RECIPROC-IT

Research Report on the Introduction of Machine Learning into the EBIOS Risk Management Framework

January 15, 2024

Final Research Report

Zuoyu Zhang

Supervised by:
Matthis Peltier
Fatima Brahami

August 2022

Contents

1 Introduction 3
1.1 EBIOS Risk Management Framework . . . 3
1.2 Project Objectives . . . 4
1.3 Project significance . . . 4
1.4 Data set selection . . . 4

2 Data pre-processing 4
2.1 Basic Operations . . . 4
2.2 Text Conversion . . . 4

3 Introduction to the realized network 5
3.1 Multi-Layer Perceptron (MLP) . . . 5
3.2 Bayesian Deep Neural Network (BNN) . . . 6

4 Specific implementation 6

5 Analysis of results 7

6 Risk loss estimation assumptions 8

Abstract

Risk management is one of the most important aspects of the safe conduct of business projects, and the gradual development of artificial intelligence technologies nowadays offers new possibilities for risk management. The purpose of this project is to investigate how to add an intelligent risk prevention and control module to the risk management framework EBIOS, so that artificial intelligence techniques can assist in the risk prevention and control process. Based on this, we propose an idea to use Bayesian deep neural networks in Seminar 1 of the EBIOS framework to analyze the company's geographic location, industry and other information to determine which categories of risk the company is most likely to be exposed to. In addition, we propose an idea for predicting the risk of loss by using regression models such as traditional logistic regression, polynomial regression, and multilayer perceptron to analyze the amount of money that the target company could lose by being exposed to this risk, based on the amount of money lost by similar companies in the same field, in the same project, and in the history of such attacks. The experiments in this project use the CISSM cyber-attack database as the experimental data set. After the basic data preprocessing operation, the fasttext model is used to perform the word vector conversion operation on the English language in the data, and the processed data is put into the network for training tests, and the final prediction accuracy is 92.13%, which provides some possibility to apply it in practice.

Key words

EBIOS, risk management, machine learning, Bayesian deep neural networks, multilayer perceptron, regression , natural language processing

1 Introduction

1.1 EBIOS Risk Management Framework

The EBIOS method is a digital risk management method that uses workshops within the organization and with partners to obtain useful resources and arguments to assess the risks associated with digital projects and identify the security measures that need to be taken to control these risks.

The EBIOS method is a scenario to fundamental technical operation method, which can be described as follows: understand and analyze the actual scenario of the object of study - predict potential risks (source and content) - identify compatible strategic operations for each proposed risk - generate a global solution.

The EBIOS framework consists of 5 workshops, and we focused on Workshop 1 for this experiment, which had the main objective of defining the general framework of the study (participants in the different workshops and roles and responsibilities in the study), specifically by identifying, based on the information provided by the study participants about the mission, main tasks, business value, and supporting assets of the digital project enterprise (key importance of the information) Technology areas of the enterprise, ranking their business value in order of magnitude. Conceptualize possible security risks and determine the security foundation to be adopted

1.2 Project Objectives

The goal of our project is to use a Bayesian deep neural network approach to machine learning to predict the most likely attacks on a given company based on information about the company, including location, industry, company name, most likely attack motives (e.g., financial, political, personal, etc.), and most likely attack types (including three types of attacks: information, process, and both). The categories of attacks that can be predicted include 11 types of attacks, namely data attacks, exploitation of application servers, exploitation of end hosts, exploitation of data in transit, exploitation of infrastructure, exploitation of network infrastructure, exploitation of sensors, external denial of service, internal denial of service, message manipulation, and physical attacks.

1.3 Project significance

In the process of risk assessment, it is an important question to locate what kind of attack the research project will be subjected to with what probability, and it is very important for the subsequent risk prevention and control process. It is of high practical significance to use machine learning algorithms to study examples of attacks on other companies, so as to deduce what kind of risk the object of our current research may face with what probability, and it can better guide the development of the next risk management measures to better avoid losses.

1.4 Data set selection

We choose the University of Maryland's cyberattack database CISSM as the training and testing dataset this time, and select 8021 attack data records from this database from July 30, 2022 to October 1, 2014 for the study.

2 Data pre-processing

2.1 Basic Operations

First, some data points with serious missing information are directly deleted, discrete data points are deleted, and information is complemented for data points with only one or two missing characteristics information. And, for each risk event in the database may have one or more of the above-mentioned attack types, we convert these 11 types of attack labels into 11-bit 0, 1 encoding, that is, for each event data, if such an attack occurs, the attack corresponding to the encoding bit value is 1, otherwise the value is 0. For example, an event data corresponding to the occurrence of the first type of data attacks and the third type of exploitation of the terminal For example, if an event data corresponds to the occurrence of the first type of data attack and the third type of exploitation of the terminal, then it corresponds to the output label 10100000000.

2.2 Text Conversion

Since all the features in the dataset are represented by English words, we need to convert the words into the form of numeric vectors for representation using natural language processing models. The commonly used conversion methods are unique thermal coding and some language models of natural language processing. We implement two conversion methods here, unique thermal coding and fasttext word representation model, and finally choose the processing results of fasttext for neural network training and testing, which will be introduced below.

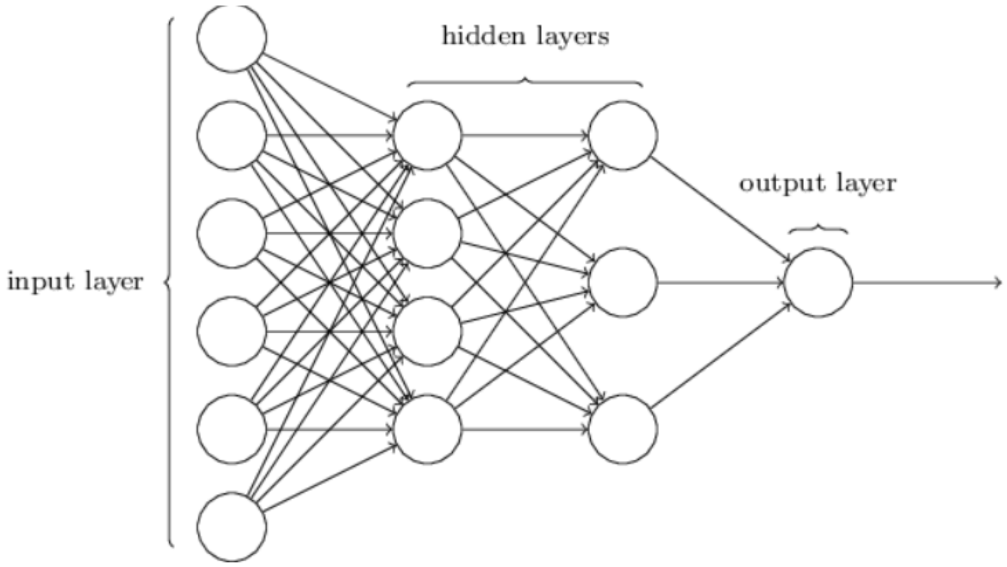
Unique hot encoding: This method represents each word as a very long vector. The dimension of this vector is the size of the vocabulary, where the majority of the elements are 0 and only one dimension has a value of 1, and this dimension represents the current word. The unique hot representation only symbolizes the word without any semantic information and does not take into account the correlation between words. For example, the word data in the first position is coded as 1000, and attack in the second position is coded as 0100, and the coded result is only related to the position of the word.

fasttext word vector model: It is a text classification tool launched by Facebook AI Research, which can also be used to train word vectors, sentence vectors, etc. The unsupervised model in the fasttext library can quickly and accurately analyze the word vector corresponding to each word in the text based on the input text

3 Introduction to the realized network

3.1 Multi-Layer Perceptron (MLP)

A Multilayer Perceptron (MLP) is a forward-structured artificial neural network that maps a set of input vectors to a set of output vectors.The MLP can be thought of as a directed graph consisting of multiple layers of nodes, each layer fully connected to the next. In addition to the input nodes, each node is a neuron (or processing unit) with a nonlinear activation function. The MLP model is a supervised learning model, and during the training process the MLP model gradually finds the appropriate values of weights and biases for these parameters through a back-propagation algorithm, so that the output meets our requirements.



3.2 Bayesian Deep Neural Network (BNN)

Bayesian neural networks differ from general neural networks in that the weight parameters are random variables rather than definite values. That is, in contrast to traditional neural networks that fit labeled values with loss functions such as cross-entropy, mse, etc., Bayesian neural networks fit posterior distributions. Combining probabilistic modeling and neural networks and being able to give confidence in the prediction results. In a Bayesian network, the prior probability distribution is used to describe the key parameters and serves as the input to the neural network. The output of the neural network is used to describe the likelihood of a particular probability distribution. The posterior distribution is computed by sampling or variational inference.

4 Specific implementation

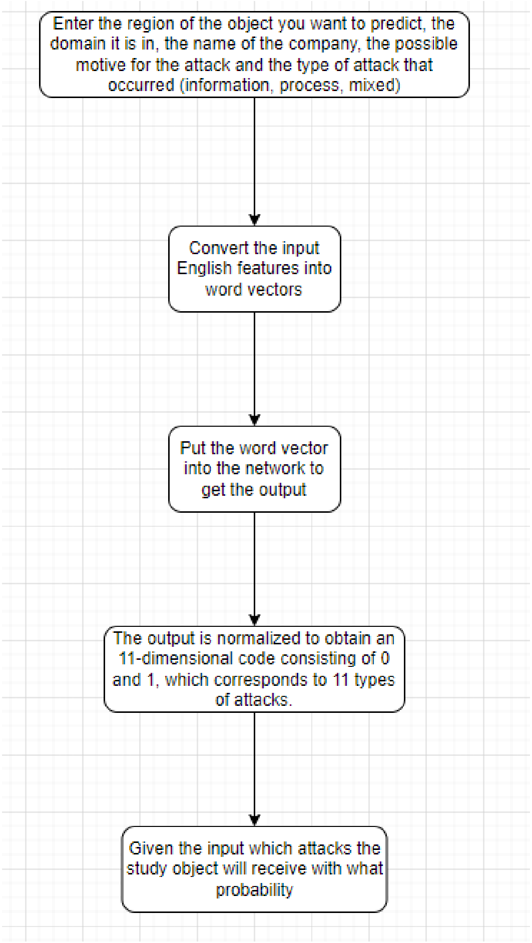
The experiments in this project are based on the deep learning library pytorch and Pyro, a general purpose probabilistic programming language written in Python supported by PyTorch on the back end, together to implement Bayesian neural networks.

In the data preprocessing phase we use the fasttext model to convert the words in the database into 150-dimensional word vectors stored as npy files, and then in the network implementation part we first import the preprocessed dataset and divide the whole into a training set and a test set in a 9:1 ratio. A four-layer MLP neural network with 750-dimensional input features and 11-dimensional output features is defined.

Since for Bayesian neural networks, the prediction weights and biases are random variables and the core of Bayesian networks for modeling and predicting data lies in the inference of a good posterior distribution, a variational inference method is used in the process of network training. Therefore, we first initialize the weights and variances of the network based on the normal distribution, and then define a function to guide the network in approximating the posterior probability distribution during the training process. Finally, the SVI variational inferer is defined and training begins.

Our network loses convergence after 160 theoretical training and finally achieves 92.14% accuracy on the test set, which can be said to be relatively satisfactory considering the small amount of training data and the complexity of the multi-label regression problem.

We can explain the prediction process more clearly with the following flow chart


Enter the region of the object you want to predict, the domain it is in, the name of the company, the possible motive for the attack and the type of attack that occurred (information, process, mixed)
Convert the input English features into word vectors
Put the word vector into the network to get the output
The output is normalized to obtain an 11-dimensional code consisting of 0 and 1, which corresponds to 11 types of attacks.
Given the input which attacks the study object will receive with what probability

5 Analysis of results

Our experiment this time introduces Bayesian network and Bayesian network improved by adding convolutional layers for risk regression, which is relatively novel and has better results. However, the results of the experiments are more limited by the data set, which may be more difficult for a wide range of generalization, and the 11 types of attacks that can be predicted, the variety is less, and the reference significance of the results is less. Here are some reflections for this project.

1. There are more instances of some attack classes in the experimental dataset, while there are basically no instances of some attack classes, and the extreme imbalance in the dataset causes the final experimental results to fall short of expectations.

2. The characteristics in the experimental dataset are not specific enough, such as the region column, which covers many countries and regions in the world and is not highly relevant.

3. The inputs are less specific and do include information that is critical to risk prediction, such as the company's asset position, historical attack history, etc.

4. In order to quickly obtain word vector representations of English text, word vector training is performed only for the input dataset text, and text semantic understanding training is not performed in a large number of texts, which may not be specific and sufficient.

5. Attempts to add some network improvements of reinforcement learning and convolutional layers to the Bayesian neural network, but none of the final results are as accurate as the basic Bayesian network.

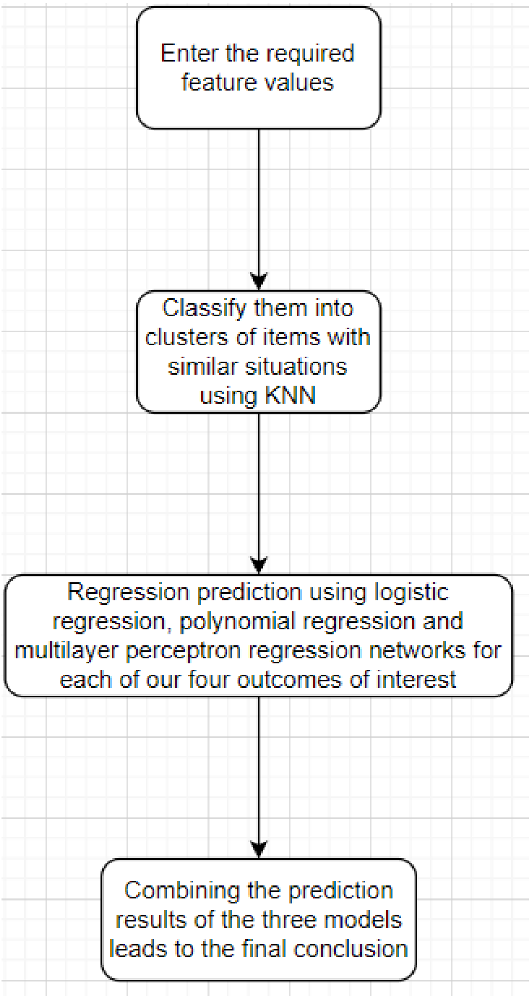
By comparing the results of Bayesian neural network with other networks such as CNN (70% accuracy) and Resnet (77% accuracy), we can see that Bayesian neural network is still more suitable for such probability regression problems. Through the analysis of the results, we should still return to the selection of data, which can be collected in the form of questionnaires targeting to various companies to collect relevant data, which may improve the current public data and the problems that exist, and also make the results more diverse and realistic.

6 Risk loss estimation assumptions

We first collect data from the dataset, including information such as company name, specific region, specific industry, main content category of the project, project assets, type of attack suffered, amount of loss, loss data, lost users or customers, number of days down, etc. This information can be either a real event or derived from the opinion of some security experts, where the first six items are the input characteristics and the last four items are the characteristics we want to obtain in the regression network to obtain the prediction results.

We classify the collected data using KNN, and then train the regression model using logistic regression, polynomial regression, and MLP regression for each similar class of cases. After that, we will first classify each new input data, and then check the regression results on each of the three regression models to make a final judgment by combining the three results.

The specific process is as follows.


Enter the required feature values
Classify them into clusters of items with similar situations using KNN
Regression prediction using logistic regression, polynomial regression and multilayer perceptron regression networks for each of our four outcomes of interest
Combining the prediction results of the three models leads to the final conclusion