Applications of Artificial Intelligence in Cyber Security

Elmurad Abbasov
University of Minnesota Morris
Computer Science Senior Seminar
November 13, 2021

Outline

- **Background information**
  - Cyber Security in the Modern World
  - The Usage of Artificial Intelligence in Cyber Security
  - Intrusion Detection System
  - Machine Learning Overview & Specifics
- Methodology
  - NSL-KDD (Network Security Laboratory - Knowledge Discovery in Databases)
  - Experimental setup
- Results Evaluation
  - Statistical summary
- Conclusion

Cyber Security in the Modern World

- 10.5 billion malware attacks since 2018
- 7.9 billion data breaches around the world in 2019
    - (112 percent more data breaches than in 2018)
    - Data breach - a security violation in which data is manipulated without a permission
- It is predicted that worldwide cyber security spending will reach $133.7 billion by 2022
- The number of cyber attacks is increasing every day

The Usage of Artificial Intelligence in Cyber Security

- Why cyber security is important?
- How AI is used in cyber security?
- Why research “Comparative Analysis of ML Classifiers for Network Intrusion Detection” by Ahmed M. Mahfouz, Deepak Venugopal, and Sajjan G. Shiva is important?

Intrusion Detection System (IDS)

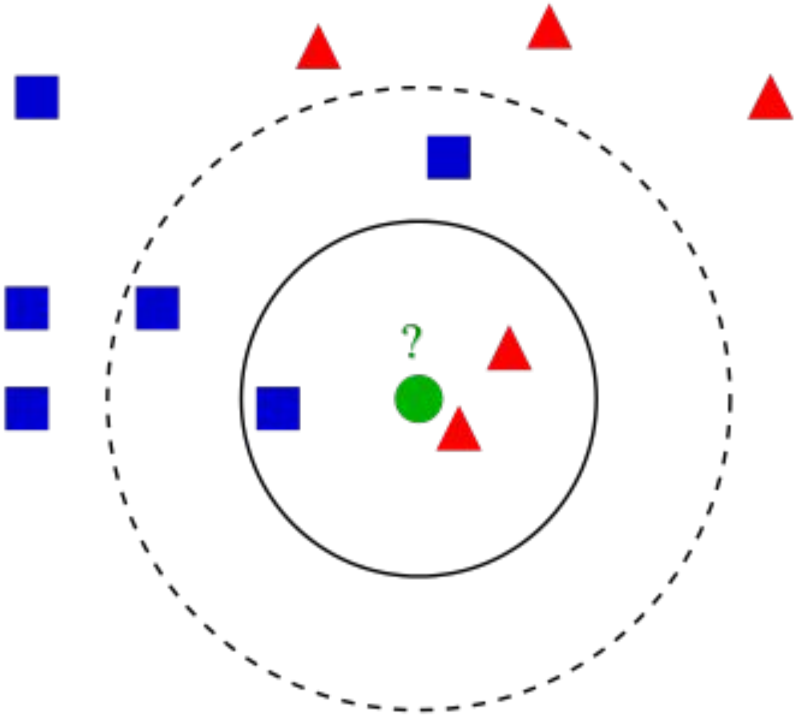
- IDS overview
  - Intrusion is an act of entering a virtual space without a proper permission
  - An IDS is software that is searching for malware in the entire network
- Types of IDS
  - *Signature-based detection*
    - Searches for patterns and compares with predetermined attack types (signatures)
  - *Statistical anomaly-detection*
    - IDS detects a suspicious traffic and compares to an established baseline
    - Usually a dataset of “normal” and “attack” files is used

Machine Learning Overview

- Supervised Learning (already labeled data used for predictions)
  - Labeled - group of samples tagged with a “tag”, “label”, or “class”
  - Prediction - output of an algorithm after it has been trained and applied to new data
  - Paired input records and their desired output
  - The output of a classification problem is a category - “normal” or “attack”
- WEKA: Naive Bayes, Logistic, MultilayerPerception, SMO, IBK and J48
- Original: Naive Bayes, Logistic, ANN, SVM, KNN, DT C 4.5

KNN (*k*-nearest neighbors) [7]

- Uses “majority voting” principle
- An object is classified by the majority vote of its neighbors
- Based on a distance function that measures the difference/similarity between two instances
- If k=1, then the object is assigned to the class of the single nearest neighbor
- The neighbors are taken from a set of objects for which the class is known



https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

KNN (*k*-nearest neighbors) [7]

- The standard Euclidean distance d(x, y) between two instances x and y is defined in the following figure.
- **Xi** is the feature element of X, **Yi** is the feature element of Y, **n** is the total number of features in the data set.

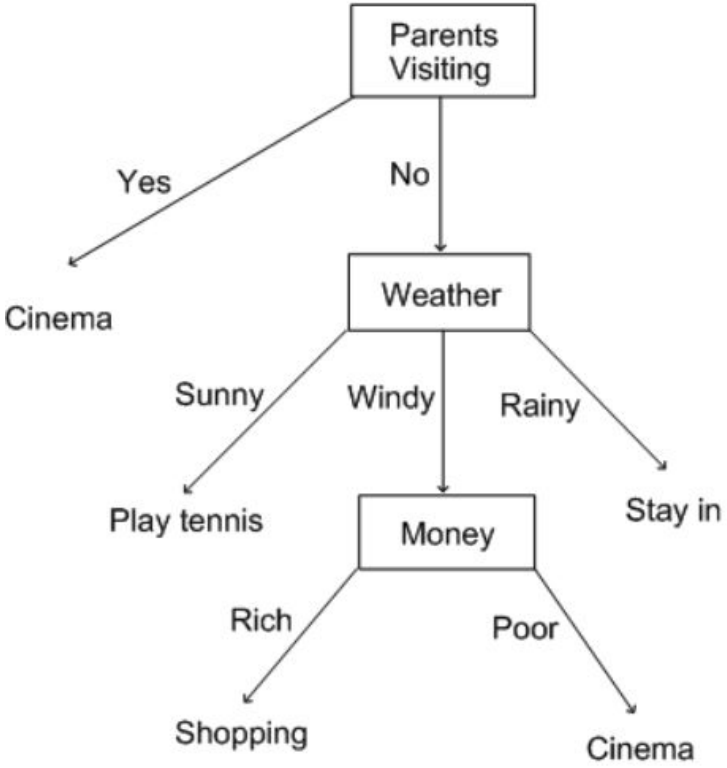
$$d(x,y)=\sqrt{\sum_{i=1}^{n}(x_i-y_i)^2}$$

KNN (*k*-nearest neighbors) [7]

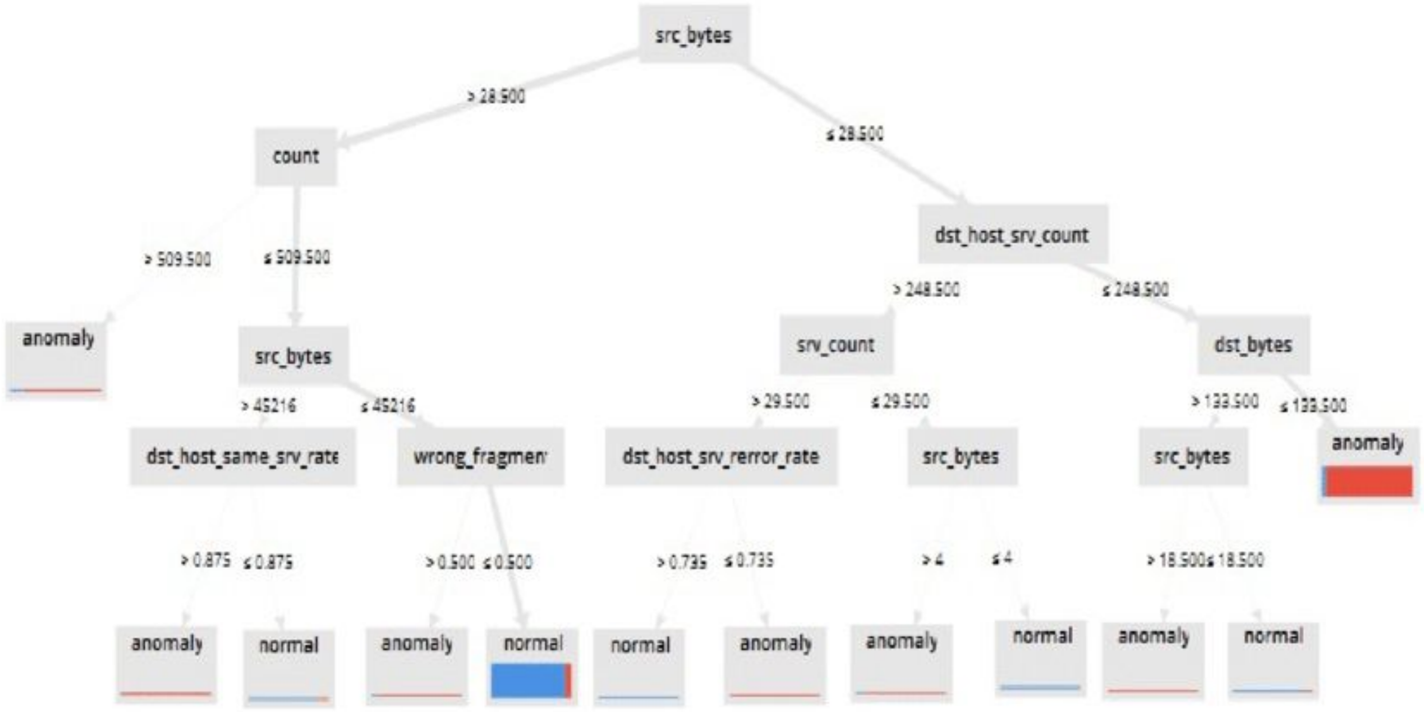
- Load the data to the model
- Choose k value as the number of neighbors
- Calculate distance between the sample and its neighbors
- Store the distance and sort in ascending order
- List out the first k entries
- Assign a class based on the majority present in the neighbor points

Decision Tree C 4.5 [9]

- Splits data recursively into subsets so that each subset contains more or less homogeneous states of target variable
- When the recursive process is completed, a DT is formed which can be converted in simple If - Then rules
- Uses **Information Gain (IG)** and **Entropy**
- **IG** - a measure of how much information a feature provides about a class and helps to determine order of features in the nodes
- **Entropy** - measures uncertainty in observations (probability of an event happening) and determines how a DT chooses to split data
- IG is inversely proportional to entropy



Decision Tree C 4.5 NSL-KDD Example [8]



Outline

- Background information
  - Cyber Security in the Modern World
  - The Usage of Artificial Intelligence in Cyber Security
  - Intrusion Detection System
  - Machine Learning Overview & Specifics
- **Methodology**
  - Data set (NSL-KDD)
  - Experimental setup
- Results Evaluation
  - Statistical summary
- Conclusion

NSL-KDD: Overview

- Currently a benchmark in the research of IDS
- Contains malware connections (attacks) and safe connections (normal)
- 42 features per records
  - 41 features are about traffic input (data packets traveling across the internet)
  - The other feature is a label of either a safe connection or a threat connection
- Includes both training and testing sets

NSL-KDD: Features [6]

- For ML model to successfully process the data, it has to be in numerical values.
- Not all features all numerical (protocol_type, service, etc.), but all must be converted to numerical values.
- **Logged_in** = If logged in then logged_in = 1, else 0
- **Root_shell** = If root shell is obtained then root_shell = 1, else 0
- **Is_guest_login** = If login as guest then is_guest_login = 1, else 0
- **Count No.** = number of connections to the same host in last 2 seconds

#	Feature	#	Feature
1	duration	22	is_guest_login
2	protocol_type	23	Count
3	service	24	srv_count
4	flag	25	serror_rate
5	src_bytes	26	srv_serror_rate
6	dst_bytes	27	rerror_rate
7	land	28	srv_rerror_rate
8	wrong_fragment	29	same_srv_rate
9	urgent	30	diff_srv_rate
10	hot	31	srv_diff_host_rate
11	num_failed_logins	32	dst_host_count
12	logged_in	33	dst_host_srv_count
13	num_compromised	34	dst_host_same_srv_rate
14	root_shell	35	dst_host_diff_srv_rate

NSL-KDD: Example Data [6]

```
0,tcp,http,SF,181,5450,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.
00,0.00,0.00,9,9,1.00,0.00,0.11,0.00,0.00,0.00,0.00,0.00,normal.
0,tcp,http,SF,239,486,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.0
0,0.00,0.00,19,19,1.00,0.00,0.05,0.00,0.00,0.00,0.00,0.00,normal.
```

Figure 2. Original samples from NSL-KDD dataset.

```
0,3,19,10,181,5450,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0,0,0,0,1,0,0,9,9,1,0,0.1
1,0,0,0,0,22
0,3,19,10,239,486,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0,0,0,0,1,0,0,19,19,1,0,0.
05.0.0.0.0.22
```

Figure 3. Results after data transformation.

NSL-KDD: Attack Classes

- Denial of Service (DoS)
  - Overloads a server with abnormal traffic that shuts down the connection to and from the target system
- Probe
  - Extracting specific personal information from the target system
- Remote to Local (R2L)
  - Gains local access to a remote machine
- User to Root (U2R)
  - Gains root access to the interested system or a network

NSL-KDD: Attack Classes Summary in Table 2 [5]

Table 2. No of samples for normal and attack classes.

Class	Training Set	Occurrences Percentage	Testing Set	Occurrences Percentage
Normal	67343	53.46 %	9711	43.08 %
DoS	45927	36.46 %	7460	33.08 %
Probe	11656	9.25 %	2421	10.74 %
R2L	995	0.79 %	2885	12.22 %
U2R	52	0.04 %	67	0.89 %
Total	125973	100.0 %	22544	100.0 %

NSL-KDD: Imbalance Issue

- An imbalance in the dataset creates biased results toward the samples from the majority classes
- The classification accuracy is higher for the majority classes than for minority classes
- The researchers offer a method to deal with the imbalance

Experimental Setup: Overview

- First phase:
  - Compare classifiers with default settings and original data set
- Second phase:
  - NSL-KDD was modified to reduce its dimension
- Third phase:
  - NSL-KDD was modified to solve imbalance issue

Experimental Setup: First Phase

- First phase:
  - Compare classifiers with default settings
  - Default data set without modifications
  - Cross-Validation of 10-folds [3]
    - Used for evaluating and comparing ML models
    - Works by separating the dataset into K equally sized folds
    - K-1 folds used to train the model, the last fold is left for model testing
    - Process reiterated until every fold gets the chance to act as the test dataset
    - The capability of the model is estimated by averaging the performance measures across all folds

Experimental Setup: Second Phase

- Second phase:
  - NSL-KDD was modified to reduce its dimension (transformation of data from a high-dimensional space into a low-dimensional space to keep only meaningful properties) [1]
    - Feature selection process (selecting a subset of the original features so that the feature space is optimally reduced to the evaluation criteria) done with InfoGainAttributeEval algorithm
      - Evaluates the worth of a feature by measuring the IG with respect to the class
      - The algorithm measured how each feature contributes in decreasing the overall entropy
    - Selected 14 out of 41 features
  - Hyperparameter optimization (the process of choosing a set of optimal hyperparameters) is done by CVParameterSelection
    - Hyperparameter - a parameter whose value is used to control the learning process [2]
    - Performs parameter selection by cross-validation

Experimental Setup: Phase Three

- Third phase:
    - NSL-KDD was modified to solve imbalance issue
        - Under-sampling the dominant classes [4]
            - WEKA’s Resample filter that takes a random subsample
            - Uses either sampling with replacement or without replacement
        - Over-sampling the minority classes [4]
            - WEKA’s (Synthetic Minority Over-sampling Technique) SMOTE filter that generates synthetic instances
            - As a result increases the minority group

Outline

- Background information
  - Cyber Security in the Modern World
  - The Usage of Artificial Intelligence in Cyber Security
  - Intrusion Detection System
  - Machine Learning Overview & Specifics
- Methodology
  - Data set (NSL-KDD)
  - Experimental setup
- **Results Evaluation**
  - Statistical summary
- Conclusion

Results Evaluation: Overview & Parameters

- Parameters such as **TP**, **TN**, **FP**, **FN** are commonly used in Machine Learning in evaluating results.
- **True Positive (TP)** - an outcome where the model correctly predicts the positive class (malware was identified as a threat)
- **True Negative (TN)** - an outcome where the model correctly predicts the negative class (a clean file was identified as a non-threat)
- **False Positive (FP)** - an outcome where the model incorrectly predicts the positive class (a clean file was identified as a threat)
- **False Negative (FN)** - an outcome where the model incorrectly predicts the negative class (a malware was identified as a non-threat)

Results Evaluation: ML Efficiency Metric

- **Accuracy** - the number of correct predictions divided by the total number of predictions

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

Test Set Results [5]

Phase 1

Classifier	Accuracy
NB	76.12 %
Logistic	75.60 %
MLP	77.60 %
SMO	75.39 %
IBK	79.35 %
J48	81.69 %

Phase 2

Classifier	Accuracy
NB	78.15 %
Logistic	81.51 %
MLP	78.15 %
SMO	79.83 %
IBK	84.35 %
J48	82.67 %

Comparison of Classifiers in Table 8 [5]

Table 8. Classifiers accuracy detection for different classes of attacks.

Classifier	Class	Phase I	Phase II	Phase III
IBK	Normal	79.3 %	86.8 %	99.4 %
	DoS	80.5 %	90.7 %	99.5 %
	Probe	71.8 %	76.2 %	99.0 %
	R2L	00.0 %	00.0 %	53.2 %
	U2R	00.0 %	00.0 %	41.5 %
J48	Normal	81.6 %	84.8 %	99.5 %
	DoS	80.1 %	89.2 %	99.2 %
	Probe	67.9 %	63.2 %	91.6 %
	R2L	18.9 %	18.2 %	55.1 %
	U2R	00.0 %	00.0 %	39.3 %

Conclusion

- Six different classifiers were evaluated on their performance to detect cyber attacks on the NSL-KDD data set
- KNN (IBK) and DT C 4.5 (J48) showed good performance comparing to other algorithms
- Imbalance mitigation method improved limitations in detecting R2L and U2R attacks

Acknowledgements

Thank you to my advisor Nic McPhee and my instructor Elena Machkasova for their feedback, support and advice.

Questions?

References

[1] Wikipedia contributors. (2021, August 26). Dimensionality reduction. In Wikipedia, The Free Encyclopedia. Retrieved 10:54, November 13, 2021, from https://en.wikipedia.org/w/index.php?title=Dimensionality_reduction&oldid=1040808431

[2] Wikipedia contributors. (2021, August 8). Hyperparameter optimization. In Wikipedia, The Free Encyclopedia. Retrieved 11:06, November 13, 2021, from https://en.wikipedia.org/w/index.php?title=Hyperparameter_optimization&oldid=1037728107

[3] Wikipedia contributors. (2021, October 28). Cross-validation (statistics). In Wikipedia, The Free Encyclopedia. Retrieved 11:15, November 13, 2021, from https://en.wikipedia.org/w/index.php?title=Cross-validation_(statistics)&oldid=1052330597

[4] Wikipedia contributors. (2021, October 26). Oversampling and undersampling in data analysis. In Wikipedia, The Free Encyclopedia. Retrieved 11:21, November 13, 2021, from https://en.wikipedia.org/w/index.php?title=Oversampling_and_undersampling_in_data_analysis&oldid=1051894978

[5] Mahfouz, Ahmed & Venugopal, Deepak & Shiva, Sajjan. (2019). Comparative Analysis of ML Classifiers for Network Intrusion Detection.

[6] Harb, Hany & Zaghrot, Afaf & Gomaa, Mohamed & S. Desuky, Abeer. (2011). Selecting Optimal Subset of Features for Intrusion Detection Systems. Advances in Computational Sciences and Technology. 4. 179-192.

[7] Wikipedia contributors. (2021, October 24). K-nearest neighbors algorithm. In Wikipedia, The Free Encyclopedia. Retrieved 12:19, November 13, 2021, from https://en.wikipedia.org/w/index.php?title=K-nearest_neighbors_algorithm&oldid=1051590352

[8 Hassannataj Joloudari, Javad & Haderbadi, Mojtaba & Mashmool, Amir & Ghasemigol, Mohammad & Band, Shahab & Mosavi, Amir. (2020). Early Detection of The Advanced Persistent Threats Attacks Using Performance Analysis of Deep Learning. 10.20944/preprints202007.0745.v1.]

[9] Saha, S. (2018, November 16). What is the C4.5 algorithm and how does it work? Medium. Retrieved November 13, 2021, from https://towardsdatascience.com/what-is-the-c4-5-algorithm-and-how-does-it-work-2b971a9e7db0.