

4. Robertson, W., & Vigna, G. (2010). An empirical study of the structural properties of malware families. In Proceedings of the 16th ACM conference on Computer and communications security (pp. 626-638).

OVERVIEW OF ML-BASED BOT DETECTION APPROACHES BASED ON SERVER LOG DATA

Holoborodko B.Y., student,

Suknov M.P., PhD, Associate Professor,

Kharkiv National University of Radio Electronics

A significant portion of the Internet traffic on modern websites is generated by bots, which pose a threat to the security, privacy and performance of websites. To develop effective methods for detecting bots and establishing reliable patterns of real user behavior, it is necessary to be able to distinguish and classify traffic. The complexity of this issue lies in the fact that not all bot traffic is bad. Many systems, such as Google or Bing, crawl and index a website with a good purpose - to understand what is on the web page and show it to interested visitors.

We will look at modern methods of detecting bots on websites, and highlight their advantages and disadvantages. We will only consider server-based models that use HTTP request logs, IP addresses, user agents, and user sessions for training and evaluation. That is, we only consider models and approaches based on server-side information, without analyzing mouse or keyboard keystrokes. In addition, we do not consider models based on information collected via JavaScript, as a large number of users use various ad blockers and other applications, which makes it impossible to combine data from the server and the client's browser.

To begin with, let's focus on an approach based on unsupervised learning. The approach proposed in 2020 [2] is based on the use of unsupervised learning (k-means and graduated probabilistic c-means) followed by supervised cluster labeling. The effectiveness of the method proposed by the authors is evaluated using experiments on real e-commerce data under realistic conditions and compared with the effectiveness of

supervised learning classifiers (multilayer perceptron neural network and support vector machine).

The results show that unsupervised learning-based classification is very effective (almost identical to fully supervised classification), but with a significant advantage - it is less sensitive to incorrectly labeled data. Despite the fact that model building and evaluation are based on ecommerce project data, this approach can be used for any type of website.

Another approach that deserves attention is the so-called concept drift, which allows detecting web robots with drastic changes in characteristics. In article [1] , a new framework for detecting web robots is proposed, consisting of a set of models, a reinforcement learning algorithm that integrates all models, and a concept drift detection module. First, we propose the use of reinforcement learning to integrate a number of models for detecting different types of web robots by dynamically adjusting the model weights. Then, a drift detection method (DDM) is applied to monitor concept drift and determine the need for model retraining over time. This approach is useful because it is very flexible, taking into account dynamic retraining and changes in model weights.

Let's consider another model [3], which was trained on HTTP logs and implements a method of binary classification of request flows to web servers in order to mark the active session as "bot" or "human". For this purpose, a sequential probabilistic analysis approach is used to identify relationships between subsequent HTTP requests in the current session. Note that there is a certain number of sessions that were identified as None. They are less than 2%, which indicates that the previous models were more accurate, although for an accurate assessment, it is necessary to evaluate the results of each of the proposed approaches on the same data.

As we can see, the use of neural networks trained on website log files and server information can be an effective basis for building a model for traffic classification and bot detection. Taking into account the articles reviewed, we note that in further research it is advisable to consider options for building a model that will take into

account both the possible abrupt change in bot behavior and be highly effective for both online and offline evaluation.

References:

1. Chen, Xue & Song, Yang & Xiong, Wei & Lu, Yutao & Wang, Xingen. (2021). Research on Web Robot Detection Technology for Concept Drift. *Journal of Physics: Conference Series*. 2010. 012161. 10.1088/1742-6596/2010/1/012161.
2. Rovetta, Stefano & Suchacka, Grażyna & Masulli, Francesco. (2020). Bot recognition in a Web store: An approach based on unsupervised learning. *Journal of Network and Computer Applications*. 157. 102577. 10.1016/j.jnca.2020.102577.
3. Suchacka, Grażyna & Cabri, Alberto & Rovetta, Stefano & Masulli, Francesco. (2021). Efficient on-the-fly Web bot detection. *Knowledge-Based Systems*. 223. 107074. 10.1016/j.knosys.2021.107074.

THE FRONT-END INDUSTRY

Volokhan V.V., student,

Suknov M.P., PhD, Associate Professor,

Kharkiv National University of Radio Electronics

Websites and applications have become an important part of our daily lives in today's digital era. Do you know how these user-friendly interfaces are created? In this article we will explore the world of front-end development and its role in shaping our online experiences.

Front-end development is the process of creating and implementing the visual and interactive elements of a website or application. It focuses on the user interface and user experience. It makes sure that users can easily navigate and interact with the digital product. It includes a lot of skills, including HTML, CSS, and JavaScript. They are the core building blocks of the front-end development process.

HTML is the backbone of the webpage. It structures the content of a webpage, defining its elements such as headings, paragraphs, and images. CSS controls the style and layout of these elements. It determines colors, fonts, spacing, and visual view of a webpage. JavaScript brings interactivity to the front-end. It enables dynamic behavior