DETECTION AND PREVENTION OF CLICK FRAUD IN ONLINE ADVERTISING

Mrs. G. Mariammal[1], Parkavi G[2], Monisha M[3], Nivetha M[4], Nandhini C[5]

[1]Asst.Professor, Department Of Computer Science Engineering, Psna College Of Engineering And Technology, Dindigul

[2,3,4,5]Department Of Computer Science And Engineering, Psna College Of Engineering And Technology, Dindigul

ABSTRACT

Recent research has revealed an alarming prevalence of click fraud in online advertising systems. In this project, after investigation of different known categories of Web-bots along with their malicious activities and associated threats, we distinguish between the important behavioral characteristics of bots versus humans in conducting click fraud with in modern-day ad platforms performance in terms of accuracy and prediction-recall rate. Subsequently, we provide an overview of the current detection and threat mitigation strategies pertaining to click fraud. The proposed algorithm is tested by extensive experiments using real-world data. Compared with the state-of-art machine learning algorithms, our model can achieve significant.

Keywords: Click Fraud, Cnn Algorithm, Web Bots, Online Advertising, Cnn Classifier

1. INTRODUCTION

With the rapid growth of the Internet, online advertisement plays a more and more important role in the advertising market. One of the current and widely used revenue models for online advertising involves charging for each click based on the popularity of keywords and the number of competing advertisers. This pay-per-click model leaves room for individuals or rival companies to generate false clicks (i.e., click fraud), which pose serious problems to the development of healthy online advertising market. To detect click fraud, an important issue is to detect duplicate clicks over decaying window models, such as jumping windows and sliding windows. Decaying window models can be very helpful in defining and determining click fraud. However, although there are available algorithms to detect duplicates, there is still a lack of practical and effective solutions to detect click fraud in pay-per-click streams over decaying window models. In this paper, we address the problem of detecting duplicate clicks in pay-per-click streams over jumping windows and sliding windows, and are the first that propose two innovative algorithms that make only one pass over click streams and require significantly less memory space and operations. In real-time systems and business applications, advertisement data is usually of gigantic volumes. Since there are multiple features and the number of clicks is growing by the minute, there is a drastic increase in the feature space dimension, and models that predict click fraud must be trained periodically to keep up with the latest attacks.

2. MACHINE LEARNING

Machine learning is a branch of artifical intelligent(AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.IBM has a rich history with machine learning. One of its own, Arthur Samuel, is credited for coining the term, “machine learning” with his research (PDF, 481 KB) (link resides outside IBM) around the game of checkers. Robert Nealey, the self-proclaimed checkers master, played the game on an IBM 7094 computer in 1962, and he lost to the computer. Compared to what can be done today, this feat almost seems trivial, but it’s considered a major milestone within the field of artificial intelligence. Over the next couple of decades, the technological developments around storage and processing power will enable some innovative products that we know and love today, such as Netflix’s recommendation engine or self-driving cars.Machine learning is an important component of the growing field of data science. Through the use of statistical methods, algorithms are trained to make classifications or predictions, uncovering key insights within data mining projects.

3. EXISTING SYSTEM

Click seems promising in detecting fraudulent clicks at various rates of attack, it seem it may be possible for an adversary to create a method that is able to bypass Click detection system. It may be difficult to keep track of malicious clicks with IP aggregation and churn. As some networks may try to avoid exposing IP addresses, it makes attributing fraudulent clicks to a certain source more difficult.

4. DRAWBACKS OF EXISTING SYSTEM

- It will take time to load all the dataset.
- Process is not accuracy.
- It will analyze slowly
- Less efficiency.
- Poor discriminatory power

5. PROPOSED SYSTEM

Machine learning can help companies use that data to make predictions. For example, if a company has collected data regarding A Proposed method of detecting fraudulent clicks that does not rely on a threshold based defense.By taking advantage that click fraud often uses previous user traffic in order to fabricate fake clicks, Click is able to identify click fraud even though the attacker may be imitating organic clicks.

Advantages of the Proposed System

- Low complexity and better flexibiliy in classification
- Descriptors provide local invariant features
- It provides better accuracy
- High efficiency.

6. HARDWARE & SOFTWARE SPECIFICATION

Hardware Specification

The hardware requirements may serve as the basis for a contract for the implementation of the system and should therefore be a complete and consistent specification of the whole system. They are used by software engineers as the starting point for the system design.

Windows 7,8,10 64 bit

RAM 4GB

Software Specification

The software requirements document is the specification of the system. It should include both a definition and a specification of requirements. It is a set of what the system should do rather than how it should do it. The software requirements provide a basis for creating the software requirements specification. It is useful in estimating cost, planning team activities, performing tasks and tracking the teams and tracking the team's progress throughout the development activity.

Data Set

Python 2.7

Anaconda Navigator

7. METHODOLOGY

- ➢ Data Collection
- ➢ Data Pre-Processing
- ➢ Feature Extraction
- ➢ Evaluation Model

Data Collection. Data used in this paper is a set of product reviews collected from web Attacks records. This step is concerned with selecting the subset of all available data that you will be working with. ML problems start with data preferably, lots of data (examples or observations) for which you already know the target answer. Data for which you already know the target answer is called labelled data. Data collection breaks down into two methods. As a side note, many terms, such as techniques, methods, and types, are interchangeable and depending on who uses them. One source may call data collection techniques "methods," for instance. But whatever labels we use, the general concepts and breakdowns apply across the board whether we're talking about marketing analysis or a specific research project.

Data Pre-Processing: Organize your selected data by formatting, cleaning and sampling from it.

Three common data pre-processing steps are:

Formatting: The data you have selected may not be in a format that is suitable for you to work with. The data may be in a relational database and you would like it in a flat file, or the data may be in a proprietary file format and you would like it in a relational database or a text file.

Cleaning: Cleaning data is the removal or fixing of missing data. There may be data instances that are incomplete and do not carry the data you believe you need to address the problem. These instances may need to be removed. Additionally, there may be sensitive information in some of the attributes and these attributes may need to be removed from the data entirely.

Sampling: There may be far more selected data available than you need to work with. More data can result in much longer running times for algorithms and larger computational and memory requirements. You can take a smaller representative sample of the selected data that may be much faster for exploring and prototyping solutions before considering the whole dataset.

Feature Extraction:

Next thing is to do Feature extraction is an attribute reduction process. Unlike feature selection, which ranks the existing attributes according to their predictive significance, feature extraction actually transforms the attributes. The transformed attributes, or features, are linear combinations of the original attributes. Finally, our models are trained using Classifier algorithm. We use classify module on Natural Language Toolkit library on Python. We use the labelled dataset gathered. The rest of our labelled data will be used to evaluate the models. Some machine learning algorithms were used to classify pre-processed data. The technique of extracting the features is useful when you have a large data set and need to reduce the number of resources without losing any important or relevant information. Feature extraction helps to reduce the amount of redundant data from the data set. In the end, the reduction of the data helps to build the model with less machine effort and also increases the speed of learning and generalization steps in the machine learning process.

Practical Uses of Feature Extraction

Auto encoders: The purpose of auto encoder is unsupervised learning of efficient data coding. Feature extraction is used here to identify key features in the data for coding by learning from the coding of the original data set to derive new ones.

Bag-of-Words: A technique for natural language preprocessing that extracts the words (features) used in a sentence, document, website, etc. and classifies them by frequency of use. This technique can also be applied to image processing.

Evalution model

Model Evaluation is an integral part of the model development process. It helps to find the best model that represents our data and how well the chosen model will work in the future. Evaluating model performance with the data used for training is not acceptable in data science because it can easily generate overoptimistic and over fitted models. There are two methods of evaluating models in data science, Hold-Out and Cross-Validation. To avoid over fitting, both methods use a test set (not seen by the model) to evaluate model performance.

Accuracy measures how often the classifier makes the correct predictions, as it is the ratio between the number of correct predictions and the total number of predictions.**Precision** measures the proportion of predicted Positives that are truly Positive. Precision is a good choice of evaluation metrics when you want to be very sure of your prediction. For example, if you are building a system to predict whether to decrease the credit limit on a particular account, you want to be very sure about the prediction or it may result in customer dissatisfaction.The **confusion matrix** (or confusion table) shows a more detailed breakdown of correct and incorrect classifications for each class. Using a confusion matrix is useful when you want to understand the distinction between classes, particularly when the cost of misclassification might differ for the two classes, or you have a lot more test data on one class than the other. For example, the consequences of making a false positive or false negative in a cancer diagnosis are very different.

8. SYSTEM ARCHITECTURE

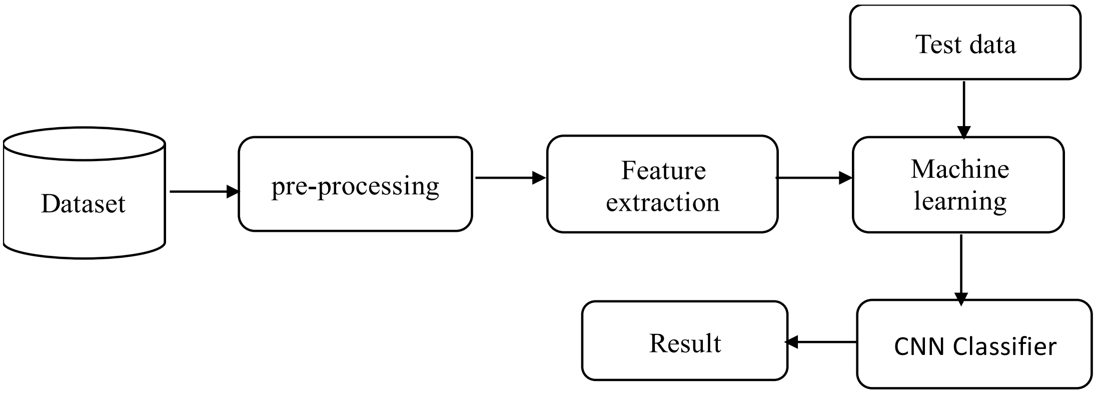
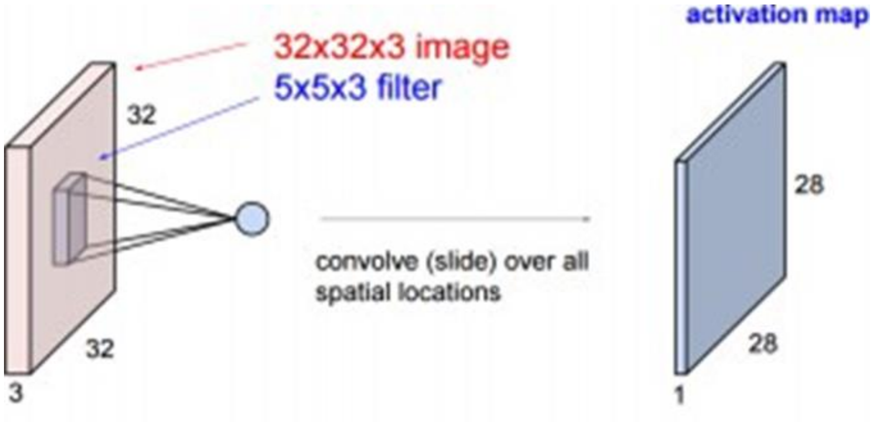


Figure 1: System architecture diagram

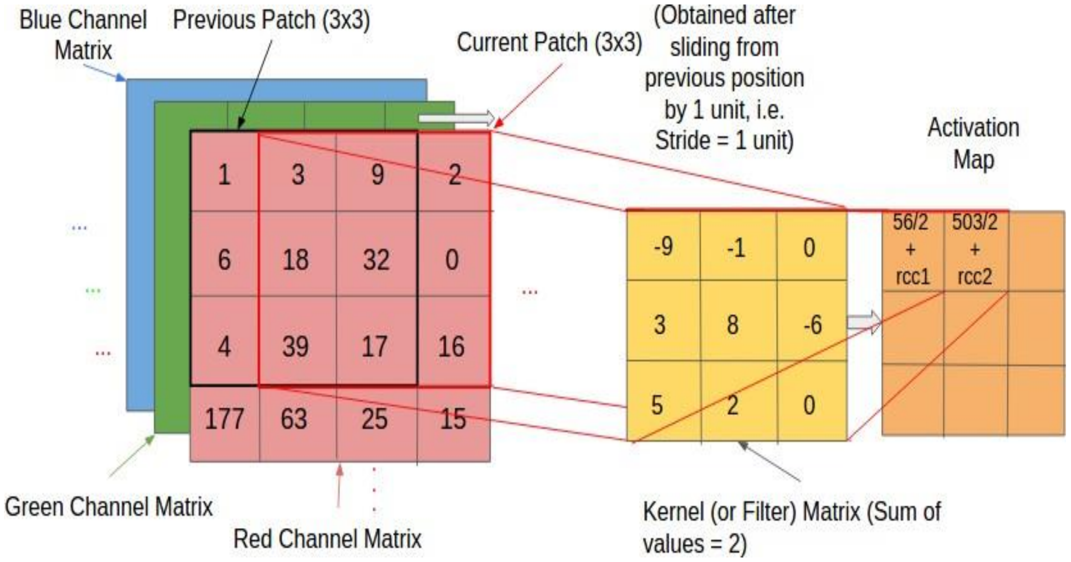
9. ALGORITHM

Convolution Neural Network (CNN)

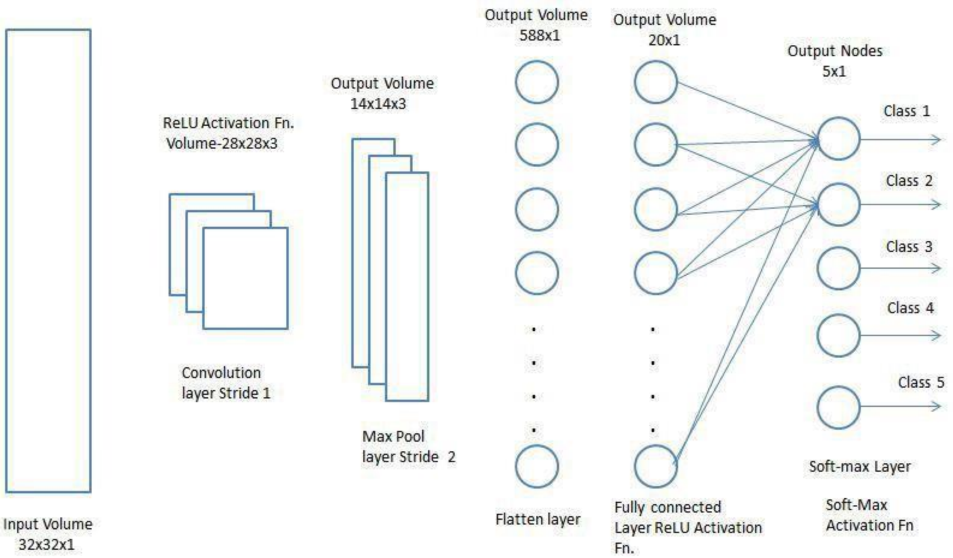
A Convolution Neural Network (ConvNet/CNN) is a deep neural networks method that can accept an image as input, give priority to numerous perspectives in the image (learnable weights and biases), and differentiate between them. The amount of pre-processing requiredby a ConvNet is far less than that required by other classification methods. Despite the fact that filters are hand-engineered in rudimentary processes, ConvNets can learn these filters/characteristics with sufficient training. The Visual Cortex organization affected the layout of a ConvNet, which is akin to that of Neurons as in Human Cognitive connectivity pattern. Neural networks only send signals in a restricted area of the peripheral vision known as the Receptive Field. A group of such sectors spans to embrace the full visual zone. This layer entails scanning the entire image for similarities and converting the results into a 3x3 matrix. Kernel is the name given to the image's binarized feature vector. The weight vector is the name given to each element in the kernel.



The Pooling division is in charge of shrinking the Convolved Format's spatial size. This is done to lower the amount of processing power process data using dimension reduction. It can also be used to remove rotational and temporal affine dominant traits while keeping the model's training loop intact. There are two kinds of pooling: maximum and average. Max-Pooling returns the whole amount of the area of the image held by the Kernel. On the other hand, Average-Pooling delivers an aggregate of all the data from the picture portion shelteredby the Kernel. Max Pooling also functions as a noise reducer. It completely eliminates noisy events and even de-noises along with dimensionality improvements.

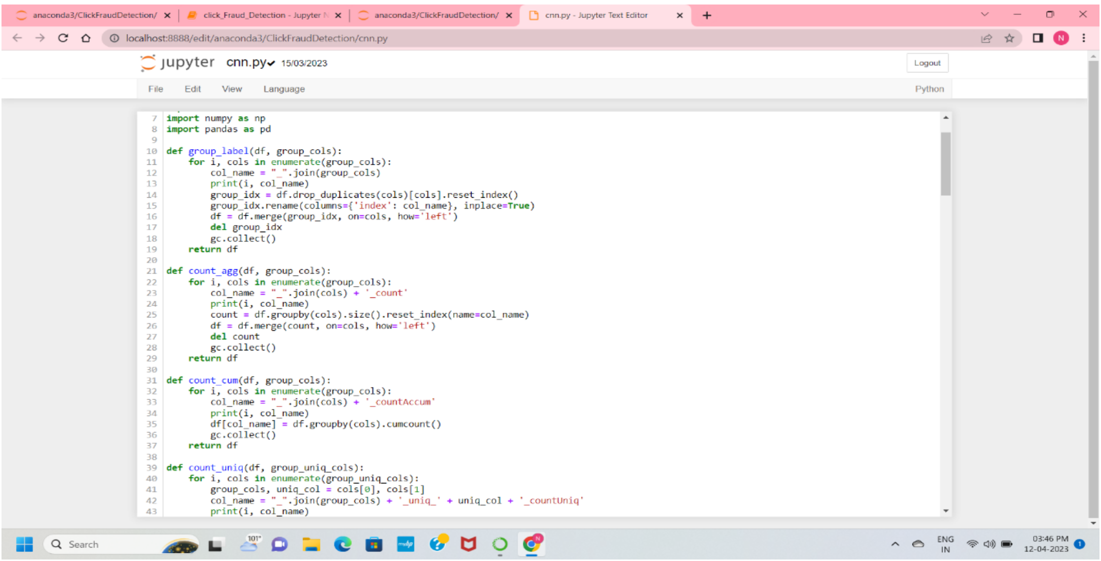


The Fully-Connected layer is a (typically) low-cost method of learning the high-level properties of non-linear topologies as expressed by the convolution kernel output. The Fully-Connected layer is acquiring a potentially non-linear variable in that space. Now that we've changed our image representation into a shape suitable for our Multi-Level Perceptron, we need to flatten it into a linear combination. The smoothed output appended to each training cycle is fed into a feed-forward neural net with back propagation.Over a series of epochs, the model will distinguish between dominant and low-level features in images and categorize them by using soft- max classification method. The features are compared to test image's attributes, and relevant traits are correlated with the provided label. Labels are typically coded as figures for computation



10. WORKING METHOD FOR PROPOSED MOD

Open the browser in your windows and open the anaconda in jupyter notebook platform. Import the python file containing the code of the model. Run the Code by pressing the run button in the anaconda. The proposed model perform analysis with dataset , to find out fraud clicks in online advertising.


anaconda3/ClickFraudDetection/
click_Fraud_Detection - Jupyter N
anaconda3/ClickFraudDetection/
cnn.py - Jupyter Text Editor
localhost:8888/edit/anaconda3/ClickFraudDetection/cnn.py
jupyter cnn.py✔ 15/03/2023
Logout
File
Edit
View
Language
Python
Search
ENG IN
03:46 PM 12-04-2023

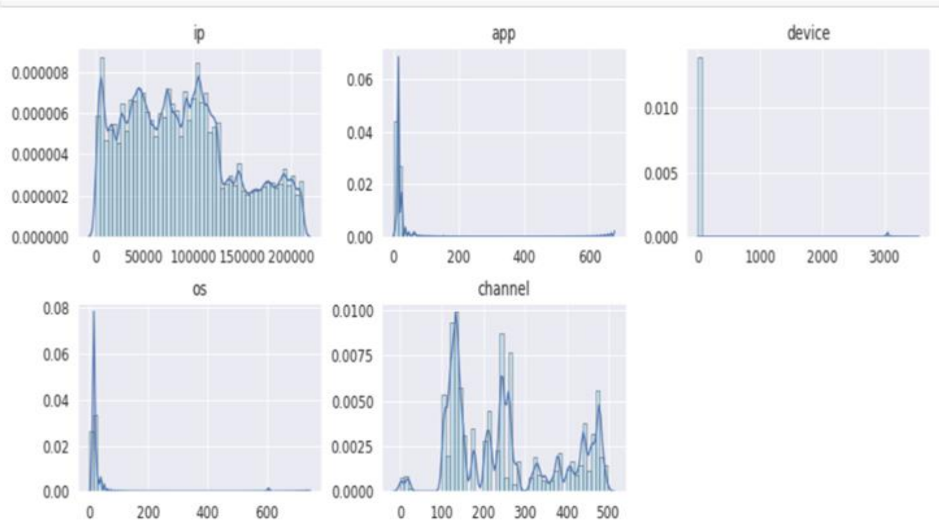
```python
import numpy as np
import pandas as pd

def group_label(df, group_cols):
    for i, cols in enumerate(group_cols):
        col_name = "_".join(group_cols)
        print(i, col_name)
        group_idx = df.drop_duplicates(cols)[cols].reset_index()
        group_idx.rename(columns={'index': col_name}, inplace=True)
        df = df.merge(group_idx, on=cols, how='left')
        del group_idx
        gc.collect()
    return df

def count_agg(df, group_cols):
    for i, cols in enumerate(group_cols):
        col_name = "_".join(cols) + '_count'
        print(i, col_name)
        count = df.groupby(cols).size().reset_index(name=col_name)
        df = df.merge(count, on=cols, how='left')
        del count
        gc.collect()
    return df

def count_cum(df, group_cols):
    for i, cols in enumerate(group_cols):
        col_name = "_".join(cols) + '_countAccum'
        print(i, col_name)
        df[col_name] = df.groupby(cols).cumcount()
        gc.collect()
    return df

def count_uniq(df, group_uniq_cols):
    for i, cols in enumerate(group_uniq_cols):
        group_cols, uniq_col = cols[0], cols[1]
        col_name = "_".join(group_cols) + '_uniq_' + uniq_col + '_countUniq'
        print(i, col_name)
```



11. CONCLUSION

We have considered various security vulnerabilities in the most prominent online advertising. We systematically examine the click fraud and its types. Click Fraud is already a threat to business online and has a great potential to increase your attack radius. New technologies like the Internet of Things can collaborate indirectly to that. The need to sharpen defenses against this coup is urgent. The financial losses resulting from this danger are very significant. We analyze and survey some detection techniques used presently in different solution domains. Click fraud not only disturbs the budget advertisers but also how bots are used to corrupt your valuable data. Hence its important to be aware and evolving to come up with solutions to circumvent and prevent them. Defense Strategies must be further improved.

12. REFERENCES

[1] Dash and S. Pal, ``Auto-detection of click-frauds using machine learning,"Int. J. Eng. Sci. Comput., vol. 10, pp. 2722727235, Sep. 2020.

[2] Y. Xie, D. Jiang, X. Wang, and R. Xu, ``Robust transfer integrated locally kernel embedding for click-through rate prediction," Inf. Sci., vol. 491,pp. 190203, Jul. 2019.

[3] L. Pan, S. Mu, and Y. Wang, ``User click fraud detection method based on Top-Rank-K frequent pattern mining," Int. J. Modern Phys. B, vol. 33, no. 15, Jun. 2019, Art. no. 1950150.

[4] T. G. Thejas, S. Dheeshjith, S. S. Iyengar, N. R. Sunitha, and P. Badrinath,``A hybrid and effective learning approach for click fraud detection,"Machine Learn. Appl., vol. 3, Mar. 2021, Art. no. 100016.

[5] Z. Li andW. Jia, ``The study on preventing click fraud in internet advertising,"J. Computing., vol. 31, no. 3, pp. 256265, 2020.

[6] D. Liu, S. Xu, L. Chen, and C. Wang, “Some observations on online advertising: A new advertising system,” in 2015 IEEE/ACIS 14th International Conference on Computer and Information Science (ICIS), pp. 387–392, June 2015.

[7] X. Li, Y. Liu, and D. Zeng, “Publisher click fraud in the pay-per-click advertising market: Incentives and consequences,” in Proceedings of 2011 IEEE International Conference on Intelligence and Security Informatics, pp. 207–209, July 2011.

[8] X. Jiarui and L. Chen, “Detecting crowdsourcing click fraud in search advertising based on clustering analysis,” in 2015 IEEE 12th Intl Conf on Ubiquitous Intelligence and Computing and 2015 IEEE 12th Intl Conf on Autonomic and Trusted Computing and 2015 IEEE 15th Intl Conf on Scalable Computing and Communications and Its Associated Workshops (UIC-ATC-ScalCom), pp. 894–900, 2015.

[9] M. Faou, A. Lemay, D. Decary-H ´ etu, J. Calvet, F. Labr ´ eche, M. Jean, B. Dupont, and ` J. M. Fernande, “Follow the traffic: Stopping click fraud by disrupting the value chain,” in 2016 14th Annual Conference on Privacy, Security and Trust (PST), pp. 464–476, 2016.

[10] B. Kitts, J. Y. Zhang, A. Roux, and R. Mills, “Click fraud detection with bot signatures,” in 2013 IEEE International Conference on Intelligence and Security Informatics, pp. 146–150, 2013.