BAT DEEP LEARNING METHODS ON NETWORK INTRUSTION DETECTION USING NSL-KDD DATASET

B.S.Swapnashanthi[1], K.Chinmayee[2], M.kovid[3], M.tejaswi[4], N.rakesh[5]

[1]Assistant Professor, Department of CSE, Sri Indu Institute of Engineering & Technology, Hyderabad

[2,3,4,5] IV[th] Btech Student, Department of CSE, Sri Indu Institute of Engineering & Technology, Hyderabad

ABSTRACT

Intrusion detection can identify unknown attacks from network traffics and has been an effective means of network security. Nowadays, existing methods for network anomaly detection are usually based on traditional machine learning models, such as KNN, SVM, etc. Although these methods can obtain some outstanding features, they get a relatively low accuracy and rely heavily on manual design of traffic features, which has been obsolete in the age of big data. To solve the problems of low accuracy and feature engineering in intrusion detection, a traffic anomaly detection model BAT is proposed. The BAT model combines BLSTM (Bidirectional Long Short-term memory) and attention mechanism. Attention mechanism is used to screen the network flow vector composed of packet vectors generated by the BLSTM model, which can obtain the key features for network traffic classification. In addition, we adopt multiple convolutional layers to capture the local features of traffic data. As multiple convolutional layers are used to process data samples, we refer BAT model as BAT-MC. The softmax classifier is used for network traffic classification. The proposed end-to-end model does not use any feature engineering skills and can automatically learn the key features of the hierarchy. It can well describe the network traffic behavior and improve the ability of anomaly detection effectively. We test our model on a public benchmark dataset, and the experimental results demonstrate our model has better performance than other comparison methods.

key words: Deep learning,NSL-KDD dataset,BAT,neural network,BLSTM,attention mechanism.

INTRODUCTION

1.1 MOTIVATION

Intrusion detection plays an important part in ensuring network information security. Machine learning methods have been widely used in intrusion detection to identify malicious traffic. However, these methods belong to shallow learning and often emphasize feature engineering and selection. They have difficulty in features selection and cannot effectively solve the massive intrusion data classification problem, which leads to low recognition accuracy and high false alarm rate. In recent years, intrusion detection methods based on deep learning have been proposed successively.

1.2 PROBLEM DEFINITION

The existing methods for network anomaly detection are usually based on traditional machine learning models, such as KNN, SVM, etc. Although these methods can obtain some outstanding features, they get a relatively low accuracy and rely heavily on manual design of traffic features, which has been absolete in the age of big data.

1.3 OBJECTIVE OF PROJECT

We adopt multiple convolutional layers to capture the local features of traffic data. As multiple convolutional layers are used to process data samples, we refer BAT model as BAT-MC. The softmax classifier is used for network traffic classification.

1.4 LIMITATIONS OF PROJECT

The current deep learning methods in the network traffic classification research don't make full use of the network traffic structured information. Drawing on the application methods of deep learning in the field of natural language processing, we propose a novel model BAT-MC via the two phase's learning of BLSTM and attention on the time series features for intrusion detection using NSL-KDD dataset.

2.LITERATURE SURVEY

Sarika Choudhary et al, the latest buzzword in internet technology nowadaysisthe Internet of Things. The Internet of Things (IoT) is an ever-growing network which will transform real-world objects into smart or intelligent virtual objects. IoT is a heterogeneous network in which devices with different protocols can connect with each other in order to exchange information. These days, human life depends upon the smart things and their activities. Therefore, implementing protected communications in the IoT network is a challenge. Since the IoT network is secured with

authentication and encryption, but not secured against cyber-attacks, an Intrusion Detection System is needed. This research article focuses on IoT introduction, architecture, technologies, attacks and IDS. The main objective of this article is to provide a general idea of the Internet of Things, various intrusion detection techniques, and security attacks associated with IoT.

Network intrusion detection

B. Mukherjee et,al Intrusion detection is a new, retrofit approach for providing a sense of security in existing computers and data networks, while allowing them to operate in their current "open" mode. The goal of intrusion detection is to identify unauthorized use, misuse, and abuse of computer systems by both system insiders and external penetrators. The intrusion detection problem is becoming a challenging task due to the proliferation of heterogeneous computer networks since the increased connectivity of computer systems gives greater access to outsiders and makes it easier for intruders to avoid identification. Intrusion detection systems (IDSs) are based on the beliefs that an intruder's behavior will be noticeably different from that of a legitimate user and that many unauthorized actions are detectable. Typically, IDSs employ statistical anomaly and rulebased misuse models in order to detect intrusions. A number of prototype IDSs have been developed at several institutions, and some of them have.

Network intrusion detection system: A machine learning approach

Mrutyunjaya Panda et,al Intrusion detection systems (IDSs) are currently drawing a great amount of interest as a key part of system defence. IDSs collect network traffic information from some point on the network or computer system and then use this information to secure the network. Recently, machine learning methodologies are playing an important role in detecting network intrusions (or attacks), which further helps the network administrator to take precautionary measures for preventing intrusions. In this paper, we propose to use ten machine learning approaches that include Decision Tree (J48), Bayesian Belief Network, Hybrid Na¨ıve Bayes with Decision Tree, Rotation Forest, Hybrid J48 with Lazy Locally weighted learning, Discriminative multinomial Na¨ıve Bayes, Combining random Forest with Na¨ıve Bayes and finally ensemble of classifiers using J48 and NB with AdaBoost (AB) to detect network intrusions efficiently. We use

NSL-KDD dataset, a variant of widely used KDDCup 1999 intrusion detection benchmark dataset, for evaluating our proposed machine learning approaches for network intrusion detection. Finally, Experimental results with 5-class classification are demonstrated that include: Detection rate, false positive rate, and average cost for misclassification. These are used to aid a better understanding for the researchers in the domain of network intrusion detection.

A new intrusion detection system based on KNN classification algorithm in wireless sensor network

L. Pan et,al The Internet of Things has broad application in military field, commerce, environmental monitoring, and many other fields. However, the open nature of the information media and the poor deployment environment have brought great risks to the security of wireless sensor networks, seriously restricting the application of wireless sensor network. Internet of Things composed of wireless sensor network faces security threats mainly from Dos attack, replay attack, integrity attack, false routing information attack, and flooding attack. In this paper, we proposed a new intrusion detection system based on K-nearest neighbor (K-nearest neighbor, referred to as KNN below) classification algorithm in wireless sensor network. This system can separate abnormal nodes from normal nodes by observing their abnormal behaviors, and we analyse parameter selection and error rate of the intrusion detection system. The paper elaborates on the design and implementation of the detection system. This system has achieved efficient, rapid intrusion detection by improving the wireless ad hoc on-demand distance vector routing protocol (Ad hoc On-Demand Distance the Vector Routing, AODV). Finally, the test results show that: the system has high detection accuracy and speed, in accordance with the requirement of wireless sensor network intrusion detection.

2.2 EXISTING SYSTEM:

Most algorithms have been considered for use in the past. In the authors make a summary of pattern matching algorithm in Intrusion Detection System: KMP algorithm, BM algorithm, BMH algorithm, BMHS algorithm, AC algorithm and AC-BM algorithm. Experiments show that the improved algorithm can accelerate the matching speed and has a good time performance. In [17], Naive approach, Knuth-MorrisPratt algorithm and RabinKarp Algorithm are compared in order to check which of them

is most efficient in pattern/intrusion detection. Pcap files have been used as datasets in order to determine the efficiency of the algorithm by taking into consideration their running times respectively.

2.2.1 DRAWBACKS OF EXISTING SYSTEM:

1.We are also facing various security threats. Network viruses, eavesdropping and malicious attacks are on the rise, causing network security to become the focus of attention of the society and government departments.

2. To identify various malicious network traffics, especially unexpected malicious network traffics, is a key problem that cannot be avoided.

2.3 PROPOSED SYSTEM:

The accuracy of the BAT-MC network can reach 84.25%, which is about 4.12% and 2.96% higher than the existing CNN and RNN model, respectively. The following are some of the key contributions and findings of our work:

1) We propose an end-to-end deep learning model BAT-MC that is composed of BLSTM and attention mechanism. BAT-MC can well solve the problem of intrusion detection and provide a new research method for intrusion detection.

2) We introduce the attention mechanism into the BLSTM model to highlight the key input. Attention mechanism conducts feature learning on sequential data composed of data package vectors. The obtained feature information is reasonable and accurate.

3) We compare the performance of BAT-MC with traditional deep learning methods, the BAT-MC model can extract information from each packet. By making full use of the structure information of network traffic, the BAT-MC model can capture features more comprehensively.

2.3.2 ADVANTAGES OF PROPOSED SYSTEM:

1.The BAT-MC model consists of five components, including the input layer, multiple convolutional Layers, BSLTM layer, attention layer and output layer, from bottom to top.

2. At the input layer, BAT-MC model converts each traffic byte into a one-hot data format. Each traffic byte is encoded as an n-dimensional vector. After traffic byte is converted into a numerical form, we perform normalization operation.

2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost

estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- ECONOMICAL FEASIBILITY
- TECHNICAL FEASIBILITY
- SOCIAL FEASIBILITY

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

2.5 FEATURES OF THE PROJECT

The BAT-MC model consists of five components, including the input layer, multiple convolutional Layers, BSLTM layer, attention layer and output layer, from bottom to top. At the input layer, BAT-MC model converts each traffic byte into a one-hot data format. Each traffic byte is encoded as an n-dimensional vector. After traffic byte is converted into

a numerical form, we perform normalization operations. At the multiple convolutional layer, we convert the numerical data into traffic images. Convolutional operation is used as a feature extractor that takes an image representation of data packet. At the BLSTM layer, BLSTM model which connects the forward LSTM and the backward LSTM is used to extract features on the the traffic bytes of each packet. BLSTM model can learn the sequential characteristics within the traffic bytes because BLSTM is suitable to the structure of network traffic.

A. DATA PREPROCESSING LAYER

There are three symbolic data types in NSL-KDD data features: protocol type, flag and service. We use one-hot encoder mapping these features into binary vectors. One-Hot Processing: NSL-KDD dataset is processed by one-hot method to transform symbolic features into numerical features. For example, the second feature of the NSL-KDD data sample is protocol type. The protocol type has three values: tcp, udp, and icmp. One-hot method is processed into a binary code that can be recognized by a computer, where tcp is [1, 0, 0], udp is [0, 1, 0], and icmp is [0, 0, 1].

B. MULTIPLE CONVOLUTIONAL LAYERS

After the above processing operations, convolutional layer is used to capture the local features of traffic data. Convolutional layer is the most important part of the CNN, which convolves the input images (or feature maps) with multiple convolutional kernels to create different feature maps. The shallower convolutional layers whose receptive field is narrow can extract local information,and while the deeper layers can capture global information with larger vision field. Hence, as the number of the convolutional layers increases, the scale of the convolutional feature gradually becomes coarser.

C. BLSTM LAYER

For the time series data composed of traffic bytes, BLSTM can effectively use the context information of data for feature learning. The BLSTM is used to learn the time series feature in the data packet. Traffic bytes of each data packet are sequentially input into an BLSTM, which finally obtain a packet vector. BLSTM is an enhanced version of LSTM (Long Short-Term Memory). The BLSTM model is used to extract coarse-grained features by connecting forward LSTM and backward LSTM.

D. ATTENTION LAYER

BLSTM eventually generates a packet vector for each packet. These packet vectors are arranged in the order of interaction between the two parties in the network stream to form a sequence of

packet vectors. The relationships within packet vectors will be learned by attention layer. Attention mechanism is used to adjust probability of packet vectors so that our model pays more attention to important features.

E. MODEL TRAINING

Training the proposed network contains a forward pass and a backward pass. Forward Propagation The BAT-MC model is mainly composed of BLSTM layer and attention layer, each of which presents different structures and thus plays different role in the whole model. The forward propagation is conducted from BLSTM layer to attention layer. The input of current model is obtained by the processing of the previous model. After the completion of forward propagation, the final recognition result is obtained.

Backward Propagation: The model is trained with adam. Adam is calculated by the back-propagation algorithm. Error differentials are back-propagated with the forward-backward algorithm. Back-Propagation Through Time (BPTT) is applied to calculate the error differentials.

2.6 TECHNOLOGIES USED FOR IMPLEMENTION

WHAT IS PYTHON :-

Below are some facts about Python.

Python is currently the most widely used multi-purpose, high-level programming language.

Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.

Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time.

Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.

The biggest strength of Python is huge collection of standard library which can be used for the following –

- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like Opencv, Pillow)
- Web scraping (like Scrapy, BeautifulSoup, Selenium)
- Test frameworks
- Multimedia

Advantages of Python :-

Let's see how Python dominates over other languages.

1. Extensive Libraries

Python downloads with an extensive library and it contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading, databases, CGI, email, image manipulation, and more. So, we don't have to write the complete code for that manually.

2. Extensible

As we have seen earlier, Python can be **extended to other languages**. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

3. Embeddable

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add **scripting capabilities** to our code in the other language.

4. Improved Productivity

The language's simplicity and extensive libraries render programmers **more productive** than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

5. IOT Opportunities

Since Python forms the basis of new platforms like Raspberry Pi, it finds the future bright for the Internet Of Things. This is a way to connect the language with the real world.

6. Simple and Easy

When working with Java, you may have to create a class to print **'Hello World'**. But in Python, just a print statement will do. It is also quite **easy to learn, understand,** and **code.** This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java.

7. Readable

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and **indentation is mandatory.** This further aids the readability of the code.

8. Object-Oriented

This language supports both the **procedural and object-oriented** programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the **encapsulation of data** and functions into one.

9. Free and Open-Source

Like we said earlier, Python is **freely available.** But not only can you **download Python** for free, but you can also download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.

10. Portable

When you code your project in a language like C++, you may need to make some changes to it if you want to run it on another platform. But it isn't the same with Python. Here, you need to **code only once**, and you can run it anywhere. This is called **Write Once Run Anywhere (WORA)**. However, you need to be careful enough not to include any system-dependent features.

11. Interpreted

Lastly, we will say that it is an interpreted language. Since statements are executed one by one, **debugging is easier** than in compiled languages.

Any doubts till now in the advantages of Python? Mention in the comment section.

Advantages of Python Over Other Languages

1. Less Coding

Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don't have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

2. Affordable

Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.

The 2019 Github annual survey showed us that Python has overtaken

Java in the most popular programming language category.

3. Python is for Everyone

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and **machine learning**, automate things, do web scraping and also build games and powerful visualizations. It is an all-rounder programming language.

Disadvantages of Python

So far, we've seen why Python is a great choice for your project. But if you choose it, you should be aware of its consequences as well. Let's now see the downsides of choosing Python over another language.

1. Speed Limitations

We have seen that Python code is executed line by line. But since Python is interpreted, it often results in **slow execution**. This, however, isn't a problem unless speed is a focal point for the project. In other words, unless high speed is a requirement, the benefits offered by Python are enough to distract us from its speed limitations.

2. Weak in Mobile Computing and Browsers

While it serves as an excellent server-side language, Python is much rarely seen on the **client-side**. Besides that, it is rarely ever used to implement smartphone-based applications. One such application is called **Carbonnelle**. The reason it is not so famous despite the existence of Brython is that it isn't that secure.

3. Design Restrictions

As you know, Python is **dynamically-typed**. This means that you don't need to declare the type of variable while writing the code. It uses **duck-typing**. But wait, what's that? Well, it just means that if it looks like a duck, it must be a duck. While this is easy on the programmers during coding, it can **raise run-time errors**.

4. Underdeveloped Database Access Layers

Compared to more widely used technologies like **JDBC (Java DataBase Connectivity)** and **ODBC (Open DataBase Connectivity)**, Python's database access layers are a bit

underdeveloped. Consequently, it is less often applied in huge enterprises.

5. Simple

No, we're not kidding. Python's simplicity can indeed be a problem. Take my example. I don't do Java, I'm more of a Python person. To me, its syntax is so simple that the verbosity of Java code seems unnecessary.

This was all about the Advantages and Disadvantages of Python Programming Language.

History of Python : -

What do the alphabet and the programming language Python have in common? Right, both start with ABC. If we are talking about ABC in the Python context, it's clear that the programming language ABC is meant. ABC is a general-purpose programming language and programming environment, which had been developed in the Netherlands, Amsterdam, at the CWI (Centrum Wiskunde &Informatica). The greatest achievement of ABC was to influence the design of Python.Python was conceptualized in the late 1980s. Guido van Rossum worked that time in a project at the CWI, called Amoeba, a distributed operating system. In an interview with Bill Venners[1], Guido van Rossum said: "In the early 1980s, I worked as an implementer on a team building a language called ABC at Centrum voor Wiskunde en Informatica (CWI). I don't know how well people know ABC's influence on Python. I try to mention ABC's influence because I'm indebted to everything I learned during that project and to the people who worked on it."Later on in the same Interview, Guido van Rossum continued: "I remembered all my experience and some of my frustration with ABC. I decided to try to design a simple scripting language that possessed some of ABC's better properties, but without its problems. So I started typing. I created a simple virtual machine, a simple parser, and a simple runtime. I made my own version of the various ABC parts that I liked. I created a basic syntax, used indentation for statement grouping instead of curly braces or begin-end blocks, and developed a small number of powerful data types: a hash table (or dictionary, as we call it), a list, strings, and numbers."

3.RESULT:

This model effectively avoids the problem of manual design features. Performance of the BAT-MC method is tested by KDDTest+ and KDDTest-21 dataset.

Experimental results on the NSL-KDD dataset indicate that the BAT-MC model achieves pretty high accuracy. By comparing with some standard classifier, these comparisons show that BAT-MC models results are very promising when compared to other current deep learning-based methods. Hence, we believe that the proposed method is a powerful tool for the intrusion detection problem.

4.CONCLUSION:

The current deep learning methods in the network traffic classification research don't make full use of the network traffic structured information. Drawing on the application methods of deep learning in the field of natural language processing, we propose a novel model BAT- MC via the two phase's learning of BLSTM and attention on the time series features for intrusion detection using NSL-KDD dataset. BLSTM layer which connects the forward LSTM and the backward LSTM is used to extract features on the the traffic bytes of each packet. Each data packet can produce a packet vector. These packet vectors are arranged to form a network flow vector. Attention layer is used to perform feature learning on the network flow vector composed of packet vectors. The above feature learning process is automatically completed by deep neural network without any feature engineering technology

5.REFERENCES

B. B. Zarpelo, R. S Miani, C. T. Kawakani, and S. C. de Alvarenga, ''A survey of intrusion detection in Internet of Things,'' J. Netw. Comput. Appl., vol. 84, pp. 25–37, Apr. 2017.

2. B. Mukherjee, L. T. Heberlein, and K. N. Levitt, ''Network intrusion detection,'' IEEE Netw., vol. 8, no. 3, pp. 26–41, May 1994.

3. S. Kishorwagh, V. K. Pachghare, and S. R. Kolhe, ''Survey on intrusion detection system using machine learning techniques,'' Int. J. Control Automat., vol. 78, no. 16, pp. 30–37, Sep. 2013.

4. N. Sultana, N. Chilamkurti, W. Peng, and R. Alhadad, ''Survey on SDN based network intrusion detection system using machine learning approaches,'' Peer-to-Peer Netw. Appl., vol. 12, no. 2, pp. 493–501, Mar. 2019.

5. M. Panda, A. Abraham, S. Das, and M. R. Patra, ''Network intrusion detection system: A machine learning approach,'' Intell. Decis. Technol., vol. 5, no. 4, pp. 347–356, 2011.

6 W. Li, P. Yi, Y. Wu, L. Pan, and J. Li, ''A new intrusion detection system based on KNN classification algorithm in

wireless sensor network,'' J. Electr. Comput. Eng., vol. 2014, pp. 1–8, Jun. 2014.

7.S. Garg and S. Batra, ''A novel ensembled technique for anomaly detection,'' Int. J. Commun. Syst., vol. 30, no. 11, p. e3248, Jul. 2017.

8. F. Kuang, W. Xu, and S. Zhang, ''A novel hybrid KPCA and SVM with GA model for intrusion detection,'' Appl. Soft Comput.., vol. 18, pp. 178–184, May 2014.

9. W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, ''Malware traffic classification using convolutional neural network for representation learning,'' in Proc. Int. Conf. Inf. Netw. (ICOIN), 2017, pp. 712–717.

10. P. Torres, C. Catania, S. Garcia, and C. G. Garino, ''An analysis of Recurrent Neural Networks for Botnet detection behavior,'' in Proc. IEEE Biennial Congr. Argentina (ARGENCON), Jun. 2016, pp. 1–6.