# Ensemble Prediction of Spatio-Temporal Behavior of Distributed Denial of Service Attacks

Ahmed Abusnaina
*University of Central Florida*
Orlando, FL, USA
ahmed.abusnaina@knights.ucf.edu

Mohammed Abuhamad
*University of Central Florida*
Orlando, FL, USA
abuhamad@knights.ucf.edu

DaeHun Nyang
*Inha University*
Incheon, South Korea
nyang@inha.ac.kr

Songqing Chen
*George Mason University*
Fairfax, Virginia, US
sqchen@gmu.edu

An Wang
*Case Western Reserve University*
Cleveland, Ohio, US
axw474@case.edu

David Mohaisen
*University of Central Florida*
Orlando, FL, USA
mohaisen@ucf.edu

***Abstract*—DDoS attacks are an immense threat to online services, and numerous studies have been done to detect and defend against them. DDoS attacks, however, are becoming more sophisticated and launched with different purposes, making the detection and instant defense as important as analyzing the behavior of the attack during and after it takes place. To this end, studying and modeling the Spatio-temporal evolvement of DDoS attacks is essential to predict, assess, and combat the problem, since recent studies have shown the emergence of wider and more powerful adversaries. This work aims to model seven Spatio-temporal behavioral characteristics of DDoS attacks, including the attack magnitude, the adversaries' botnet information, and the attack's source locality down to the organization. To the best of our knowledge, this work is the first to address all behavioral characteristics of DDoS, especially for observing patterns of botnet families and information across distributed geographical localities, for modeling and prediction during the progression of the attack. Due to the variety of underlying nature of the modeling task associated with each characteristic, we leverage the state-of-the-art deep learning methods, namely: DNN, Transformers, LSTM, and CNN, to construct an ensemble of models to capture and predict behavioral patterns of the attack. The proposed ensemble operates in two frequencies, i.e., hourly and daily, to actively model and predict the attack behavior and evolvement, and oversee the effect of implementing a defense mechanism. We evaluate our approach on a large-scale real-world dataset of roughly nine million records of 50,704 verified DDoS attacks. The ensemble shows remarkable performance in predicting the behavior of all behavioral characteristics.**

***Index Terms*—Distribution Denial of Service, DDoS Behavior Prediction, Network Security, Spatio-temporal Analysis,**

## I. Introduction

Distributed Denial-of-Service (DDoS) attacks are explicit malicious attempts to prevent legitimate users from accessing a service by sending an overwhelming amount of traffic to the service server. According to Netscout annual worldwide infrastructure security report [1], the traffic generated for launching DDoS attacks exceeded 1 TBPS in size in 2019. On a more recent event, an attack of size 1.7 TBPS has been recorded. These attacks, if successful, result in a service shutdown that costs a provider an average of $221,836 per attack, as reported in Netscout [1].

The DDoS attacks have become a serious threat with the increasing growth of the verified attacks, for instance, 6.13 million attacks were recorded in 2018 with an average of 700 attacks per hour. This increase correlates with the availability of the purchasing option of bots that assist the launching of an attack, which also correlates with the increased scalability and wide-distribution of bots across geographical locations [2].

The growing threat of DDoS attacks has inspired many recent research studies to contribute to the efforts toward the analysis and characterization of the attacks [3]–[6], including methods for the attacks detection and prediction [7]–[12]. These efforts have made the field of detecting DDoS attacks widely-explored and resulting in highly-accurate detection systems [13]–[17]. However, there are limited studies that explore behavioral patterns and characteristics of the DDoS attacks during the progression of the attack and after the detection. Understanding the Spatio-temporal behavior and characteristics of the attack is crucial for defending against the attack, limiting its impact, and planing countermeasures to prevent it from occurring in the future. This study aims to contribute to this area by providing in-depth analyses and insights for modeling seven behavioral characteristics of DDoS attacks using deep learning-based methods. This analysis and modeling task takes place after the detection of the attack and continues as the attack progresses (in space and time). The Spatio-temporal analysis of DDoS behavior can be done by addressing various characteristics, such as the attack magnitude, botnet information, and attack source location.

Studying the Spatio-temporal characteristics of the DDoS attacks goes beyond the capabilities of the attacked server, as most servers are oblivious to the attacking bots and traffic sources over a long period of time. This makes it really challenging for both research and applied services to obtain data that allow such analyses. Further, even when obtaining such data from internet service providers (ISPs) for real DDoS attacks, this data is hardly containing enough information to study several characteristics of the attack, such as the botnet information. Moreover, the nature of the large-scale traffic source and information adds more complications to the modeling tools for successful capturing of patterns. Despite these many challenges, exploring methods for capturing Spatio-temporal behavior of the attacks is of paramount importance. For instance, predicting the attack magnitude is essential for a proper defense and future precautions, since it reflects the scale of the attacks as they evolve in time, and the needed countermeasures and resources to mitigate the attack effects.

These benefits of possessing the capabilities of modeling behavioral patterns of DDoS attacks are highlighted in Gupta  et al. [11], who conducted a magnitude analysis on either synthetic dataset or with laboratory constraints. Another work by Wang  et al. [18] studied the magnitude of each botnet family and the autonomous systems (ASN) distribution over the total duration of the attack. This work proposes temporal modeling of the magnitude information over time as sequences that enable the prediction of magnitude at each time-step.

Other crucial characteristics of the DDoS attacks are associated with botnet information, such as botnet family and ID, that is responsible for generating the traffic. To the best of our knowledge, we are the first to attempt to model such characteristics for real-world large-scale DDoS verified attacks. Modeling botnet-related information during and after the detection of the attack plays an important role in understanding the attack patterns [19], [20], since each botnet family tends to follow distinctive patterns that provide insights for countermeasures.

In addition to modeling the magnitude and botnet information, investigating the Geo-temporal relations and progressions of the attack's sources would provide insights to the intent, purpose, utilized resources in launching the attack, and possible magnitude estimation for the attack's evolution. This Geo-temporal analysis and modeling are captured by studying patterns from the attack source locality, i.e., country, autonomous systems, cities, and organizations. However, this task is challenging due to the wide distribution of attack sources across numerous geographical locations. Moreover, a study by Wang  et al. [18] showed that DDoS attacks have a volatile nature where bots launching an attack may shift from one geographical location to another during the attack duration. The study showed the possibility of predicting the distribution of the ASN of attackers. However, predicting the countries, cities, and registered organizations of the attack sources have not been addressed in the literature of the field, and to the best of our knowledge, we are the first to model the Geo-temporal relations and patterns during and through the entire attack duration.

This paper is dedicated to investigate several Spatio-temporal characteristics of the DDoS attacks, namely, attack magnitude, botnet family and ID, attack source locations including countries, ASNs, cities, and organizations. Due to the underlying nature of patterns to be extracted for separate characteristics, we leverage current state-of-the-art machine learning methods, including Deep Neural Networks (DNN), Long Short Term Memory (LSTM), Transformer, and Convolutional Neural Networks (CNN), to model separate characteristics and construct an ensemble of models to predict at different frequencies the behavioral patterns of DDoS attacks. The ensemble incorporates 14 different models, two for each characteristic, and operates in two frequencies, hourly-based, and daily-based frequencies, to actively monitor and account for the latest status of the attack while in progress. The ensemble is built and evaluated on a large-scale real-world dataset that includes 50,704 verified DDoS attacks launched by eleven botnet families and 674 botnet IDs on 9,026 targets from August 2012 to March 2013. Further, this work sheds light on different aspects and patterns of DDoS attacks.

**Contribution.** This work presents an ensemble of models to predict the Spatio-temporal behavioral patterns of DDoS attacks. The contribution of this work is as follows:

- **Modeling Spatio-temporal Characteristics:** Predicting seven different characteristics of the ongoing DDoS attacks using Spatio-temporal behavioral patterns of the attack, namely: *attack magnitude*, *botnet family*, *botnet ID*, *attack source country*, *ASN*, *city*, and *organization*, using large-scale real-world dataset of approximately nine million records of verified DDoS attacks.
- **Constructing Predictive Ensemble:** Implementing an ensemble of seven models based on four machine learning architectures, namely, DNN, LSTM, Transformer, and CNN, to actively predict the attack behavior on different operational frequencies (hourly and daily bases). The ensemble shows remarkable performance in both data sampling frequencies, achieving high accuracy in predicting different behavioral aspects of the attack.
- **Addressing Unseen Attacks and Targets:** Evaluate the performance of the ensemble on a real-world large-scale dataset of known and unseen targets and DDoS attacks. The ensemble offers high accuracy over targets with no attacking history, and new represented DDoS attacks.
- **Addressing the Cold Start Problem:** We investigate the effect of cold start problem, i.e., modeling with insufficient information such as at the beginning of the attack. We show that the ensemble can achieve high accuracy even under the cold start situation.

**Organization.** The work is organized as follows: in section II, a description of the dataset is provided. Then, system design and utilized architectures are described in section III. The experiments and evaluation results are shown and discussed in section IV. We provide the related work in section V. Finally, we conclude our work in section VI.

## II. Dataset Overview

### A. Dataset Collection

The utilized dataset is provided by the monitoring unit of a DDoS mitigation company [21]. Traces of malicious infected hosts were collected by collaborating with over 300 major Internet Service Providers (ISPs) globally monitoring attacks launched by specific malicious actors worldwide across America, Europe, Asia, Africa, and Australia. The activities of the participating hosts in the given botnet attacks, by either communicating with the infected infrastructure or launching the actual DDoS attack on the monitored targets, were monitored and analyzed over time. To this end, the traces of the traffic associated with various botnets were collected using different sensors on the Internet, in corporation with several ISPs, where the source of the collected traffic is an infected host participating in botnet attacks, and the destination is a verified targeted client. Afterward, malware botnets used in launching various attacks were reverse engineered and labeled to a known malware family using best practices (i.e., AMAL, a fully automated system for analysis, classification, and clustering of malware samples) [22], [23]. The dataset consists of 50,704 verified DDoS attacks collected in the period of 08/29/2012 to 03/24/2013, a total of 207 days, targeting 9,026 clients, represented as hourly snapshots of each family activities over the monitored period, including the botnet information, targeted client IP, and the IPs of the hosts associated with the botnet attack.

### B. Behavioral Characteristics of DDoS Attacks

To create a better understanding of the DDoS attacks, we focus on three groups of characteristics: attack magnitude, botnet information, and attack source location. The following is a description of each group.

**Attack Magnitude (AM).** This attribute refers to the number of DDoS attacks launched by infected hosts on a specific target over a period of time. It is important to understand the magnitude of the attack to estimate and allocate the correct resources to counter the attack. The attack magnitude is defined by the total number of attacks on a single target over a period of time, even when the attacking bots belong to different families and with different attacking objectives. We observe a maximum hourly and daily attack magnitudes of 581,893 and 13,876,995, respectively.

**Botnet Information.** The importance of knowing the attacking botnet families lies in implementing the correct defense against the attack since popular botnets have well-known attack patterns. Therefore, two characteristics have been extracted: *botnet family (BF)* and *ID*. The DDoS attacks reported in our dataset originated mainly from eleven popular botnet families: *dirtjumper*, *darkshell*, *ddoser*, *nitol*, *pandora*, *blackenergy*, *optima*, *aldibot*, *yzf*, *colddeath*, and *armageddon*. Botnet families may evolve over time. Therefore, new botnet generations are marked by their unique MD5 and SHA-1 hashes. We consider the botnet ID as a standalone characteristic, as the behavior of the botnet may change over several generations. Table I shows the number of botnet IDs associated with DDoS attacks for each family. Note that the eleven botnet families have a total of 674 different botnet IDs, indicating the continuous evolvement of botnets over time. The number of records represents the instances of recorded DDoS attacks associated with infected hosts from a malicious botnet family. The number of attack records is an indicator of the activity of the botnet family during the monitored period. However, it does not reflect the evolvement of the botnets, since the number of records associated with botnet IDs that belong to *pandora* (41 botnet IDs with 1,397,027 records) are relatively larger than those of *darkshell* (166 botnet IDs with 80,129 records).

TABLE I: Distribution of the botnet IDs over botnet families. *Dirtjumper* is associated with 251 botnet IDs, and 6,902,882 records within the monitored duration. However, the popularity of a malware family during the monitored period does not reflect the evolvement of it (i.e., *darkshell* and *pandora*).

| Family | # Botnet IDs | # Records |
|---|---|---|
| dirtjumper | 251 | 6,902,882 |
| darkshell | 166 | 80,129 |
| ddoser | 102 | 37,172 |
| nitol | 43 | 20,411 |
| pandora | 41 | 1,397,027 |
| blackenergy | 28 | 95,330 |
| optima | 25 | 41,321 |
| aldibot | 9 | 269 |
| yzf | 6 | 113,923 |
| colddeath | 2 | 28,259 |
| armageddon | 1 | 906 |
| Total | 674 | 8,717,629 |

**Attack Source Location.** It has been shown that botnets have strong geographical and organizational localities [24]. Therefore, such information can be used to predict future attack source locations and the shifting patterns of attackers across geographical locations to help in planning defenses and countermeasures. To this end, the hosts IP addresses were used to extract the attack source country *(CN)*, city *(CT)*, organization *(OG)* and *(ASN)*, using the IP-to-region dataset and MaxMind online database [25]. In the monitored duration in which the dataset is collected, the attack source locations were distributed over 186 countries, 2,996 cities, 4,036 organizations, and 4,375 ASNs, The distribution of the infected hosts indicates the existence of worldwide botnet infections.

### C. Dataset Splitting

The focus of this study is to predict the incoming attack characteristics in order to assist the targeted client in planning a defense mechanism. To this end, the dataset is split into three parts as shown in Figure 1 and as follows. ① *Training dataset:* The training dataset contains the traces and records of 80% (7220) of DDoS attacks' victims (i.e., targeted clients). For the purpose of predicting the behavioral patterns of the attacks during the attack progression, we considered the records that occurred at the first 80% of the attack duration for each victim (*target*) as the actual training dataset. ② *Known targets testing dataset:* This dataset contains the remaining records that occurred during the last 20% of the attack duration per target. This sub-dataset is used to evaluate the prediction models in modeling the behavioral pattern of DDoS attacks on targets with known history (by observing the earlier 80% of the attack duration) since such modeling capabilities help in predicting patterns of ongoing attacks after the detection. ③ *Unseen targets testing dataset:* This dataset consists of DDoS attack records of the remaining 20% (1806) of targeted clients that are not considered in the training dataset. The

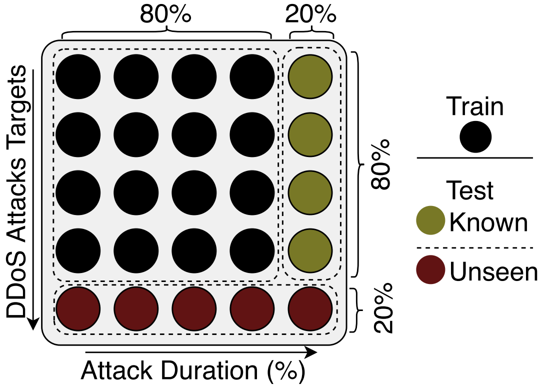


Fig. 1: Distribution of the dataset over different dataset partitions. The training dataset consists of 80% of the DDoS attacks duration launched on 80% of the targeted. Known targets dataset consists of the remaining 20% of the attack duration of targets in the training dataset. Unseen targets datasets consist of the remaining 20% of the targets.

aim of this dataset is to evaluate the prediction models over targets that have never been attacked before. Despite this challenging task, recent studies [26] showed that DDoS attacks have certain repetitive patterns, and put forward this challenge to testing and aim to evaluate our approach on records of DDoS attacks targeting victims for the first time. Table II shows the distribution of the dataset characteristics over each partition of the dataset. Note both test datasets are diverse enough to evaluate the performance of the prediction models.

## III. System Design

The focus of the proposed system is to predict the attacker behavior over the seven different characteristics. The system design is shown in Figure 2.

### A. Operational Frequency and Data Pre-processing

We adopted two operational frequencies to model and analyze behavioral patterns of DDoS attacks, namely: *agile* and *passive* approaches. The data pre-processing and handling follows the same manner in both approaches with slight modifications to fit the operational duration and data availability associated with the adopted operational mode.

**Operational Mode.** For studying attack behavior manifested with the considered characteristics, data records were aggregated at different frequencies (i.e., *Agile* mode with hourly frequency and *Passive* mode with daily frequency). We highlight the data processing stages for both operational modes, as data are aggregated and processed according to the frequency of adopted mode. The agile mode requires six hours of data to be fully-functional at an hourly frequency, while the passive mode requires three days of information to be full-functional in modeling behavioral patterns at a daily frequency.

**Data Processing and Sequence Generation.** Addressing different characteristics of DDoS attacks captured by their records, the data is represented as $\Phi_{\mathcal{X}} = \{\phi_1, \phi_2, \ldots, \phi_t\} \in \mathbb{R}^{N\times T}$, where $\phi_\alpha \in \mathbb{R}^{1\times T}$ is a vector of the attribute in hand ($\Phi$) for the attack targeting $\mathcal{X}$ at a given time step $\alpha$ (e.g., $\phi_1$ and $\phi_t$ represent the vectors of the first and last time step), $T$ is the maximum length of the reported attacks, and $N$ is the total number of targeted clients. For instance, addressing the *botnet ID* of an attack targeting $\mathcal{X}$, the data is represented as a matrix $ID_{\mathcal{X}} = \{id_1, id_2, \ldots, id_t\} \in \mathbb{R}^{N\times T}$, where $id_\alpha \in \mathbb{R}^{1\times T}$ is a vector of botnet IDs targeting $\mathcal{X}$ at a given time step $\alpha$. We achieve such representation by the following steps.

Ⓐ *Tokenization and Encoding:* When studying the attack characteristics, we assign identifiers for unique elements (e.g., botnet IDs are assigned to unique identifiers when processing the *ID* attribute). Assuming an attack at target $\mathcal{X}$ in a time step $\alpha$, the *ID* attribute is represented with a vector of all unique botnet IDs identifiers occurring in the attack record within $\alpha$, arranged in ascending order. For example, assuming the IDs appear in a certain attack record at the first time step are $\{id_{32}, id_{105}, id_{12}\}$, then, we present the vector as $\text{ID}_0 = \{id_{12}, id_{32}, id_{105}\}$. Following the sequence of the attack through time, the sequence of attribute vectors is generated with different lengths depending on the magnitude of the attribute. Note the differences between attribute magnitude and attack magnitude, where the former demonstrates the number of unique attribute elements included for a given attack, while the latter represents the number of bots launching the attack.

Ⓑ *Sequence Extraction:* The sequence of attribute behavior of DDoS attacks is extracted with different frequencies. In this study, we examine one-hour and one-day data frequency to analyze the behavior of the attack characteristics. Sequence extraction refers to the length of the previous time steps required to predict future steps. In the agile approach, we chose six-time steps (i.e., six hours) to be a sufficient time needed to predict future behaviors based on our experiment. For example, IDs sequences are generated as follows: $\text{Seq}_1 = \{\text{ID}_1, \text{ID}_2, \ldots, \text{ID}_6\}$, $\text{Seq}_2 = \{\text{ID}_2, \text{ID}_3, \ldots, \text{ID}_7\}$, and so on. Figure 3 shows the accumulative distribution of the DDoS attacks duration. Note that the agile approach uses hourly insights of the attack characteristics. Here, the average attack duration is 24 hours. The agile approach is suitable for such attacks as the prediction is made on an hourly bases. Operating with the passive approach, we chose three time steps (three days) as a sufficient information to predict daily future behavior. Similarly, the IDs sequences generated from passive approach are as follows: $\text{Seq}_1 = \{\text{ID}_1, \text{ID}_2, \text{ID}_3\}$, and $\text{Seq}_2 = \{\text{ID}_2, \text{ID}_3, \text{ID}_4\}$.

Ⓒ *Attribute Vector Padding:* The input data for each attribute are presented with different lengths based on the attribute magnitude at each time step. To allow efficient processing and tensor calculation, all vectors are padded to the maximum length enabling the packing of several attribute vectors in one sequence as well as packing several sequences in one batch.

Ⓓ *Attribute Vector Embedding:* Attribute vectors are forwarded to an embedding layer in all deep learning-based models in our ensemble, to enable compact representation of vectors. The embedding layer allows the learning process of a better presentation of the attribute vectors in response to

TABLE II: Overall characteristics of the dataset distribution. Both train and known targets datasets consist of 7,220 targets as they are parts of the same DDoS attacks. Unseen targets dataset consists of attacks on the remaining 1,806 targets, distributed over 151 countries and 234,113 IP addresses.

| Partition | # Targets | # Families | # Botnet IDs | # IPs | # Countries | # ASN | # Cities | # Organization |
|---|---|---|---|---|---|---|---|---|
| ① Train Dataset | 7,220 | 11 | 605 | 841,471 | 186 | 4150 | 2,877 | 3,831 |
| ② Known Targets | 7,220 | 11 | 606 | 158,230 | 179 | 3,275 | 2,275 | 3,024 |
| ③ Unseen Targets | 1,806 | 10 | 248 | 234,113 | 151 | 2,571 | 1,800 | 2,382 |
| Overall | 9026 | 11 | 674 | 880,451 | 186 | 4,375 | 2,996 | 4,036 |

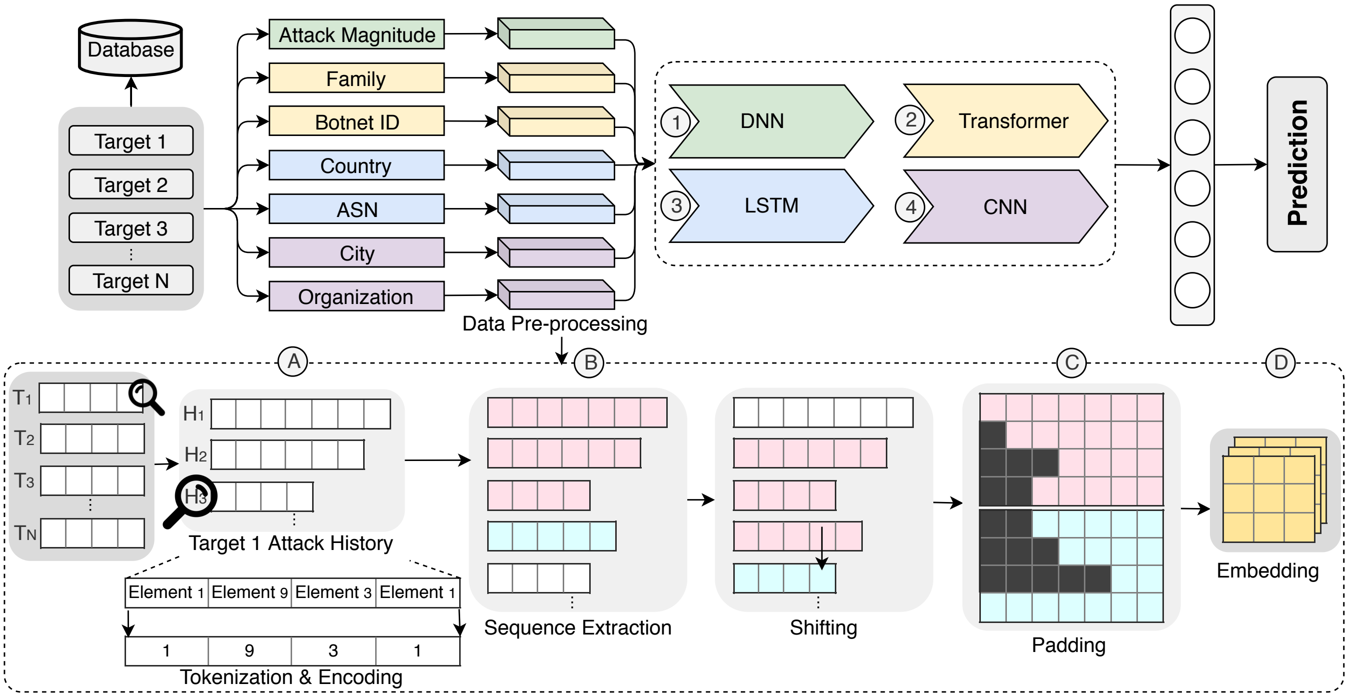


Fig. 2: The general flow of the DDoS attacks prediction design. Here, T refers to the attacked target, whereas H refers to one hour in the attack duration. Colors indicate the deep learning architecture associated with the attack attribute. The data pre-processing consists of tokenization and encoding, sequence extraction, and vector padding and embedding. Then, the processed input is inputted to different deep learning algorithms to learn and predict the DDoS attacks behavior and characteristics.

a given task during the training process. Vectors represented with attribute identifiers $\phi_\alpha \in \mathbb{R}^T$, where $T$ is the maximum occurrence of unique identifiers in an attack, will be embedded to $\gamma_\alpha \in \mathbb{R}^{128}$, where 128 is the size of the vector embedding. We chose the size of the embedding based on several experiments that showed 128 is adequate to incorporate the information present in the attribute vector. Sequences are then viewed as matrices of $\Gamma_\alpha \in \mathbb{R}^{t_s \times 128}$, where $t_s$ is the number of time steps (i.e., the sequence length).

**Attack Magnitude.** We also study the attack magnitude with different frequencies as in agile and passive approaches. We note that the approach to predict and study attack magnitude is different from the one adopted for other characteristics. While other attribute vectors are important to extract, the magnitude of the attack is calculated per targeted client at each time step and presented as one real value (instead of attribute vector). Thus, only stepⒷ is required from the aforementioned approach, which aims to generate sequences of the calculated value of magnitude at each time step. To present the values of magnitude to the deep learning model, we normalize the values in the range of zero to one.

### B. Prediction Models Architectures

Our approach adopts an ensemble of powerful classifiers to predict different behaviors of DDoS attacks including DNN, Transformer, LSTM, and CNN. In contrast to traditional machine learning methods, deep learning allows deep features and patterns extraction, which is required in predicting the future characteristics of the DDoS attack, particularly that the input is a sequence of the attack characteristics within a window time. The utilized model should be able to extract and learn different patterns of the DDoS attack in order to correctly predict the future behavior of the attack. Here, LSTM and Transformer are used for their memory-based learning capabilities. Additionally, DNN and CNN are selected for their capabilities in extracting deep feature representations of the

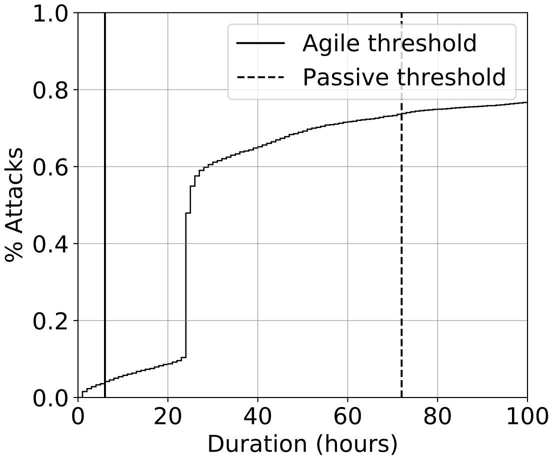


Fig. 3: Accumulative attacks duration distribution. The average attack duration is 24 hours, with 23% of the attacks duration more than 100 hours. The agile approach operates within 6 hours window, which is suitable for 95.84% of the attacks.

sequences, and therefore predicting the upcoming behavior. Further, we chose different model architectures for modeling different tasks (i.e., characteristics behaviors) since certain architectures are proven to work better than the others in certain circumstances. In particular, the best performing deep learning architecture in predicting each DDoS attack characteristic is reported. Based on our experiments, we present an ensemble of models that predicts seven DDoS attacks characteristics with a high degree of accuracy. The ensemble includes *DNN* for predicting attack magnitude, *Transformer* for predicting botnet information (ID and family), *LSTM* for predicting the wide attack source locality (country and ASN), and *CNN* for predicting the specific attack source locality (city and organization). A brief description of the experimental settings and hyperparameters to build the models are in the following:

**DNN for Attack Magnitude.** The model architecture consists of four dense layers of size 1,000 units with ReLU activation function. Each dense layer is followed by a dropout operation with a rate of 30%. The last layer is connected to a sigmoid layer of size one signaling the normalized number of the attack magnitude (i.e., the scale of the magnitude from zero to one). Given sequences of ground-truth magnitudes samples, the model is trained to predict an output $y_i = \{x_i^{n+1}\}$ given an input $x_i = \{x_i^1, x_i^2, \ldots, x_i^n\}$, where $n$ is the sequence length (e.g., six in the agile approach), following straightforward supervised learning process.

**Transformer for Botnet Information.** The model is adopted from the model proposed by Vaswani et al. [27]. It consists of stacked layers in both encoder and decoder components. Each layer has eight sub-layers comprising multi-head attention layer, $\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^t}{\sqrt{d_k}})V$, where $Q$ is set of queries, $K$ is set of keys and $V$ is set of values. The model performs attention $h$ times, $\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, \text{head}_2, \ldots, \text{head}_h)W^o$, where $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$, followed by a feed forward network. Here, $W_i^{[.]}$ and $W^o$ are projection parameters. The prediction is done by conducting a beam search with length penalty ($\lambda = 0.6$). The Transformer is used to train two models performing two separate tasks, predicting botnet family and botnet ID.

**LSTM for Wide Attack Locality.** The model consists of one LSTM layer with a size of 128 units. The LSTM layer is followed by a dense layer of size 128 and a dropout operation with a rate of 20%. Then, a dense layer with a sigmoid activation function is used to output the prediction of attack source locality. The output layer (the sigmoid layer) has the size of the attribute vector addressed in a given task. For example, for predicting the country, the model has an output layer of size 186, one sigmoid node for each country identifier. The predicted sigmoid value at each unit is rounded with a threshold of 0.5 to indicate whether the country identifier assigned to this unit is included in the attack or not. The LSTM architecture is used to train two models for predicting attack source country and ASN.

**CNN for specific attack locality.** The model architecture consists of one convolutional layer with 64 kernels of size $1\times3$ convolving over the input vector, followed with a sigmoid output layer of size equals to the size of the addressed attribute vector (i.e., to predict the future status of the attack). The CNN architecture is used to train two models for predicting the specific attack locality (i.e., city and organization).

**Training Setting.** All models were trained with 100 training epochs. The weights of the models were initialized using a zero-mean random uniform distribution. The training was guided by minimizing the loss using *Adam* optimizer set with a fixed learning rate of $10^{-3}$. The binary-cross-entropy loss was used to train all models except for the DNN, which uses mean-squared-error loss due to the nature of the performed task (attack magnitude). The training process follows a straightforward supervised learning process and other details related to specific model architecture are provided with the model description in subsection III-B.

## IV. Evaluation and Discussion

We evaluate our approach for predicting DDoS characteristics behaviors on a large-scale dataset. We report our results using two evaluation metrics, namely True Positive Rate (TPR) and True Negative Rate (TNR). TPR represents the number of correctly predicted elements over all the elements that occurred within the duration of the prediction. For instance, if the DDoS attack launched from four countries, of which, the prediction model predicts three correctly, the TPR is equal to 75% (3/4). TNR is referred to as $1 - (FP/N)$ where $FP$ is the number of the incorrectly predicted elements and $N$ represents all the elements that did not occur within the duration of the prediction. For instance, if the DDoS attack launched from four countries out of 186, and the prediction model incorrectly predicts two elements, the TNR is equal to 98.90% ($1 - (2/182)$). Note that TPR and TNR are preferred metrics in evaluating the systems as true indicators of performance in different scenarios. For example, achieving a TNR of 100% means zero false alarms (e.g., TNR = 100% for a model predicting attack organization means not once the model falsify an organization for an attacker). On the other hand, TPR indicates the precision of predicting the attack behavior (e.g., TPR = 100% for a model predicting the organizations of attackers means correctly predicting all the involved organizations in the attack). Therefore, it's important for all models to maintain high TPR and TNR.

### A. *Attack Characteristics Evaluation*

The performance of the system is evaluated with the ensemble performance shown by the individual models for each of the attack characteristics. In the following, we present our findings for each attribute. Figure 4 summarizes the results of six attack characteristics using both *known* and *unseen* test targets when adopting the agile and passive approaches. The seventh attribute (i.e., the attack magnitude) is evaluated separately due to the data nature of the attribute. Two models were implemented for each attribute, one for each operational frequency, and evaluated on known and unseen targets.

**Attack Magnitude.** We evaluate the DNN model for predicting the attack magnitude using the *mean error* metric, where the error is calculated as the difference between the actual and predicted attack magnitude. Since the data observations were normalized, the output of the model indicates the magnitude as a fraction of the maximum recorded magnitude, i.e., 581,893 for the agile approach and 13,876,995 for the passive approach. Then, to calculate the magnitude, we multiply the model's output by the maximum magnitude rounded to the decimal point, e.g., the agile model's output 0.031 indicates a magnitude of 18,039 attacking hosts.

We report the results in Table III for evaluating the models on *known* and *unseen* test targets using both agile and passive approaches. The results were reported using the mean error rate and the average shift error, that can be used as indicators of the error margin of the model's prediction. The shift error is reported by the actual number of attacking hosts contributed to the attacks. For example, the error rate of the agile model on *unseen* data is 0.0014% which is off by roughly 86 hosts from the actual number of the attacking hosts. Even though this number might seem large at first, it appears to be a good estimate knowing that the average attack magnitude on the agile data sampling rate (i.e., per the hour) equals 551 hosts (15.60% deviation). Similarly, the average shift rate for the passive approach is roughly 1,977 hosts for predicting the magnitude of *unseen* targets, which is also acceptable estimation knowing the average of magnitude is 15,394 hosts per day (12.97% deviation).

**Botnet Family.** Figure 4a shows the evaluation of the Transformer architecture trained to predict botnet family using different settings. The models achieve TPR of 95.97% and TNR of 99.65% for predicting botnet families of known targets on one-hour frequency, while maintaining TPR of 79.50% and TNR of 98.94% for unseen targets. The TPR score increases to 96.97% and 93.62% when using lower frequency (i.e., one-day) for known and unseen targets, respectively. We show the average accuracy of predicting each botnet family using both the agile and passive approaches in Table IV. The result demonstrates high accuracy for almost all botnet families except for *aldibot* and *pandora*. The model failure for detecting *aldibot* could be for several reasons, e.g., *aldibot* has the smallest number of records in our dataset (269 records). We also observed that the detection accuracy of the *pandora* family is 80.74%, which is less than those achieved for other families, despite the high number of records for this family (1,397,027 records). The ambiguity of patterns for this family can be explained by the length of the attacks (on average 184.33 hours per attack), which is larger than 81.21% of the attacks. Moreover, we observed a large number of attackers (7,922) which were distributed over 69 countries, 56.93% (4,510) of them are relocated in southeast Asia.

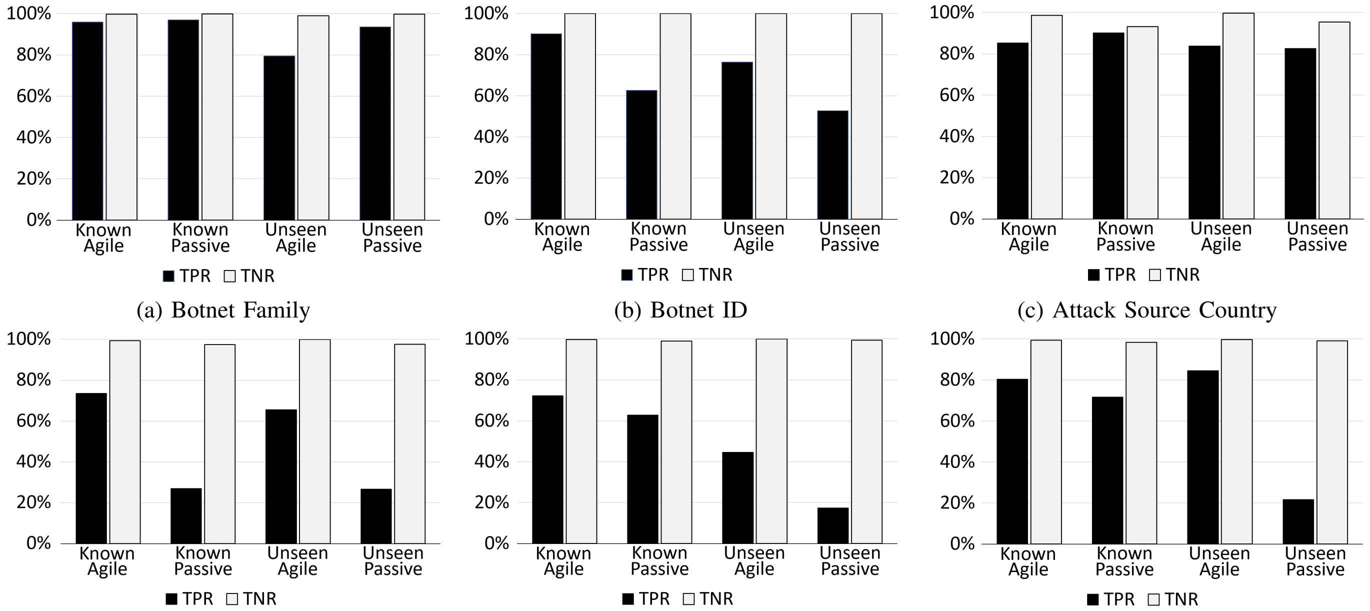
TABLE III: Attack magnitude prediction evaluation. The mean error rate is reported using the percentage, and the average shift error is reported by the actual number of the hosts. Overall, the model can predict the incoming attack magnitude within a time window with high accuracy.

| Approach | Target | Mean Error Rate | Avg. Shift Error |
|---|---|---|---|
| Agile | Known | 0.015% | ∓88.64 |
| | Unseen | 0.014% | ∓85.87 |
| Passive | Known | 0.012% | ∓1,734.40 |
| | Unseen | 0.014% | ∓1,976.65 |

TABLE IV: Overall botnet families prediction accuracy. *Armageddon* and *ddoser* have a 100% prediction accuracy, while *aldibot* has 0% prediction accuracy, this is due to the low number of records (269) associated with it. Maintaining high botnet family prediction accuracy is essential as the overall attack behavior and progression over time is highly associated with the botnet family originating the attack.

| Botnet Family | Accuracy | # Bots |
|---|---|---|
| armageddon | 100% | 80 |
| ddoser | 100% | 9 |
| darkshell | 99.96% | 146 |
| optima | 99.85% | 37,625 |
| blackenergy | 98.95% | 151,043 |
| colddeath | 98.65% | 1,552 |
| nitol | 97.97% | 46 |
| yzf | 94.57% | 39 |
| dirtjumper | 93.49% | 718,881 |
| **pandora** | *80.74%* | 7,923 |
| **aldibot** | *0.00%* | 380 |

**Botnet ID.** The Transformer-based model achieved a remarkable TPR and TNR on predicting botnet IDs of attacks. Figure 4b shows the evaluation of the performance of the Transformer-based prediction model on known and unseen targets for agile and passive operational frequencies. The Transformer-based model achieved a TPR of 90.16% and 76.42% for predicting known targets, and unseen targets with a TNR of 99.97% and 99.95%, respectively, using agile operational frequency (hourly-based). For passive operational frequency, the model achieved a TPR of 62.96% and 52.74%



(a) Botnet Family

(b) Botnet ID

(c) Attack Source Country

(d) Attacker Source ASN

(e) Attacker Source City

(f) Attacker Source Organization

Fig. 4: Evaluation of the prediction models over known and unseen targets. The models are evaluated based on the TPR and TNR. In general, agile approach outperform passive approach in most cases. Similarly, the performance of the models against known targets attacks outperforms its performance against unknown targets attacks, as it is already trained on the history of the attack. The TNR in all cases are almost above 99%, this is intended due to the criticality of the application, and the effect of false information on the usability of the ensemble.

for predicting known targets, with a TNR of 99.95% and 99.93% for unseen targets, respectively. Note that the agile frequency-based model outperforms the passive frequency-based model.

**Attack Source Country.** Figure 4c shows the performance of the LSTM-based model on known and unseen targets for agile and passive operational frequencies. The LSTM-based model achieved a high TPR and TNR on predicting the attack source country, using agile approach, we achieved a TPR and TNR of 85.26% and 98.62% for known targets, and 83.83% and 99.95% for unseen targets, respectively. Similarly, the model achieved a TPR and TNR of 90.19% and 93.21% in predicting known targets attack source countries using passive frequency, and a TPR and TNR of 82.60% and 95.39% in predicting unseen target attack source countries. Similar to the previous characteristics, the performance of the LSTM-based model operating in agile frequency outperforms the passive frequency-based model, particularly for TNR metric.

**Attack Source ASN.** For predicting the source ASNs of the attack, the same LSTM architecture is utilized. Figure 4d shows the evaluation of the LSTM-based models in predicting the attack source ASNs operating in two frequencies, agile and passive. The LSTM-based model achieved a TPR and TNR of 73.59% and 99.41% on known targets, and 65.68% and 99.96% on unseen targets, respectively, operating in agile frequency. Similarly, the model achieved a TPR and TNR of 27.06% and 97.53% on known targets, and 26.66% and 97.60% on unseen targets, respectively, on passive approach. While the passive frequency-based LSTM model performance is low, it maintains a high TNR, reducing the false alarms.

**Attack Source City.** We used the CNN-based architecture to capture the botnet behavioral patterns, particularly, attack source cities. Using agile frequency-based CNN model, we achieved a TPR and TNR of 72.23% and 99.72% for known targets, and 44.61% and 99.98% for unseen targets, respectively. For daily-based frequency (passive operational frequency), we achieved a TPR and TNR of 62.81% and 99.02% for known targets, and 17.39% and 99.34% for unseen targets, respectively. While the CNN-based models performance varies, the high TNR (low false alarms) makes it possible to utilize the provided information by the model to implement a proper defense with high confidence. Figure 4e shows different evaluation results of the CNN-based models in predicting the attack source city.

**Attack Source Organization.** Similar to predicting attack sources, CNN-based architecture is used for attack source organization prediction. Figure 4f shows the evaluation of the performance of the prediction models on known and unseen targets for both operational frequencies. We achieved a TPR and TNR of 80.42% and 99.40% on known targets, and 84.48% and 99.72% on unseen targets, respectively, using agile frequency operational mode. Similarly, we achieved a TPR and TNR of 71.69% and 98.40% on known targets, and 21.73% and 99.10% on unseen targets, respectively, using passive frequency. While attack source organization prediction models may not provide high performance in some scenarios (i.e., unseen targets using passive approach), the information provided by the model can be used to defend against the attack, particularly that the models maintain high TNR.

### *B. Discussion and Limitation*

**DDoS Attack Behavior Prediction.** This work focuses on predicting the DDoS attack behavioral patterns after the detection of the attack. Therefore, the proposed ensemble operates on top of the DDoS attack detection system, providing the starting signal and initial input data for the ensemble to operate. The purpose of the ensemble is to provide critical information and insights to help the targeted victims in designing and planning a proper defense mechanism. Such planning incorporates advantages of the behavioral patterns detected by the proposed approach to formulate defenses as follows.

- *Magnitude driven defenses:* The magnitude of the attack directly reflects its effects on the targeted client resources. For instance, a DDoS attack with a low magnitude will unlikely result in total denial of service, while ones with high magnitudes can cause shutting down the service. Understanding the ongoing attack magnitude within a continuous time window allows a better decision making process in planning and allocating resources to combat the attack and mitigate its effects.
- *Botnet-based driven defenses:* It has been shown that certain botnet families have repetitive attacking patterns. In addition, botnet families can collaborate to conduct a DDoS attack. Understanding the attack nature and behavior through its associate botnet families and IDs create a better awareness of how the attack will progress, and better defend against it.
- *Region-based driven defenses:* DDoS attacks have regional dependencies, as the infected hosts may be originated from the same region, or several related regions. Understanding the regional distribution of the infected hosts, and the over-time shifting will provide better insights to implement region-based defenses.

**Behavioral Characteristics Stability.** The stability aspect of behavioral characteristics of DDoS attacks is measured by the frequency of which the behavior changes over time. When it comes to behavioral characteristics, the stability measurement varies depending on the studied characteristic as some attributes (e.g., locality-based characteristics) are more volatile than others (e.g., botnet-based characteristics). For instance, increasing the time window of the model's prediction, as in passive approach, will result in increasing the elements to be predicted by the model, which may lead to more false positives in the prediction results. To ensure the information quality provided by the ensemble, we adjust the decision threshold of the predictors so that the TNR is high (above 99%). This explains the higher TPR for the agile approach in comparison with the passive approach in modeling locality-based characteristics. Moreover, in almost all cases, the stability decreases when the period of observation increases. This is intuitive as recent DDoS attacks follow shifting patterns through time, which also can be shown in the results as operating in high frequency (hourly) can achieve better modeling results than operating in lower frequency (daily).

**Unseen Targets.** This study shows the performance of the ensemble using known and unseen targets, each of which has its merits. While having a history of an attack can help to predict the behavioral characteristics for the progression of the ongoing attack; providing an evaluation of the ensemble on unseen targets provides insights to predict an attack's behavioral patterns even without a previous record. We show that our approach is capable of modeling the behavioral patterns of DDoS attacks for completely unseen targets, a capability that provides a more realistic approach for analyzing attacks.

**Sequence Length.** The attack's behavioral characteristics are modeled with sequences of the characteristics' information sampled with a different frequency (i.e., one-hour and one-day). The sequence length of the attribute information for both agile and passive approaches are chosen based on the experiment and the observed attacks' duration. Since 52.07% of the total attacks in our dataset exceed the one-day sampling time, the passive approach operates on the one-day frequency with a sequence length of three days that are required to predict the next day's behavior of attacks. To address the behavioral patterns on a higher frequency, the agile approach operates on the one-hour frequency and requiring six hours to predict the next-hour behavior. Figure 3 shows that the agile approach is more desirable as most (95.84%) of attacks in our dataset can be monitored and studied.

**First-hour Attack: The Cold Start.** One shortcoming of using static sequence length (e.g., six hours for the agile approach) is in addressing attacks with a shorter duration such as the attacks in their beginnings. For example, predicting the behavioral patterns of a three-hours attack or the next hour behavior of the just-reported attack. To overcome this problem, we implemented the ensemble to operate on the specified frequency using the available information aggregated using the sampling time while padding the unavailable sequence steps with zero-vectors. For example, assume an attack with only two-hours information is available, the agile approach will process the two-hours vectors and pad four-steps of zero-vectors to predict the third hour. This approach has shown to be effective in our experiments, especially for predicting botnet families and attack source countries. However, it comes with a cost when addressing volatile characteristics such as the attack source cities. For instance, using six-hours information for known and unseen targets, the agile approach predicts the attack source cities with TPR of 72.23% and 44.61%, respectively; while using only one-hour information results in TPR of 65.56% and 17.49% for the same settings, respectively, while maintaining a high TNR (≈99%).

## V. Related Work

DDoS attacks have been intensively investigated to achieve a better understanding of them, in both detecting and predicting the malicious attacks.

**DDoS Attacks Detection.** DDoS attacks detection is well explored in different environments. Mirsky et al. [28] presented

Kitsune, a plug and play online network intrusion detection system by tracking the patterns of every network channel. Similarly, Sekar  et al. [13] proposed LADS (Large-scale Automated DDoS detection System), a triggered, multi-stage in-network DDoS detection system to overcome the scalability issues in detecting DDoS attacks over large-scale monitored network. In addition, Chang  et al. [24], [29] performed an in-depth analyses of botnet behavior patterns by monitoring and analyzing the data of the most active 16 botnet families over a period of seven months. Their analysis showed that different botnets start to collaborate when launching DDoS attacks. Similarly, they conducted an in-depth analyses measurement study of 23 active botnet families for a period of seven months. The findings of their analysis showed that bots recruitment has strong geographical and organizational locality, different than the common perception that bots are randomly recruited in a best-effort manner. Gu  et al. [30] developed a framework to detect botnets by analyzing botnet communication patterns using unsupervised machine learning techniques. Base on this work, Perdisci  et al. [31] presented a network-level behavioral HTTP-based malware clustering system based on the structural similarity between the malicious HTTP traffic. Moreover, Lu  et al. [10] detected and clustered botnet traffic into C&C channels using the K-mean clustering algorithm on large-scale network traffic payload signatures. In a more recent work, Doshi  et al. [32] distinguished normal traffic from DDoS attack traffic using limited packet-level features. By training five different machine learning algorithms, they achieved DDoS traffic detection rate of 99.9%. Further, Bhatia  et al. [33] proposed a network-centric and behavioral learning-based unsupervised machine learning technique for network anomaly detection, particularly, DDoS attacks. Their design benefits from the SDN-based mechanisms in detecting and mitigating different DDoS attacks. Additionally, Kesavamoorthy  et al. [34] used autonomous multi-agent system for DDoS attacks detection. Agents use a particle swarm optimization for reliable communication with each other and a coordinator, allowing an accurate detection of DDoS attacks.

**DDoS Attacks Behavior Prediction.** In addition to detecting the attacks, recent studies predicted different aspects of the attack behavior, such as Gupta  et al. [11], where they estimated the number of bots involved in a flooding DDoS attack with high accuracy by calculating various statistical performance measures. In addition, Fachkha  et al. [35] proposed a systematic approach for inferring DDoS activities, predicting DDoS attack characteristics, namely, the intensity rate (packets/sec) and size (number of used bots), in addition to clustering various targets to the same DDoS campaign. Furthermore, Wang  et al. [18] designed three DDoS attacks models from temporal (attack magnitudes), spatial (attacker origin), and Spatio-temporal (attack inter-launching time) perspectives by analyzing 50,000 verified DDoS attacks. The models were able to predict the DDoS attacks with high accuracy in terms of the magnitude, duration, inter-launching time, and location (ASN).

Even though recent studies investigated the attack detection and behaviors, only a few of them provided information that would assist the client in implementing a proper defense on the spot. However, our design provides the victim with essential information, such as botnet family and exact location, including the city and organization, in addition to the magnitude of the attack and the botnet ID, specifying the generation of the botnets involved in the attack while it progresses over time, such information can be utilized to properly implement a magnitude-based, region-based, and malware-based DDoS attacks mitigation techniques and defenses.

## VI. Conclusion

This work proposes an ensemble approach for studying and predicting the behavioral characteristics of DDoS attacks. This work introduces an approach to building an ensemble of models to predict seven behavioral characteristics of DDoS attacks, which provides insights for handling such attacks. All models in the ensemble leverage the capabilities of deep learning methods that obviate the burden of hand-crafting features for specific characteristics. Instead, the models learn to capture distinctive patterns within the sequences of attribute information. Evaluating our approach on a large-scale real-world dataset that contains records of more than fifty thousand verified attacks, the results of our approach show remarkable performance when operating on different sampling frequencies and under different settings. This success of efficient and accurate modeling of DDoS attack characteristics can help in implementing proper defenses and future planning for mitigating and handling of the problem.

## References

[1] Netscout. (2019) Netscout 14th annual worldwide infrastructure security report. Available at [Online]: https://www.netscout.com/report/.

[2] J. J. Santanna, R. van Rijswijk-Deij, R. Hofstede, A. Sperotto, M. Wierbosch, L. Z. Granville, and A. Pras, "Booters - an analysis of ddos-as-a-service attacks," in *Proceedings of the IFIP/IEEE International Symposium on Integrated Network Management, IM.*, 2015, pp. 243–251.

[3] R. Rasti, M. Murthy, N. Weaver, and V. Paxson, "Temporal lensing and its application in pulsing denial-of-service attacks," in *Proceedings of the IEEE Symposium on Security and Privacy, SP.*, 2015, pp. 187–198.

[4] S. K. Fayaz, Y. Tobioka, V. Sekar, and M. Bailey, "Bohatei: Flexible and elastic ddos defense," in *Proceedings of the 24th USENIX Security Symposium, USENIX Security.*, 2015, pp. 817–832.

[5] T. Vissers, T. van Goethem, W. Joosen, and N. Nikiforakis, "Maneuvering around clouds: Bypassing cloud-based security providers," in *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security.*, 2015, pp. 1530–1541.

[6] C. Rossow, "Amplification hell: Revisiting network protocols for ddos abuse," in *Proceedings of the 21st Annual Network and Distributed System Security Symposium, NDSS.*, 2014.

[7] D. Gong, M. Tran, S. Shinde, H. Jin, V. Sekar, P. Saxena, and M. S. Kang, "Practical verifiable in-network filtering for ddos defense," in *39th IEEE International Conference on Distributed Computing Systems, ICDCS 2019, Dallas, TX, USA, July 7-10, 2019*, 2019, pp. 1161–1174.

[8] M. M. Najafabadi, T. M. Khoshgoftaar, C. Calvert, and C. Kemp, "A text mining approach for anomaly detection in application layer ddos attacks," in *Proceedings of the Thirtieth International Florida Artificial Intelligence Research Society Conference, FLAIRS.*, 2017, pp. 312–317.

[9] L. Bilge, D. Balzarotti, W. K. Robertson, E. Kirda, and C. Kruegel, "Disclosure: detecting botnet command and control servers through large-scale netflow analysis," in *Proceedings of the 28th Annual Computer Security Applications Conference, ACSAC.*, 2012, pp. 129–138.

[10] W. Lu, G. Rammidi, and A. A. Ghorbani, "Clustering botnet communication traffic based on n-gram feature selection," *Computer Communications*, vol. 34, no. 3, pp. 502–514, 2011.

[11] B. Gupta, R. Joshi, and M. Misra, “Prediction of number of zombies in a ddos attack using polynomial regression model,” *Journal of advances in information technology*, vol. 2, no. 1, pp. 57–62, 2011.

[12] C. Li, Y. Wu, X. Yuan, Z. Sun, W. Wang, X. Li, and L. Gong, “Detection and defense of ddos attack–based on deep learning in openflow-based sdn,” *International Journal of Communication Systems*, vol. 31, no. 5, p. e3497, 2018.

[13] V. Sekar, N. G. Duffield, O. Spatscheck, J. E. van der Merwe, and H. Zhang, “LADS: large-scale automated ddos detection system,” in *Proceedings of the 2006 USENIX Annual Technical Conference, Boston, MA, USA, May 30 - June 3, 2006*, 2006, pp. 171–184.

[14] X. Ma and Y. Chen, “Ddos detection method based on chaos analysis of network traffic entropy,” *IEEE Communications Letters*, vol. 18, no. 1, pp. 114–117, 2014.

[15] K. Lee, J. Kim, K. H. Kwon, Y. Han, and S. Kim, “Ddos attack detection method using cluster analysis,” *Expert systems with applications*, vol. 34, no. 3, pp. 1659–1665, 2008.

[16] Q. Niyaz, W. Sun, and A. Y. Javaid, “A deep learning based ddos detection system in software-defined networking (SDN),” *ICST Transactions on Security and Safety*, 2017.

[17] L. Feinstein, D. Schnackenberg, R. Balupari, and D. Kindred, “Statistical approaches to ddos attack detection and response,” in *3rd DARPA Information Survivability Conference and Exposition (DISCEX-III 2003), 22-24 April 2003, Washington, DC, USA*, 2003, p. 303.

[18] A. Wang, A. Mohaisen, and S. Chen, “An adversary-centric behavior modeling of ddos attacks,” in *Proceedings of the 37th IEEE International Conference on Distributed Computing Systems, ICDCS.*, 2017, pp. 1126–1136.

[19] L. Krämer, J. Krupp, D. Makita, T. Nishizoe, T. Koide, K. Yoshioka, and C. Rossow, “Amppot: Monitoring and defending against amplification ddos attacks,” in *Proceedings of the Research in Attacks, Intrusions, and Defenses - 18th International Symposium, RAID.*, 2015, pp. 615–636.

[20] A. Wang, W. Chang, S. Chen, and A. Mohaisen, “Delving into internet ddos attacks by botnets: Characterization and analysis,” *IEEE/ACM Trans. Netw.*, vol. 26, no. 6, pp. 2843–2855, 2018.

[21] T. Cymru. (2019) Cymru. Available at [Online]: https://www.team-cymru.com/.

[22] A. Mohaisen and O. Alrawi, “Av-meter: An evaluation of antivirus scans and labels,” in *Proceedings of the Detection of Intrusions and Malware, and Vulnerability Assessment - 11th International Conference, DIMVA.*, 2014, pp. 112–131.

[23] A. Mohaisen, O. Alrawi, and M. Mohaisen, “AMAL: high-fidelity, behavior-based automated malware analysis and classification,” *Computers & Security.*, vol. 52, pp. 251–266, 2015.

[24] W. Chang, A. Mohaisen, A. Wang, and S. Chen, “Understanding adversarial strategies from bot recruitment to scheduling,” in *Proceedings of the Security and Privacy in Communication Networks - 13th International Conference, SecureComm.*, 2017, pp. 397–417.

[25] MaxMind. (2019) Maxmind. Available at [Online]: https://www.maxmind.com/.

[26] A. Wang, W. Chang, S. Chen, and A. Mohaisen, “A data-driven study of ddos attacks and their dynamics,” *IEEE Transactions on Dependable and Secure Computing*, 2018.

[27] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems.*, 2017, pp. 5998–6008.

[28] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, “Kitsune: An ensemble of autoencoders for online network intrusion detection,” in *Proceedings of the 25th Annual Network and Distributed System Security Symposium, NDSS.*, 2018.

[29] W. Chang, A. Mohaisen, A. Wang, and S. Chen, “Measuring botnets in the wild: Some new trends,” in *Proceedings of the 10th ACM Symposium on Information, Computer and Communications Security, ASIA CCS.*, 2015, pp. 645–650.

[30] G. Gu, R. Perdisci, J. Zhang, and W. Lee, “Botminer: Clustering analysis of network traffic for protocol- and structure-independent botnet detection,” in *Proceedings of the 17th USENIX Security Symposium.*, 2008, pp. 139–154.

[31] R. Perdisci, W. Lee, and N. Feamster, “Behavioral clustering of http-based malware and signature generation using malicious network traces,” in *Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation, NSDI.*, 2010, pp. 391–404.

[32] R. Doshi, N. Apthorpe, and N. Feamster, “Machine learning ddos detection for consumer internet of things devices,” in *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2018, pp. 29–35.

[33] R. Bhatia, S. Benno, J. Esteban, T. Lakshman, and J. Grogan, “Unsupervised machine learning for network-centric anomaly detection in iot,” in *Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks*, 2019, pp. 42–48.

[34] R. Kesavamoorthy and K. R. Soundar, “Swarm intelligence based autonomous ddos attack detection and defense using multi agent system,” *Cluster Computing*, vol. 22, no. 4, pp. 9469–9476, 2019.

[35] C. Fachkha, E. Bou-Harb, and M. Debbabi, “On the inference and prediction of ddos campaigns,” *Wireless Communications and Mobile Computing*, vol. 15, no. 6, pp. 1066–1078, 2015.