\max Q(\lambda_1,\lambda^*)$,得到新估值 λ_2。反复迭代,直到所求参数值变化足够小。上述过程描述如下:

$$\lambda_0 \xrightarrow{\max\limits_{\lambda^*} Q(\lambda_0,\lambda^*)} \lambda_1 \xrightarrow{\max\limits_{\lambda^*} Q(\lambda_1,\lambda^*)} \lambda_2 \xrightarrow{\max\limits_{\lambda^*} Q(\lambda_2,\lambda^*)} \lambda_3\cdots$$

2　基于 HMM 的恶意域名检测

2.1　恶意域名检测问题描述

根据域名恶意程度的不同,从海量 DNS 日志中提取出的域名在不同时间序列中具有不同的属性特征,且这些域名状态不可观测。从海量 DNS 日志中提取出的相关域名属性,如域名的自身特性、时间性、对应 IP 的特征以及相关域名集等,可通过观察进行量化统计,且域名在不同时间序列中的可观测变量特征由域名所隐藏的恶意危害强度所决定,这种域名的恶意危害强度又由其所处的状态决定,因此,模型构建需要应用双重映射过程[20]。

用作域名识别的观测变量应该具备区别善恶域名的能力,即在域名检测过程中,针对不存在恶意域名和存在恶意域名的情况,这些观测变量的值将存在很大差异。本文通过分析 DNS 日志记录中域名的相关特征信息,提取出 5 个类别中的 12 个特征属性值,详细信息如表 1 所示。

表 1　域名特征说明

属性集	属性名
域名字符特征	数字占总长度的比例
	域名字符熵值
	域名长度
域名对应 IP 特征	域名对应 IP 的个数
	域名对应 IP 的差异性
	域名对应的国家个数
TTL 特征	TTL 的平均值
	TTL 的变化次数
	TTL 的标准差
NS 记录特征	域名服务器变化次数
	域名服务器平均变化时长
域名生存期特征	域名的生存周期

2.2 基于 HMM 的恶意域名检测问题描述

一种有效的恶意域名检测方法即对海量 DNS 日志的域名属性特征进行监控,获取该域名的自身特性、时间性、对应 IP 特征以及相关域名集等,分析这些海量的域名历史数据得出不同域名在特定时间序列中的 HMM 模型初始化参数,然后通过 HMM 模型中的 Baum-Welch 算法进行反复迭代并训练出最优的模型参数,在此基础上,通过 Viterbi 算法检测出相应属性特征下域名最可能的恶意隐藏状态行为。基于 HMM 的恶意域名检测系统总体框架如图 1 所示。

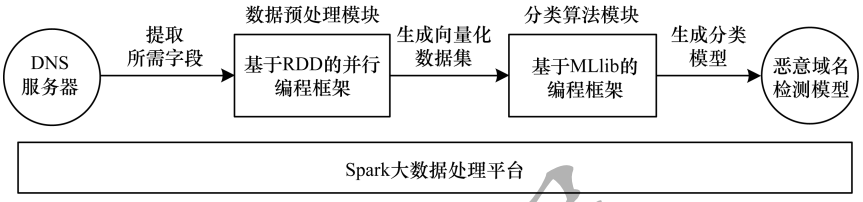

图 1 恶意域名检测系统总体架构

在域名检测系统中,构建 HMM 最重要的步骤是确定三元组 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$ 中各元素之间的映射关系,即模型中初始概率、转移概率和混淆矩阵最合适的数据组合。由于无法直接观测到域名在不同时间序列中的状态,因此本文将域名的状态作为 HMM 的隐状态,这些隐状态包括善意域名和恶意域名,根据恶意域名的不同类型又可划分为 N 个隐状态,观测变量的选取即从 DNS 日志中提取出 5 大类共 12 个属性特征。其中,初始概率、转移概率和混淆矩阵可通过分析 DNS 日志的历史记录并根据域名之间的实际分布情况进行参数初始化。

2.3 基于 HMM 的恶意域名识别系统

2.3.1 系统特征值提取

为实现本文系统,需从域名的自身特性、时间性、对应 IP 特征和相关域名集中提取出 12 个特征值。为加快提取速度,利用 Spark 大数据处理平台中 Transfromation 算子的懒加载机制来避免产生中间数据,在 Action 操作时才进行特征提取。特征数据提取过程如图 2 所示。

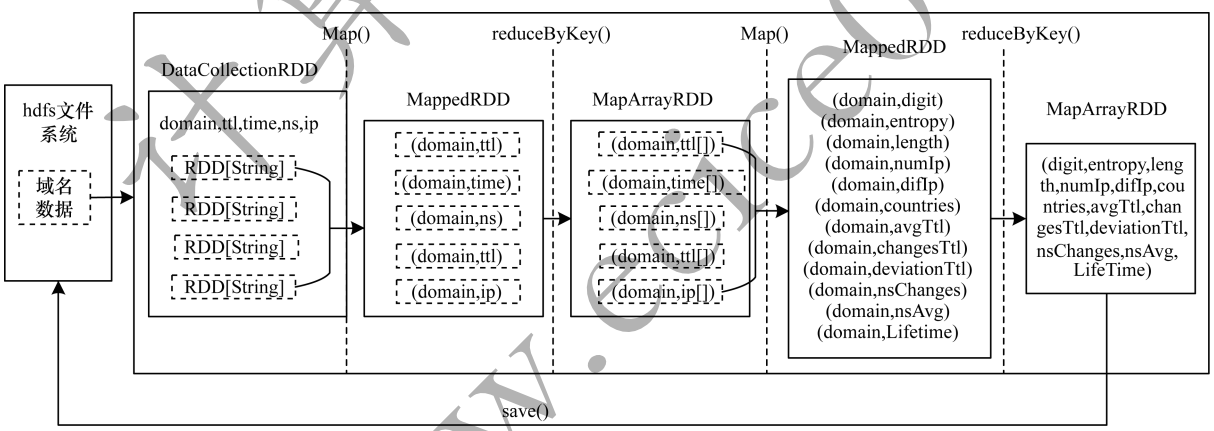

图 2 特征数据提取过程

首先,Spark 从 hdfs 文件系统中读取相关的域名数据,并在内存中将其表示为 DataCollectionRDD;然后,使用 Map 算子提取出域名与 TTL 值、IP 地址、访问时间、NS 记录的组合对,再通过 reduceByKey 算子将相同域名的 TTL 值、IP 地址、访问时间、NS 记录进行聚合,并根据域名计算出数据所占的百分比,从聚合到的 TTL 值中计算出 TTL 均值、TTL 变化数和 TTL 标准差 3 个特征值,从聚合到的访问时间中得到域名生存期,从聚合到的 IP 地址中计算出域名对应的 IP 个数、域名对应国家数、域名对应 IP 的差异性 3 个特征值,从聚合到的 NS 记录特征中计算出域名服务器变化次数、域名服务器平均变化时长;最终,得到 12 个特征值并形成样本集,存入 hdfs 文件系统中。

2.3.2 DNS 数据流实时获取

本文采用建立在 Spark 上的实时计算框架 Spark Streaming,从 DNS 日志中获取实时的数据流。Spark Streaming 的处理机制是将持续不断输入的数据流根据一定的时间间隔拆分成多个批处理作业[21],然后通过 Spark Engine 对这些数据进行处理。本文系统通过 Spark Streaming 从 DNS 获取实时数据流的主要步骤如下:

步骤 1 Spark Streaming 参数初始化。

步骤 2 从 hdfs 的数据源不断监听是否有新数据到达。

步骤 3　将接收的新数据以时间片为单位进行分批,利用 Spark Engine 处理这些数据。

步骤 4　将步骤 3 中经过各种 Spark 算子处理后的结果数据流 DStream 转化为对应的 RDD。

2.3.3　HMM 模型实现

HMM 模型的实现步骤具体如下:

步骤 1　对 HMM 的以下相关参数进行初始化:

1)状态 S。善恶域名在检测时间序列中进行访问操作时,域名的隐状态类型无法预知。因此,根据域名变换的多样性和灵活性,可通过试探法对域名模型的隐状态进行探测。在实验中,根据危害程度从弱到强将域名攻击暂分为 $S_1,S_2,\cdots,S_5$ 共 5 类,对应于 HMM 模型中的不同状态。

2)观测值 V。该值为海量 DNS 日志中相关域名的解析记录及其相应的特征行为模式。实验中使用的观测特征值共有 12 个,即 $O=\{O_1,O_2,\cdots,O_{12}\}$。

3)初始状态的概率分布 $\Pi=\{\pi_i\}$,$\pi_i=P(Q_1=S_i)$表示系统中所有域名 S_i 初始出现的概率,其计算如下:

$$\pi_i=\frac{\text{初始时不同状态域名 } S_i \text{ 出现的次数}}{\sum_{j=1}^{N}\text{初始时不同状态域名 } S_j \text{ 出现的次数}} \tag{1}$$

其中,$1\leqslant i\leqslant N$。

4)状态转移概率矩阵:

$$\boldsymbol{A}=[\alpha_{ij}]_{N\times N},\alpha_{ij}=P(Q_{t+1}=S_j\mid Q_t=S_i)(1\leqslant i,j\leqslant N) \tag{2}$$

其中,α_{ij}表示系统中所有不同状态域名在某一时刻 t 处于 S_i、在时刻 $t+1$ 转移到 S_j 的概率,此处 t 只表示域名状态出现的先后顺序。α_{ij}计算如下:

$$\alpha_{ij}=\frac{\text{从 } S_i \text{ 转移到 } S_j \text{ 的次数}}{\sum_{k=1}^{N}\text{从 } S_i \text{ 转移到 } S_k \text{ 的次数}} \tag{3}$$

其中,$1\leqslant i,j,k\leqslant N$。

5)观测值的概率分布矩阵:

$$\boldsymbol{B}=[b_{jk}]_{M\times N}$$

其中,$b_{jk}=P(O_t=V_k\mid Q_t=S_j)$ $(1\leqslant j\leqslant N,1\leqslant k\leqslant M)$表示不同状态域名 S_j 出现时域名解析记录的相应特征为 V_k 的概率。b_{jk}计算如下:

$$b_{jk}=\frac{\text{域名状态为 } S_j \text{ 时解析记录特征为 } V_k \text{ 的次数}}{\text{解析记录特征为 } V_k \text{ 的总次数}} \tag{4}$$

其中,$1\leqslant j\leqslant N,1\leqslant k\leqslant M$。

步骤 2　使用 Baum-Welch 算法,根据训练集中 T 时刻 DNS 服务器中域名解析记录的相应观测序列 $O=\{O_1,O_2,\cdots,O_T\}$、隐状态序列 $Q=\{Q_1,Q_2,\cdots,Q_T\}$,对 HMM 模型中的参数进行训练,使得 $P(O\mid\lambda)$ 最大。具体流程如下:

1)确定由观测序列 O 和隐状态数据 S 构成的对数似然函数:

$$\lg P(O,S\mid\lambda) \tag{5}$$

2)E 步。求 Q 函数:

$$Q(\lambda,\lambda^*)=\sum_S P(S\mid O,\lambda)\lg P(O,S\mid\lambda^*) \tag{6}$$

3)M 步。通过反复迭代使 Q 函数最大化,求解最优的模型参数:

$$\bar{\lambda}=\arg\max_{\lambda^*} Q(\lambda,\lambda^*) \tag{7}$$

$$Q(\lambda,\lambda^*)=\sum_S P(S\mid O,\lambda)\lg P(O,S\mid\lambda^*)=\sum_S\frac{P(O,S\mid\lambda)}{P(O\mid\lambda)}\lg P(O,S\mid\lambda^*)=\frac{1}{P(O\mid\lambda)}\sum_S P(O,S\mid\lambda)\lg P(O,S\mid\lambda^*)$$

对于给定的参数 λ,$P(O\mid\lambda)$是常量,故:

$$\max_{\lambda^*} Q(\lambda,\lambda^*)=\max_{\lambda^*}\sum_S P(O,S\mid\lambda)\lg P(O,S,\mid\lambda^*)$$

$$P(O,S\mid\lambda^*)=\hat{\pi}_{s_1}\hat{b}_{s_1}(o_1)\hat{\alpha}_{s_1s_2}\hat{b}_{s_2}(o_2)\cdots\hat{\alpha}_{s_{T-1}s_T}\hat{b}_{s_T}(o_T)$$

$$\begin{aligned}Q(\lambda,\lambda^*)&=\sum_S P(O,S\mid\lambda)\lg P(O,S\mid\lambda^*)=\\&\sum_S P(O,S\mid\lambda)\lg(\hat{\pi}_{s_1}\hat{b}_{s_1}(o_1)\hat{\alpha}_{s_1s_2}\hat{b}_{s_2}(o_2)\cdots\hat{\alpha}_{s_{T-1}s_T}\hat{b}_{s_T}(o_T))=\\&\sum_S P(O,S\mid\lambda)\lg\hat{\pi}_{s_1}+\sum_S P(O,S\mid\lambda)(\sum_{t=1}^{T-1}\lg\hat{\alpha}_{s_ts_{t+1}})+\sum_S P(O,S\mid\lambda)(\sum_{t=1}^{T}\lg\hat{b}_{s_t}(o_t))\end{aligned}$$

$$\max\sum_S P(O,S\mid\lambda)\lg\hat{\pi}_{s_1},\sum_{i=1}^{N}\hat{\pi}_i=1$$

$$\max\sum_S P(O,S\mid\lambda)(\sum_{t=1}^{T-1}\lg\hat{\alpha}_{s_ts_{t+1}}),\sum_{j=1}^{N}\hat{\alpha}_{ij}=1$$

$$\max\sum_S P(O,S\mid\lambda)(\sum_{t=1}^{T}\lg\hat{b}_{s_t}(o_t)),\sum_{k=1}^{N}\hat{b}_j(k)=1$$

迭代执行 $n=1,2,\cdots$,直到满足终止条件:

$$\pi_i^{n+1}=\frac{P(O,s_1=i\mid\lambda^n)}{P(O\mid\lambda^n)},i=1,2,\cdots,N$$

$$\alpha_{ij}^{n+1}=\frac{\sum_{t=1}^{T-1}P(O,s_t=i,s_{t+1}=j\mid\lambda^n)}{\sum_{t=1}^{T-1}P(O,s_t=i\mid\lambda^n)}$$

$$b_j(k)^{n+1}=\frac{\sum_{t=1}^{T}P(O,s_t=j\mid\lambda^n)\delta_{t,k}}{\sum_{t=1}^{T}P(O,s_t=j\mid\lambda^n)} \tag{8}$$

$$\lambda^{n+1}=(\pi^{n+1},\boldsymbol{A}^{n+1},\boldsymbol{B}^{n+1}) \tag{9}$$

输出最终模型:

$$\lambda^*=(\hat{\pi},\hat{\boldsymbol{A}},\hat{\boldsymbol{B}})$$

基于 HMM 的恶意域名检测算法具体流程如图 3所示。

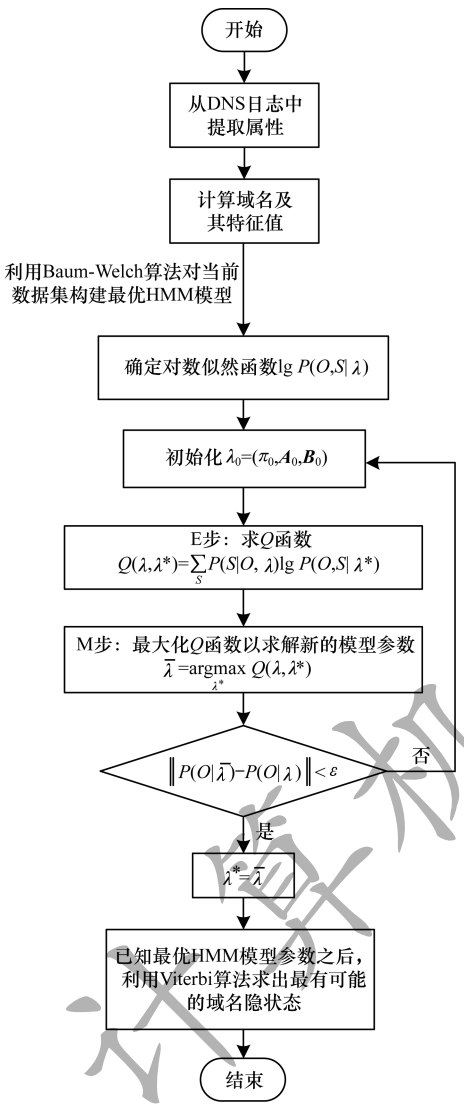

图 3 基于 HMM 的恶意域名检测算法流程

3 实验结果与分析

3.1 实验准备

实验准备工作主要包括：从 DNS 服务器中获取 DNS 日志，从不同数据源搜集善恶域名集，部署 Hadoop 和 Spark 集群。本文选取准确率和召回率 2 个指标来评估基于 HMM 的恶意域名识别系统的检测效果。其中，准确率指所有正确被检索的域名占所有实际被检索的域名比例，召回率指所有正确被检索的域名占所有应该被检索的域名比例。

3.1.1 善恶域名集获取

在域名检测系统中，为形成高质量的训练集并提高检测算法的效果，需要搜集一个包含范围较广且数量较多的善恶域名集。本文从 malware. com、Malware Domain List 等不同类型的数据源搜集大量的恶意域名，另外选取一些为逃避域名黑名单检测而利用随机字符生成的 C &C 域名。善意域名来自于 Alexa 网站中几乎涵盖各行业、各种类中排名靠前的可信赖域名。

3.1.2 测试环境

由于 Spark 平台没有存储大文件的能力，但本文系统的检测需要处理海量数据，因此引入 Hadoop 的分布式文件系统 hdfs。同时，利用 Hadoop YARN 使集群具备良好的扩展性。综上，本次实验需要搭建一个 Spark 集群和一个 Hadoop 集群，在 Spark 集群下，一台 PC 作为 master 节点，负责集群的资源管理，另外 3 台 PC 作为 Slave 节点，用于执行检测模型的计算任务。

3.2 HMM 模型下的分类效果测试

3.2.1 HMM 模型参数初始化

在实验中利用 HMM 模型进行域名检测时，根据域名的不同恶意类型，将其初始隐状态划分为 5 类，主要包括：基于网络钓鱼，基于 Domain-Flux，基于 Fast-Flux，基于混合 Domain-Flux 和 Fast-Flux，基于善意域名，分别用 $e_1 \sim e_5$ 进行表示。观测序列为 12 个从 DNS 日志中提取出的域名特征值。对 HMM 模型进行参数训练时，初始值的选取非常重要，其能大幅提高模型的计算准确性和效率，本文根据域名的危害程度给出 HMM 模型的初始化参数。域名隐状态转移概率矩阵为：

$$\boldsymbol{A}=\begin{pmatrix}0.720 & 0.210 & 0.030 & 0.002 & 0.001\\ 0.001 & 0.700 & 0.100 & 0.010 & 0.008\\ 0.000 & 0.000 & 0.730 & 0.050 & 0.019\\ 0.000 & 0.000 & 0.000 & 0.750 & 0.169\\ 0.000 & 0.000 & 0.000 & 0.000 & 0.999\end{pmatrix}$$

从状态转移概率矩阵 $\boldsymbol{A}$ 可以看出，当前域名危害状态向低危险状态转移的概率非常低，向相邻较高级别危害状态转移的概率较高，维持在本状态的概率最大。混淆矩阵概率分布为：

$$\boldsymbol{B}=\begin{pmatrix}0.21 & 0.20 & 0.23 & 0.10 & 0.07 & 0.04 & 0.06 & 0.02 & 0.01 & 0.12 & 0.08 & 0.16\\ 0.08 & 0.13 & 0.21 & 0.18 & 0.16 & 0.12 & 0.03 & 0.15 & 0.07 & 0.12 & 0.10 & 0.21\\ 0.01 & 0.02 & 0.01 & 0.05 & 0.21 & 0.15 & 0.12 & 0.10 & 0.20 & 0.13 & 0.10 & 0.06\\ 0.01 & 0.01 & 0.15 & 0.03 & 0.08 & 0.13 & 0.07 & 0.18 & 0.12 & 0.08 & 0.21 & 0.03\\ 0.02 & 0.01 & 0.03 & 0.05 & 0.04 & 0.10 & 0.06 & 0.12 & 0.07 & 0.11 & 0.20 & 0.12\end{pmatrix}$$

混淆矩阵概率分布表示在已知域名隐状态的情况下观测序列出现的概率。

初始化状态概率分布为：

$$\pi=\{\pi_{e_1},\pi_{e_2},\pi_{e_3},\pi_{e_4},\pi_{e_5}\}=\{0.42,0.21,0.13,0.07,0.01\}$$

从初始概率分布可以看出，域名危害强度最高

的状态 e_5 出现的概率很低,而危害强度最低的 e_1 状态出现的概率相对较高,说明大部分域名最初均处于比较安全的状态。

在模型初始化后,将初始概率 π、转移概率矩阵 $\boldsymbol{A}$、混淆矩阵 $\boldsymbol{B}$ 带入 Baum-Welch 算法(式(6)~式(8))中进行反复迭代,求解出最优的模型参数,然后通过 Viterbi 算法计算出域名最可能的隐状态。

3.2.2　原始参数下的分类器效果测试

Baum-Welch 算法利用 EM 算法的思想,通过反复迭代使其达到局部最优[22]。若能将 $\lambda=(\pi,\boldsymbol{A},\boldsymbol{B})$ 进行较好的参数初始化,就能得到使观测数据概率最大的全局最优解。经研究分析,初始概率 π 和转移概率矩阵 $\boldsymbol{A}$ 的初值对模型的影响较小,可通过随机选取的方法进行初始化,影响较大的是混淆矩阵 $\boldsymbol{B}$ 的初始概率值选取。因此,本文首先利用 K-means 方法进行归类划分[23-24],然后根据划分结果计算出混淆矩阵 $\boldsymbol{B}$ 的初值。用 Spark 集群建立模型,记录下模型的检测次数以及分类结果的准确率、召回率。原始参数下的测试结果如表 2、图 4 所示。

表 2　原始参数下的模型分类结果

检测次数	数据量(域名数)	准确率/%	召回率/%
1	17 200	80.32	82.78
2	26 906	83.51	80.72
3	37 809	86.79	83.89
4	48 976	81.86	79.86
5	57 680	90.32	81.57
6	65 789	82.57	80.78
7	78 970	89.85	84.67
8	89 070	84.62	87.64
9	90 789	91.52	89.32
10	109 080	87.76	85.87

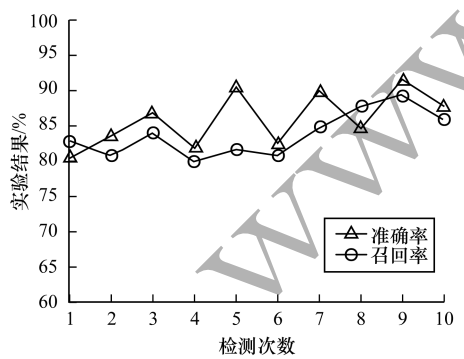

图 4　原始参数下的模型准确率与召回率

从表 2、图 4 可以看出,原始参数下 HMM 模型在测试集中对恶意域名分类的准确率和召回率均基本达标,但并未达到理想效果。因此,有必要通过聚类方法对观测概率初始值进行调整,以提升准确率和召回率。

3.2.3　调优参数下的分类器效果测试

通过 K 均值聚类方法对观测概率矩阵初始值进行划分,在初始值选取后,使用训练集和测试集分别进行模型训练和效果检测,结果如图 5 所示。从图 5 可以看出,进行参数优化后,模型的分类准确率和召回率均有显著提升。因此,可以初步判断 HMM 模型在经过观测概率矩阵的初始值调优后可以提高恶意域名的检测效果。

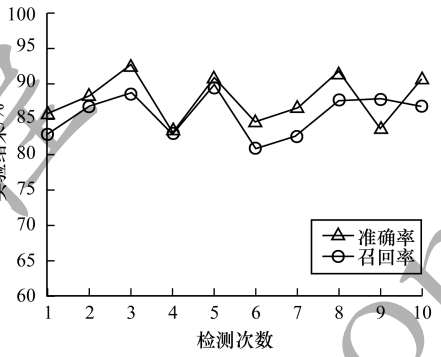

图 5　调优参数下的模型准确率与召回率

3.2.4　HMM 模型与 RF 模型的分类效果对比

在完成观测值的初始参数调优后,本文 HMM 模型的分类效果得到提升。在机器学习领域存在众多优秀的分类模型,RF 模型为其中的典型代表。利用 RF 模型训练分类器的具体步骤如下:

1)从初始训练集中有放回地重复随机抽取 n 个训练样本集。

2)根据从所有域名特征中随机抽取的 k 个特征,对新抽取的样本集建立决策树模型。

3)循环进行以上操作步骤 m 次,最后生成由 m 棵决策树组成的随机森林。

4)按照对森林中各决策树的投票数量来决定新数据分类。

在保持检测次数和相应域名数据量不变的前提下,使用训练集和测试集分别进行模型训练和效果检测。HMM 模型与 RF 模型的分类准确率和召回率对比结果分别如图 6、图 7 所示。

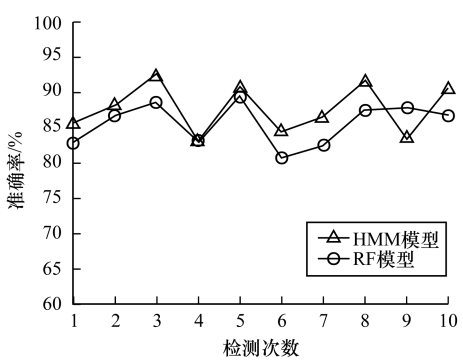

图 6　2 种模型的分类准确率对比

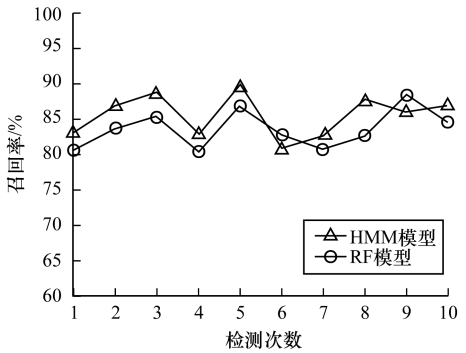

图 7 2 种模型的分类召回率对比

由图 6、图 7 可以看出,HMM 模型的准确率与召回率均优于 RF 模型,即 HMM 能够获得更加精确的检测结果。

4 结束语

本文提出一种在 Spark 平台上通过 HMM 模型进行恶意域名分类的方法。利用 Spark 基于内存的运算速度优势和对实时数据流进行处理的特性,结合 HMM 模型中的 Baum-Welch 算法和 Viterbi 算法进行精确分类,以提升检测系统的识别效果。实验结果表明,该方法可以提高分类结果的准确率与召回率。

参考文献

[1] 中国互联网络信息中心. 第 41 次中国互联网络发展状况统计报告[R/OL]. [2018-04-20]. http://www.cac.gov.cn/2018-01/31/c_1122347026.htm.

[2] YAN Yida, LIU Zhenyan, ZHONG Junwei, et al. Malicious domain detection based on machine learning [EB/OL]. [2018-04-25]. http://dpi-proceedings.com/index.php/dtcse/article/view/19866.

[3] SHI Yong, CHEN Gong, LI Juntao. Malicious domain name detection based on extreme machine learning[J]. Neural Processing Letters, 2018, 48(3): 1347-1357.

[4] ALIEYAN K, ALMOMANI A, MANASRAH A, et al. A survey of botnet detection based on DNS[J]. Neural Computing and Applications, 2017, 28(7): 1-18.

[5] POMOROVA O, SAVENKO O, LYSENKO S, et al. A technique for the botnet detection based on DNS-traffic analysis[C]//Proceedings of International Conference on Computer Networks. Berlin, Germany: Springer, 2015: 127-138.

[6] 马旸,强小辉,蔡冰,等. 大规模网络中基于集成学习的恶意域名检测[J]. 计算机工程,2016,42(11):170-176.

[7] SINGH K, GUNTUKU S C, THAKUR A, et al. Big data analytics framework for peer-to-peer botnet detection using random forests[J]. Information Sciences, 2014, 278(19): 488-497.

[8] ZAHARIA M, CHOWDHURY M, FRANKLIN M J, et al. Spark: cluster computing with working sets[C]//Proceedings of USENIX Conference on Hot Topics in Cloud Computing. San Diego, USA: USENIX Association, 2010: 2-10.

[9] MENG X, BRADLEY J, YAVUZ B, et al. MLlib: machine learning in Apache spark[J]. Journal of Machine Learning Research, 2016, 17(1): 1235-1241.

[10] VERMA A, MANSURI A H, JAIN N. Big data management processing with Hadoop MapReduce and spark technology: a comparison[C]//Proceedings of 2016 Symposium on Colossal Data Analysis and Networking. Washington D. C., USA: IEEE Press, 2016: 1-4.

[11] BAUM L E, PETRIE T, SOULES G, et al. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains[J]. Annals of Mathematical Statistics, 1970, 41(1): 164-171.

[12] 朱明,郭春生. 隐马尔可夫模型及其最新应用与发展[J]. 计算机系统应用,2010,19(7):255-259,216.

[13] 苏文胜,王奉涛,朱泓,等. 双树复小波域隐 Markov 树模型降噪及在机械故障诊断中的应用[J]. 振动与冲击,2011,30(6):47-52.

[14] 李航. 统计学习方法[M]. 北京:清华大学出版社,2012.

[15] 杨安驹,杨云,周媛媛,等. 基于隐马尔科夫模型的融合推荐算法[J]. 计算机与现代化,2015(9):60-65.

[16] SMYTH P. Hidden Markov models and neural networks for fault detection in dynamic systems[C]//Proceedings of 1993 IEEE-SP Workshop on Neural Networks for Signal Processing. Washington D. C., USA: IEEE Press, 1993: 582-592.

[17] VITERBI A J. A personal history of the Viterbi algorithm[J]. IEEE Signal Processing Magazine, 2006, 23(4): 120-142.

[18] JR G D F. The Viterbi algorithm[J]. Proceedings of the IEEE, 1973, 61(3): 268-278.

[19] WELCH L R. Hidden Markov models and the Baum-Welch algorithm[J]. IEEE Information Theory Society News Letter, 2003, 53(2): 194-211.

[20] 张峻飞. Hadoop 环境下的恶意域名检测方案研究[D]. 武汉:华中科技大学,2015.

[21] 赵成龙. 一种检测恶意域名的增量并行 SVM 算法研究与实现[D]. 武汉:华中科技大学,2016.

[22] MOLENBERGHS G, KENWARD M G. The expectation-maximization algorithm[M]//MOLENBERGHS G, KENWARD M G. Missing data in clinical studies. [S.l.]: Wiley-Blackwell, 2007.

[23] 夏丽莎. 基于隐马尔可夫模型的故障诊断及相关算法研究[D]. 武汉:华中科技大学,2014.

[24] KANUNGO T, MOUNT D M, NETANYAHU N S, et al. An efficient k-means clustering algorithm: analysis and implementation[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2002, 24(7): 881-892.

编辑 吴云芳