階層 Logistic Boosting を用いた コンピュータ・プログラムの異常検知

慶應義塾大学大学院理工学研究科　青島 達大
慶應義塾大学理工学部　南 美穂子

近年のインターネットの普及を受けて、データからコンピュータ・プログラムの異常行動をリアルタイムに検知する必要性が増している。さらに、攻撃手法の多様化 [3] から、正常なプログラムの行動のみを用いる手法への関心が高まっている。このような異常検知を統計的な学習問題として捉えるために、正常なプログラムによる次の行動を予測するモデルを構築し、そのモデルによって異常を検知する手法を考える。

一般に、行動のログなどのデータはシンボルの列とみなせる。このようなデータを解析する上では、異なるシンボル間の類似度が自明でないという問題がある。この問題を解決するために、本発表では、**階層 Logistic Boosting** という手法を提案する。階層 Logistic Boosting は、シンボル同士の共起から算出される類似度に基づく階層クラスタリングの結果を用いて、マルチクラスの logistic 回帰モデルを当てはめる手法である。階層クラスタリングを用いることで、異なるシンボル間の類似度が明らかになるだけでなく、特徴ベクトルの次元も削減できることから、学習がより安定する。ニューラルネットワーク [1] や隠れ Markov モデル [4] とは違い、提案手法は状態の初期値や多くのハイパーパラメータの調整が不要となる。

本発表では、特にコンピュータのアプリケーションの異常行動の検知への適用を考える。アプリケーションは OS によって隔離されており、外部とのやり取りには**システムコール**と呼ばれるインターフェイスを経由しなくてはいけない。よって、アプリケーションの異常行動を検知するために、システムコールの種類だけからなるシンボル列 (例えば、“`open`, `read`, `write`, ...”) を解析する。

階層 Logistic Boosting のアルゴリズムは次のようになる。以下、数学的な記述のために、シンボル全体の集合を $\mathcal{X}$ とおく。はじめに、シンボル列 $x_1, \ldots, x_L \in \mathcal{X}$ から 1 個ずつずらしながら、長さ J の部分列 (以下、J-gram という) を取り出し、その集合 $\{(x_1, \ldots, x_J), (x_2, \ldots, x_{J+1}), \ldots, (x_{L-J+1}, \ldots, x_L)\} \subset \mathcal{X}^J$ を標本 $\mathcal{D}$ とする。ここで、$w_{\mathcal{D}}(\boldsymbol{x})$ を、$\boldsymbol{x} \in \mathcal{D}$ が現れた回数とする。

各 J-gram $\boldsymbol{x} \in \mathcal{X}^J$ に対して、各シンボル $x \in \mathcal{X}$ が現れた回数を表す特徴ベクトルを $\boldsymbol{c}(\boldsymbol{x}) := (\sum_{j=1}^{J-1} \delta[x_j = x])_{x \in \mathcal{X}} \in \mathbb{Z}_{\geq 0}^{\mathcal{X}}$ とする。これを用いて、各シンボル $x \in \mathcal{X}$ に対する特徴ベクトル $\boldsymbol{v}_x := (w_{\mathcal{D}}(\boldsymbol{x}_i) c_x(\boldsymbol{x}_i))_{i=1}^n$ を作る。各シンボル間で $\cos^2$ 非類似度 $d(x, x') := 1 - (\boldsymbol{v}_x^\top \boldsymbol{v}_{x'})^2 / ((\boldsymbol{v}_x^\top \boldsymbol{v}_x)(\boldsymbol{v}_{x'}^\top \boldsymbol{v}_{x'})) \in [0, 1], x, x' \in \mathcal{X}$ を計算し、この非類似度に基づく階層クラスタリングによって、シンボル同士の類似性を表す二分木を得る。この二分木を根から辿ることによって、次第に細分化されるシンボルのクラスタ群の集合 $\{\mathcal{C}_k\}_{k=1}^K$ を得る。最後に、Gradient Boosting を用いて Logisitc 回帰モデルを当てはめる。このとき、Boosting の各段階 t では、クラスタ群 $\mathcal{C}_{\min\{t,K\}}$ を用いて、同じクラスタに属するシンボルを同じシンボルとみなす。

実データ [2, 4] による実験結果より、提案手法は、異常な行動によるログを全て検知しつつ、誤って異常であると判断してしまう割合は 0.3%以下であった。また、提案手法による検知の解釈が容易であることを確認するために、正常な行動のログの特徴や、どのようなパターンが実際に異常であると判断されたかについても考察する。

参考文献

[1] B. Cha, B. Vaidya and S. Hani. “Anomaly Intrusion Detection for System Call Using the Soundex Algorithm and Neural Networks.” *10th IEEE Symposium on Computers and Communications (ISCC’05)*, 2005, pp. 427–433.

[2] S. A. Hofmeyr, S. Forrest, and A. Somayaji. “Intrusion Detection Using Sequences of System Calls.” *Journal of computer security*, 1998, vol. 6, no. 3, pp. 151–180.

[3] C. Hosmer. “Polymorphic and Metamorphic Malware.” *Black Hat USA 2008*. [Online], Available: `https://www.blackhat.com/presentations/bh-usa-08/Hosmer/BH_US_08_Hosmer_Polymorphic_Malware.pdf`, 2008.

[4] C. Warrender, S. Forrest, and B. Pearlmutter. “Detecting Intrusions Using System Calls: Alternative Data Models.” *Proceedings of the 1999 IEEE Symposium on Security and Privacy*, 1999, pp. 133–145.