Anomaly Detection technique

GOAL

- The key idea is to detect behaviors that are abnormal when compared to normal behaviour. ***Crux of this goal is that it can have multiple normal standard*** i.e, there are some peak cases where it appears to be abnormal behaviour but it isn't it is one of such normal behaviour. We have to identify them to properly classify them.

- Specifically, our approach is to

  (1) Identify some traffic parameters, which can be used to describe the network traffic and that vary significantly from the normal behavior to the anomalous one

  (2) Implementing the Random Forest Classifier with above traffic parameters.

Example for multiple normal behaviour

- paths in P4 program that invoke switch CPU are typically expensive and not seen often. However, if we suddenly observe too many packets taking this path, we can consider this abnormal behavior.

Literature review

1. Chi-square – using Packet Execution Paths

We build expected distribution, say E, from periodically collected per-path statistics

$$\chi^2_{w(i)} = \sum_{j=1}^{k} \left(\frac{(E_{w(i)}(j) - O_{w(i)}(j))^2}{E_{w(i)}(j)} \right)$$

w(i) be a window time

Let there be k paths in w(i) time window

Ow(i) is observed i^th distribution

- But in chi square stats we can only have one normal behaviour

- **current anomaly-based IDS** models classify by analyzing features,utilizing a number of different models. A final decision is based on results of all models used.
- The main problem of these models is high number of False positive rate.

Cause of above problem:

1. One is the simplistic aggregation of model outputs in the decision phase.

2. Lack of additional information which would help in identifying exceptional normal behaviors

2. Bayesian Event Classification for Intrusion Detection

To overcome these problems(mentioned in prev slide) we introduce a classification based on **Bayesian networks**.

- It improves the aggregation of different model outputs and allow one to smoothly incorporate additional information.
- In Bayesian network, each node contains the states of the random variable that represents conditional probability.

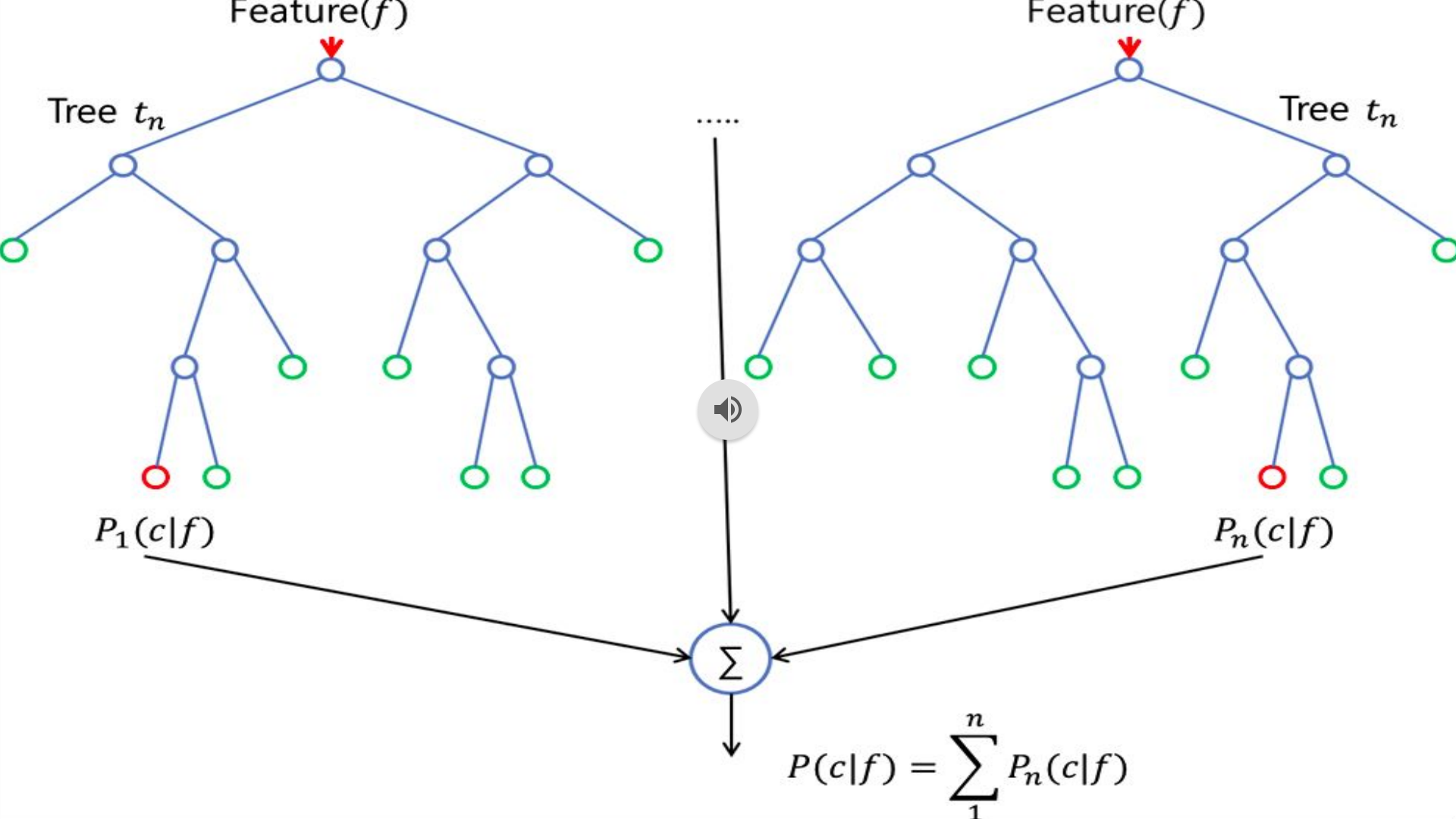
- In Bayesian networks we can model, inter-model dependencies and to integrate additional information such as model confidence (i.e., dropping the restriction of at most a single parent node)
- We add casualties between the nodes to indicate the dependencies.
- The model confidence is represented as one of 5 discrete levels: very high, high, medium, low and none.

System design

- Each model of M analyzes one or multiple features of a given input event and compares the event's feature(s) to the model's previously established profile (i.e., the description that specifies the normal features or properties).
- For threshold based IDS each of o(i) is added and concluded if anomaly/not
- For bayesian based system we add appropriate link b/w each child nodes to show the casual relationship
- Depending on the input events that are utilized for establishing the profile, a certain feature might not be very suitable to distinguish between attacks and regular behavior. It might be the case that the same values of a feature appear in both regular behavior and attacks or that the variance of a feature is very high.

Random Forest Classifier

- Random forest is an ensemble tool which takes a subset of observations and a subset of variables to build a decision trees.
- It builds multiple such decision tree and them merge together to get a more accurate and stable prediction.
- RFC is an supervised learning


Feature(f)
Tree t_n
.....
Feature(f)
Tree t_n
$P_1(c|f)$
$P_n(c|f)$
$\sum$
$P(c|f) = \sum_1^n P_n(c|f)$

Implementing random forest on CICIDS 2017 Dataset

Preprocessing:

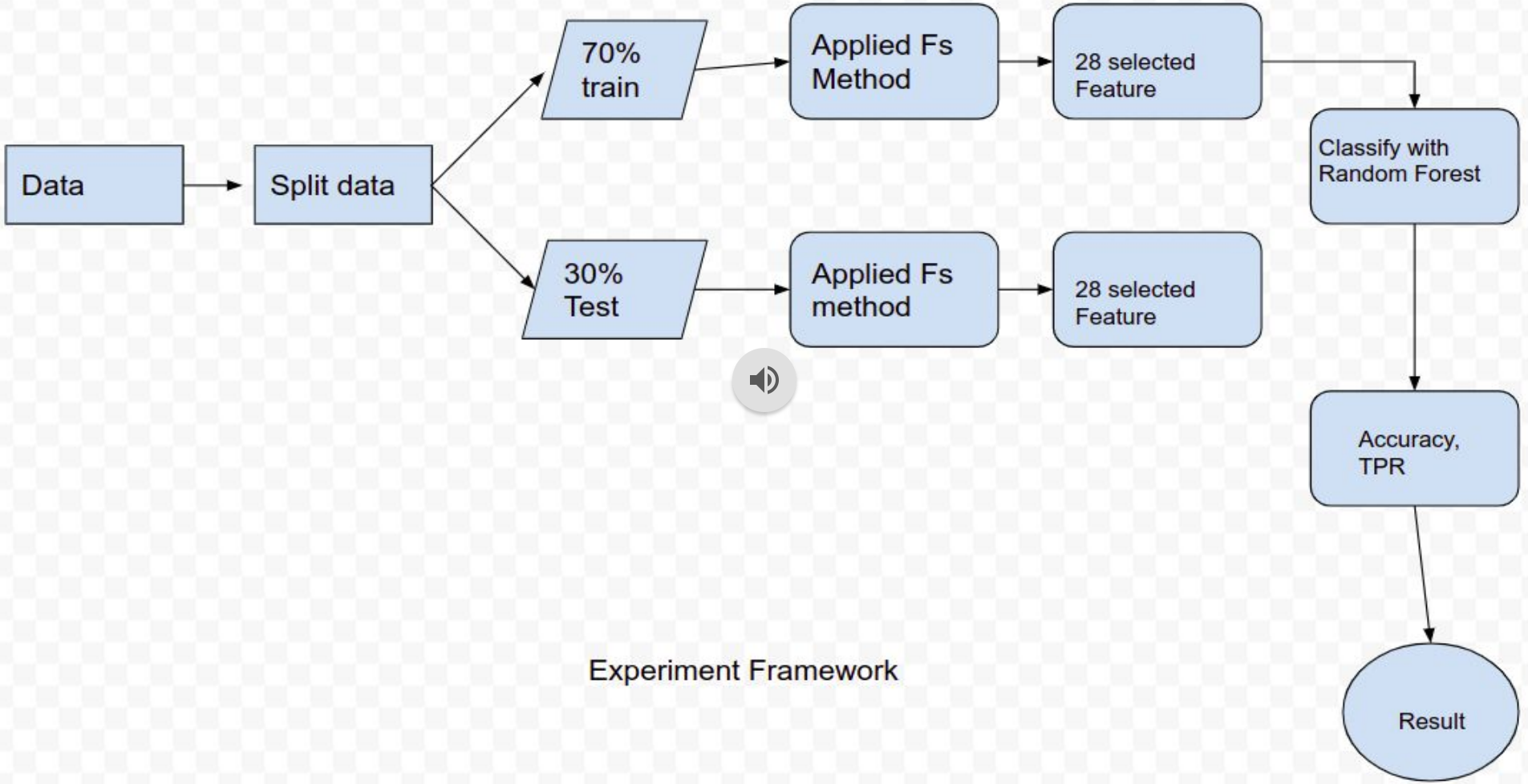
- There were 88 features in this dataset. Some of them don’t contribute to the decision making process so we have to remove them.
- After this steps there were around 77 features left.
- We clean the dataset by removing some unnecessary values and fill the nan values whenever required within column.

Feature selection

- Feature selection is carried out using Information Gain Based on previous research, this approach produces 28 features that result in ideal detection performance. Furthermore, the features of this selection are used to identify attacks on imbalanced datasets.

Experiment Configuration

- Use Training set: classification performance test using all input data
- Cross Validation: classification performance test using k-fold cross-validation. In the experiment, 10- fold and 5-fold cross-validation were used.
- Presentage Split: classification performance test using split data. The experiment use 30 to 70 splits



Experiment Framework

Detection Performances

- To test whether the features generated by the proposed method can be used to detect normal or attacks traffics on high-dimensional and imbalance data, validation is carried out.
- This detection test uses the Random Forest classification algorithm. Experiment results show very high accuracy as selected features are more and they are relevant and important in characterizing the attacks patterns, thus the classification algorithms are able to identify very well the attacks.
- In addition, to maintain the reliability of the test results, several testing modes were used, i.e.: the use of full train, 5-fold cross-validation, 10-fold cross-validation, and 30-70% data splitting which were applied to training data and testing data.

Results

The accuracy of Random Forest algorithm in predicting traffic is presented in Table. The experimental results also show the accuracy of Random Forest algorithm which is excellent with an average accuracy value of 99.842% for training data and 99.797% for testing data.

Testing Mode	Training	Testing
5-fold	99.92	99.81
10-fold	99.86	99.84
30 split	99.80	99.76
40-split	99.79	99.78

Code link

Click here to access

Further work

- Working on Real time traffic dataset.
- Applying the model to Blink system , Bloom filter.
- Experiment the above systems with unsupervised learning.