ISBN: 978-602-5649-417

COMPARISON OF PREDICTION ACCURACY BETWEEN DECISION TREE, NAÏVE BAYES AND K-NN ON WEB PHISING

Sidharta[1], Albert Verasius Dian Sano[2]

[1] Computer Science Program, Bina Nusantara Institute of Creative Technology
Malang, Indonesia
sidharta@binus.ac.id

[2] Computer Science Program, Design and Technology, Bina Nusantara Institute of Creative Technology
Malang, Indonesia
avds@binus.ac.id

Abstract,

The objective of this research is to find the best performing predictive model in term of accuracy among three classification models on web phising dataset,i.e., decision tree, naive bayes, and k-NN. The dataset has 1353 examples and 9 regular attributes and a class attribute describing whether a website is phishy or not. Cross validation with 10 folds repetition is applied to each model for training and testing. Particular parameters that significantly affect the performance are set to get optimized for each model. The result of this study shows that the best performing predictive model is decision tree model.

Keywords: Decision Tree, Naïve Bayes, K-NN, Phising.

Introduction

According to a trend report from the Anti Phishing Working Group (APWG) in 2014 (Activity & Report, 2009), Phishing is a crime mechanism that implements a combination of technical and social engineering skills to steal identity data and credentials of a person's financial account. Phishers will try to get these credentials like usernames, passwords and credit card details masquerading as another trusted entity while trying to get internet user data over the web, email, messenger, during the interaction process between a user and a system.

The exponentially rising level of Internet connectivity from devices such as computers, smartphones, and other devices encourages Internet users to connect actively with countless organizations and systems around the world. The Internet becomes the largest place for its users to meet each other and share data. This is the basis for phishers to use the way through the Internet for various data as if it were a point of contact to run a widespread phishing activity by embedding malware onto a PC that would mislead users into a fake site page (MBAH, 2017). According to APWG reports from 2014 to 2015 (Activity & Report, 2009), the number of emails that contained unique phishing had a very sharp increase of 68270 emails in October 2014 to 106421 emails in September 2015 and this caused the phishing to be an interesting topic area for research.

This research applies predictive model prediction testing to a dataset containing 1353 websites through nine attributes that characterize phishing websites (Abdelhamid, Ayesh, & Thabtah, 2014). The nine attributes can be seen from the table below.

Table 1. List of website phising attributes

No.	Attribute	Description	Data Type	Data Content
1	SFH	Server Form Handler	Polynominal	Legiti-mate, suspi-cious, phishy
2	popUpWid now	Pop-up form	Polynominal	Legiti-mate, suspi-cious, phishy

3	SSLfinal_State	Identify HTML with Secure Sockets Layer	Polynominal	Legiti-mate, suspi-cious, phishy
4	Request_URL	Identify an external url in the domain	Polynominal	Legiti-mate, suspi-cious, phishy
5	URL_of_Anchor	Identify the <a> tag within website page	Polynominal	Legiti-mate, suspi-cious, phishy
6	web_traffic	Measure website popularity based on alexa	Polynominal	Legiti-mate, suspi-cious, phishy
7	URL_Length	Identify url length (based on 54 character)	Polynominal	Legiti-mate, suspi-cious, phishy
8	age_of_domain	Domain time	Binomi-nal	Legiti-mate, phishy
9	having_IP_Address	IP address usage in url	Binomi-nal	Legiti-mate, suspi-cious
10	Result	Label or calss in classification model	Polynominal	Legiti-mate, suspi-ciouc, phishy

Research methods

A. Decision Tree

Decision Tree is a classifier used to classify data using recursive techniques based on data attributes (Gorade, Deo, & Purohit, 2017). This model consists of many nodes and one root. A node will have a branch called edge after testing data based on certain criteria. Nodes that no longer have branches are called leaves. In the decision tree a node can have two or more branches.

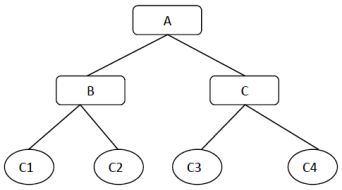

Figure 1. Decision tree

A is root, B and C are nodes, while C1, C2, C3, and C4 are leaves. In the decision tree, leaves are representations of a class or label.

B. Naïve Bayess

Naïve Bayes is a classification with probability and statistical methods proposed by the British scientist Thomas Bayes, predicting future opportunities based on past experience known as Bayes Theorem. Naïve Bayes for each decision class, calculate the probability on condition that the decision class is true, given the object

information vector (Olson & Delen, 2008). This algorithm assumes that the object attribute is independent. The probability involved in producing the final estimate is calculated as the number of frequencies from the "master" decision table (Olson & Delen, 2008).

Naive Bayes Classifier works very well compared to other classifier models (Xhemali, J. Hinde, & G. Stone, 2009). Naïve Bayes classifiers based on the Bayes theorem are probabilistic statistical classifier (Han, Kamber, & Pei, 2012), where the word "naïve" denotes conditional independence between features or attributes. The main advantage is simpler than any other classification algorithm that can handle datasets with a large number of attributes.

Naïve Bayesian classifier, or simple Bayesian classifier, has the following workings (Han et al., 2012) :

1. D is a training set and class labels. Each tuple is represented by an n-dimensional attribute vector, $X = (x_{1,}x_2, \dots, x_n)$, denoting n measurement of a tuple consisting of n attributes, respectively, $A_1, A_2, \dots, A_n$.
2. If there is class m, $C_1, C_2, \dots, C_m$. Given a tuple, X, the classifier would predict X belongs to the class with the highest posterior probability, conditioned on X. The Naïve Bayesian classifier predicts the tuple X including the C_i class if and only if $P(C_i|X) > P(C_j|X)$ for
   $1 \leq j \leq m, j \neq i$
   So that $P(C_i|X)$ needs to be maximized. C_i class that has $P(C_i|X)$ maximized is called maximum posteriori hypothesis. According to the Bayes theorem,
   $$P(C_i|X) = P(X|C_i)P(C_i)/P(X)$$
3. Since $P(X)$ is constant for all classes, only $P(X|C_i)P(C_i)$ needs to be maximized. If the prior probability class is not known, then it is assumed that each class has the same prior probability, ie $P(C_1) = P(C_2) = \cdots = P(C_m)$, so it only needs to maximize the value of $P(X|C_i)$. If otherwise, $P(X|C_i)P(C_i)$ is maximized. Prior probability classes can be calculated by $P(C_i) = |C_i, D|/|D|$, where $|C_i, D_j|$ is the number of training tuples that belong to the C_i class within the D dataset.
4. Datasets that have many attributes cause computational time $P(X|C_i)$ to be high. So as to reduce computing time in calculating $P(X|C_i)$, naive assumes class conditional independence, ie the values of the attribute are conditionally independent between one attribute with another attribute, if given the class label of the tuple. So:
   $$P(X|Ci) = \prod_{k=1}^{n} P(x_k|Ci)$$
   $$P(X|Ci) = P(x_1|Ci) \times P(x_2|Ci) \times \dots \times P(x_n|Ci)$$

C. K-NN

K-Nearest Neighbor (k-NN) is a classifier based on learning in training sample data. Each sample presents a data in n-dimensional space (Gorade et al., 2017). All sample data for training is stored in an n-dimensional pattern space. When an unknown sample is given, the k-nearest neighbor classifier will search for the pattern space in the previous training sample closest to the unknown sample. This proximity is defined by the measurement of Euclidean distance, where the Euclidean distance between two points, $X = (x_1, x_2, \dots, x_n)$ and $Y = (y_1, y_2, \dots, y_n)$ is expressed by $d(X, Y)$.

$$d(X, Y) = \sqrt{\sum_{i-1}^{n} (x_i + y_i)^2}$$

Classifier k-NN will give equal weight to each attribute. This classifier can be used for prediction on unknown samples.

This research generally covers three main stages: data acquisition phase, data preparation or data pre-processing, and data mining.

The data to be used is taken from the UCI Machine Learning Repository's historical data which can be downloaded from the UCI Machine Learning Repository - Web Phishing Dataset site page. This dataset contains 1353 samples and 10 attributes.

At the preparatory stage of data or pre-processing data used in this research will adopt some of the theories (Han et al., 2012) :

a. Data cleansing

This process will identify incomplete data and will then process incomplete data with a particular method accordingly.

b. Data selection
This process is to identify the attributes in the dataset that will be required by the models used, in this case the decision tree, naive bayes, and k-NN. Only the selected attributes will be inputs for the prediction models.

c. Data transformation
This process will identify and modify certain attributes to match the process that will be applied to the prediction model.

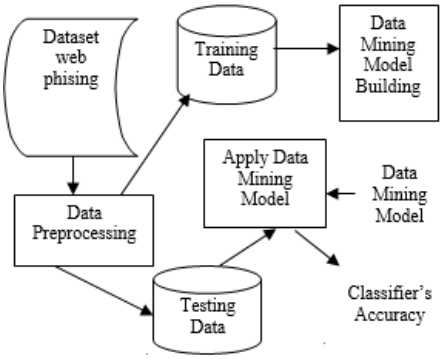

Figure 2. Block diagram of data mining model

Research Results and Discussion

A. Decision Tree

There are two optimized parameters on decision tree that significantly affect the performance i.e., criterion and minimal gain. Criterion has four different possible values on polynominal values dataset, that is :

1. Information gain
2. Gain ratio
3. Gain index
4. Accuracy

Minimal gain is set with minimum value to 0.01, maximum value to 100, and steps or loops to 100 times. The best combination of parameters found for decision tree is either combination between information gain and minimal gain equals to 0.109 or cobination between gain ratio and minimal gain equals to 0.030

Table 2. Parameter Optimization for Decision Tree

Iteration	Decision Tree (2).criterion	Decision Tree (2).minimal_gain	Accuracy ↓
42	information_gain	0.109	0.901
9	gain_ratio	0.030	0.901

The performance of accuracy is 90.1% with standard deviation +/- 2.35%.

Table 3. Confusion Matrix for Decision Tree

	True Suspi-cious	True Legitimate	True Phishy
Pred.Suspicious	96	9	11
Pred.Legi-timate	3	512	80
Pred.Phi-shy	4	27	611
Class recall	93.20%	93.43%	87.04%

B. Naïve Bayes

Since naive bayes model has no many attributes, then the only parameter set to be optimized is laplace correction which has two possible values, i.e., true or false. The best value found for laplace correction is 'true' and the best accuracy equals to 84.55% with a standard deviation +/- 1.82.

Table 4. Confusion Matrix for Naïve Bayes

	True Suspicious	True Legitimate	True Phishy
Pred.Suspicious	15	17	9

Pred.Legitimate	43	495	59
Pred.Phishy	45	36	611
Class recall	15.45%	90.33%	87.04%

C. K-NN

There are two parameters to be optimized for k-NN, i.e., the number of k and nominal measure. The number of k is set between 1 and 100 with steps or loops of 10 times. The possible values for nominal measure are :

1. nominal distance,
2. dice similarity,
3. jaccard similarity,
4. kulczynski similarity,
5. rogers tanimoto similarity,
6. russell rao similarity,
7. simple matching similarity.

The best combination of parameters is 11 for the number of k and russel rao similarity for nominal measure.

Table 5. Parameter Optimization for K-Nearest Neighbor

Iteration	k-NN (2).k	k-NN (2).nominal_measure	accuracy ↓
57	11	RusselRaoSimilarity	0.883

The best accuracy performance for k-NN is 88.32% with a standard deviation +/- 2.11.

Table 6. Confusion Matrix for K-Nearest Neighbor

	True Suspicious	True Legitimate	True Phishy
Pred.Suspicious	43	5	6
Pred.Legitimate	28	501	45
Pred.Phishy	32	42	651
Class recall	41.75%	91.42%	92.74%

Conclusions and recommendations

Based on the overall result of accuracy comparation amongst those three models, thus the best performing predictive model in term of accuracy on this (web phising) dataset is decision tree.

References

Abdelhamid, N., Ayesh, A., & Thabtah, F. (2014). Phishing detection based Associative Classification data mining. *Expert Systems with Applications*, *41*(13), 5948–5959. https://doi.org/10.1016/j.eswa.2014.03.019.

Activity, P., & Report, T. (2009). Phishing Activity Trends Report 4 Quarter. *Methodology*, (December).

Gorade, S. M., Deo, A., & Purohit, P. (2017). A Study of Some Data Mining Classification Techniques. *International Research Journal of Engineering and Technology (IRJET)*, 3112–3115.

Han, J., Kamber, M., & Pei, J. (2012). Data Mining: Concepts and Techniques. *San Francisco, CA, Itd: Morgan Kaufmann*. https://doi.org/10.1016/B978-0-12-381479-1.00001-0.

MBAH, K. F. (2017). a Phishing E-Mail Detection Approach Using Machine Learning.

Olson, D., & Delen, D. (2008). *Advanced Data Mining Techniques*. *Springer-Verlag*. https://doi.org/10.1017/CBO9781107415324.004.

Xhemali, D., J. Hinde, C., & G. Stone, R. (2009). Naive Bayes vs. Decision Trees vs. Neural Networks in the Classification of Training Web Pages. *International Journal of Computer Science*, *4*(1), 16–23. https://doi.org/1694-0814.