Revista Angolana de Ciências

Vol. 8, Nº 1. e080114. Janeiro – Junho, 2026
(Publicação em Fluxo Contínuo)

Detecção de Bots Baseada em Caracterização de Dados

Bot Detection Based on Data Characterization

Detección de Bots basada en la caracterización de datos

Hélder João Chissingui[1]

RESUMO

Nos últimos anos, mitigar as ameaças de Bots tornou-se uma tarefa desafiadora. Além do enorme impacto das actividades maliciosas perpetradas por Bots, o crescimento do uso da Internet contribuiu bastante para o estado actual. Os danos à infra-estructura informática, as perdas económicas, a insatisfação dos utilizadores humanos em determinados ambientes de prestação de serviços, entre outras actividades, estão directamente associados a Bots maliciosos. O problema torna-se ainda mais complexo porque, em algumas ocasiões, utilizadores humanos utilizam aplicações móveis com suas contas de utilizador para obter privilégios no acesso a determinados serviços de comércio. Ou seja, o nível de sofisticação dos Bots é cada vez mais elevado, o que faz com que, em determinadas circunstâncias, os padrões de actividades humanas apresentam as mesmas características que as actividades de Bots. Com esse nível de desenvolvimento, as tarefas de detecção tornam-se cada vez mais complexas e vitais. Neste estudo, propõe-se uma abordagem de detecção baseada em meta-aprendizagem, que serve de apoio à detecção por meio da caracterização de dados de utilizadores (Bots e Humanos). O processo de caracterização baseia-se em um multiclassificador construído a partir de dados de episódios anteriores, nos quais foi utilizado um classificador baseado em Proactive Forest. Realiza-se uma análise estatística para seleccionar o multiclassificador mais adequado baseando-se nos tipos Bagging, Boosting, Voting e Stacking. Sendo o desempenho, medido pelo percentual de instâncias correctamente caracterizadas, o multiclassificador Voting foi o que melhor se adequou, apresentando uma média de 99,6% de instâncias correctamente caracterizadas.

Palavras-chave: Detecção de Bots; Meta-aprendizagem; Multiclassificadores; Descrição de dados.

RECEBIDO: 15/10/2025
ACEITE: 15/01/2026
PUBLICADO: 27/06/2026

Como citar: Chissingui, H.J. (2026). Detecção de Bots Baseada em Caracterização de Dados. *RAC: Revista Angolana de Ciências*, 8(1), e080114. https://doi.org/10.54580/R0801.14

E-ISSN. 2664-259X

1. Instituto Superior Técnico Militar (ISTM). Luanda. Angola.
hjc@istm-faa.co.ao / https://orcid.org/0000-0002-7538-3865

Abstract

Recent years, mitigating bot threats has become a challenging task. In addition to the enormous impact of malicious activities perpetrated by bots, the growth in internet usage has contributed significantly to the current situation. Damage to IT infrastructure, economic losses, human user dissatisfaction in certain service delivery environments, among other activities, are directly associated with malicious bots. The problem becomes even more complex because, on some occasions, human users use mobile applications with their user accounts to gain access privileges to certain commercial services. In other words, the level of sophistication of bots is increasingly high, which means that, under certain circumstances, human activity patterns exhibit the same characteristics as bot activity. With this level of development, detection tasks become increasingly complex and vital. In this study, we propose a meta-learning-based detection approach that supports detection through the characterization of user data (both bots and humans). The characterization process is based on a multi-classifier built from data from previous episodes, which used a Proactive Forest-based classifier. Statistical analysis is performed to select the most appropriate multi-classifier (Bagging, Boosting, Voting, or Stacking). Performance, measured by the percentage of correctly characterized instances, showed that the Voting multi-classifier performed best, with an average of 99.6% of correctly characterized instances.

Keywords: Bot Detection; Meta-Learning; Multi-Classifiers; Data Description.

Resumen

En los últimos años, mitigar las amenazas de bots se ha convertido en un desafío. Más allá del enorme impacto de las actividades maliciosas perpetradas por bots, el crecimiento del uso de internet ha contribuido significativamente a la situación actual. Daños a la infraestructura de TI, pérdidas económicas e insatisfacción entre los usuarios en ciertos entornos de prestación de servicios, entre otros problemas, están directamente asociados con los bots maliciosos. El problema se vuelve aún más complejo porque, en ocasiones, los usuarios utilizan aplicaciones móviles con sus cuentas de usuario para obtener acceso privilegiado a ciertos servicios de comercio electrónico. En otras palabras, el nivel de sofisticación de los bots es cada vez mayor, lo que significa que, en ciertas circunstancias, los patrones de actividad humana exhiben las mismas características que la actividad de los bots. Con este nivel de desarrollo, las tareas de detección se vuelven cada vez más complejas y vitales. Este estudio propone un enfoque de detección basado en metaaprendizaje que apoya la detección mediante la caracterización de datos de usuarios (bots y humanos). El proceso de caracterización se basa en un clasificador múltiple construido a partir de datos de episodios anteriores, en el que se utilizó un clasificador basado en Proactive Forest. Se realizó un análisis estadístico para seleccionar el multiclasificador más adecuado según los tipos Bagging, Boosting, Voting y Stacking. El rendimiento, medido por el porcentaje de instancias correctamente caracterizadas, mostró que el multiclasificador Voting fue el que mejor se ajustó, con un promedio del 99,6 % de instancias correctamente caracterizadas.

Palabras clave: Detección de bots; Metaaprendizaje; Multiclasificadores; Descripción de datos.

Introdução

O impacto da Internet na sociedade humana está actualmente causando mudanças educacionais, econômicas, sociais e políticas em todo o mundo, confirmando que a facilidade de acesso à Internet é muito importante para o desenvolvimento da sociedade civilizada (Rahman & Tomar, 2020). Um navegador web é uma ferramenta fundamental no processo de exploração da Internet, facilitando no acesso a diferentes aplicações web (Hayawi et al., 2023). Com o objectivo de melhorar as prestações dos sistemas web, são utilizadas aplicações informáticas que se encarregam de automatizar tarefas previamente definidas, uma vez que é possível maximizar a produção e até mesmo reduzir o erro humano. Alguns desses *Softwares* são conhecidos por robôs web, *web crawlers*, *scrappers*, entre outros nomes ou simplesmente *Bots*.

Os *Bots* implementam funcionalidades que os fazem imitar habilidades de utilizadores humanos e até melhoram algumas em determinados contextos (Stassopoulou & Dikaiakos, 2009). Podem pertencer a uma comunidade ou conjunto de *Bots* denominado *Botnet*, a qual é controlada de forma remota pelo *Botmaster* (papel de um utilizador humano) por intermédio de canais de Comando e Controle (*C&C*, do inglês, Command and Control) dedicados, estes últimos baseados fundamentalmente nos protocolos de comunicação: *Internet Relay Chat (IRC)*, *Peer-To-Peer (P2P)* e o Protocolo de Transferência de Hipertexto (HTTP, do inglês, *Hypertext Transfer Protocol*). Podem ser classificados

segundo a finalidade como: (1) benignos, (2) malignos. No caso dos primeiros, desenvolvem-se para executar tarefas como motores de busca e outras que são parte importante dos sistemas. Os segundos são utilizados por atacantes para perpetrar actividades maliciosas (Hayawi et al., 2023).

De acordo com a sua evolução, os *Bots* maliciosos podem ser classificados em: (1) *Bots Simples*, (2) *Bots Moderados*, (3) *Bots Avançados* e (4) *Bots Evasivos*. No caso dos primeiros, caracterizam-se por se conectar a um único endereço IP atribuído pelo Provedor de Serviço de Internet (*ISP*, do inglês, *Internet Service Provider*), usando *scripts* automatizados, não navegadores, disfarçam-se, além de não se identificarem como um navegador. Os segundos possuem uma determinada complexidade, simulam a tecnologia do navegador e podem executar JavaScript. Os terceiros, produzem movimentos do mouse e cliques que enganam até mesmo os métodos de detecção mais sofisticados, imitam os humanos, empregam comportamentos mais evasivos, utilizam *Software* de automação do navegador ou *Malware* instalado em navegadores reais para se conectar aos sites. Os quartos são uma agrupação de *Bots* maliciosos moderados e avançados, tendem a percorrer endereços IP aleatórios, entram através de proxies anónimos e redes P2P, e podem mudar os seus agentes de utilizador, utilizam uma combinação de tecnologias e métodos para evadir a detecção (Imperva, 2022).

Segundo o *Bad Bot Report* de Imperva (2025), o tráfego de *Bots* superou a actividade humana, representando 51% de todo o tráfego na web em 2024. Isso foi amplamente impulsionado pela rápida adopção de Inteligência Artificial (IA) e modelos de linguagem de grande escala (*LLMs*, do inglês, *Large Language Models*), que tornaram a criação de *Bots* mais acessível.

As abordagens de detecção baseadas em Aprendizagem Automática (*ML*, do inglês, *Machine Learning*) se destacam no estado da arte (Chissingui et al., 2022, 2023), em contrapartida, não são projectadas considerando uma etapa prévia de caracterização de dados, a qual, além de apoiar a detecção de *Bots*, pode prevenir ataques a determinados recursos do sistema e reduzindo a superfície de ataque de determinado sistema. Segundo Lorena et al. (2019) a caracterização dos dados pode centrar-se em termos de (1) a ambiguidade das classes; (2) a escassez e dimensionalidade dos dados; e (3) a complexidade da fronteira que separa as classes.

Uma vez identificados os elementos motivacionais e analisado o contexto, identifica-se a seguinte pergunta de partida: Como caracterizar os dados para melhorar o desempenho da detecção de *Bots*?

Para responder a pergunta proposta, o estudo tem como objectivo desenvolver um mecanismo de detecção de *Bots* baseado na caracterização de dados por intermédio de meta-aprendizagem.

O presente estudo permite-nos assimilar abordagens de detecção de *Bots*, bem como as especificidades de implementação de abordagens de aprendizagem supervisionada e meta-aprendizagem, o que constitui o valor teórico do trabalho. O algoritmo desenvolvido e validado pode ser utilizado para apoiar o processo de detecção de *Bots* em plataformas web, o que constitui o valor prático do trabalho. O documento, obedece a seguinte organização: organizado da seguinte forma: A introdução na secção 1, na secção 2 se faz uma descrição dos materiais e da metodologia utilizada, na secção 3 são apresentados os resultados com a correspondente discução na secção 4 e por último as conclusões e recomendações na secção 5, além das referências bibliográficas.

Estado da Arte

No estado da arte empregam-se cinco (5) tipos de abordagens para a detecção de *Bots*: Mineração de dados complexos, Abordagens Distribuídas, Honeypots, Teste de *Turing* público e automático para distinguir computadores de humanos (*CAPTCHA*, do inglês, *Completely Automated Public Turing test to tell Computers and Humans Apart*) e *ML*. A última abordagem é a mais frequente, com maior incidência no uso de algoritmos supervisionados (Chissingui et al., 2022). No entanto, Suchacka & Iwanski (2020) mencionam a importância da abordagem de aprendizagem não supervisionada na detecção de *Bots*, devido a medidas de mascaramento que muitos *Bots* utilizam, para que sejam invisíveis aos sistemas de detecção.

Bots benignos e maliciosos são projectados para propósitos específicos, portanto, a tarefa de detectá-los também possui um nível de especificidade. De modo geral, essa especificidade determina as abordagens de detecção e a selecção de técnicas apropriadas para determinado ambiente. Embora que funcionalmente detectar *TrickBots* em serviços bancários pela internet não seja o mesmo que detectar *SpamBots* em redes sociais, tudo porque uma transação bancária se concretiza em menos que a disseminação de *Fake News*, os algoritmos de *ML* têm sido aplicados em entornos diferenciados. Gezer et al. (2019) emprega aprendizagem supervisionada para detectar transações bancárias fraudulentas por *TrickBots*, cujo processo de classificação das transações é baseado no algoritmo *Random Forests*. Por outro lado, Rovetta et al. (2020) e Suchacka & Iwanski (2020), empregam abordagens de detecção baseada em aprendizagem não supervisionada por meio dos algoritmos de *Graded Possibilistic C-Means (GPCM)* e *Agglomerative Information Bottleneck (AIB)* respectivamente. A utilização da abordagem de aprendizagem não supervisionada é motivada pelo facto de, na vida real, vários *Bots* empregam camuflagem, portanto, tendo um conjunto de dados históricos de sessões de utilizadores, algumas sessões de *Bots* podem ser rotuladas incorrectamente como se geradas por humanos (Suchacka & Iwanski, 2020).

A complexidade da identificação do perfil comportamental dos utilizadores humanos e de *Bots* reduz o desempenho das abordagens de *ML* convencionais. A mineração de dados complexos como dados multimédia, espaciais, de séries temporais, texto e outros tipos de dados complexos surge como uma importante alternativa (Chissingui et al., 2022). A materialização dessa abordagem é fundamentalmente baseada em algoritmos de aprendizagem profunda, com destaque das redes neurais profundas. A mineração de sequência é uma das abordagens de destaque, Cresci et al.

(2018) empregam sequências de *DNA-Digital* para detectar *SPAMBOTS* em redes sociais, enquanto Suchacka et al. (2021) utilizam cadeia de tempo discreto de Marcov (*DTMC*, do inglês, *Discrete-Time Marcov Chain*). A mineração em grafos figura nesta abordagem devido suas especificidades no quesito mineração de dados; Rheault & Musulan (2021) apresentam uma proposta de detecção de comunidades *online* baseada em *Uniform Manifold Aproximation and Projection (UMAP)*.

As abordagens de detecção distribuídas, surgem com o propósito de combater a resiliência dos ataques de *Bots*. Baseiam-se no princípio de que o processo de detecção pode ser realizado por detectores distribuídos de forma lógica ou física, resultando na redução da superfície de ataque. Estas abordagens são utilizadas para projectar Sistemas Coperativos de Detecção de Intrusões (*CIDS*, do inglês, *Cooperative Intrusion Detection Systems*) utilizando técnicas como *Blockchain* aplicadas nos estudos de Venkatesan et al. (2016) e Alkadi et al. (2021). Quando as configurações de rede ou sistemas são estáticas, isto constitui uma vulnerabilidade. As abordagens distribuídas também se destacam nestes ambientes, pois permitem o planeamento de estratégias dinâmicas para alterações nas configurações do sistema. Nos estudos Albanese et al. (2018); Zha et al. (2019), a técnica de Defesa de Alvo Móvel (*MTD*, do inglês, *Moving Target Defense*) é utilizada para alterações periódicas nas configurações do detector, enquanto nos estudos de Chen et al. (2020); Maeda et al. (2019), a abordagem de Rede Definida por *Software* (SDN) é utilizada para o controlo de rede flexível e dinâmico.

As abordagens baseadas no *Test Turing* centram-se nos princípios definidos na pesquisa de *Turing* (1950), resumem-se no Processamento de Linguagem Natural (*NLP*, do inglês, *Natural Language Processing*), representação do conhecimento, raciocínio e aprendizado de máquina. Para a detecção de *Bots*, uma maneira de implementar a abordagem é desenvolver um conjunto de testes *online* para classificar os utilizadores com base em seus resultados. Um teste muito comum em aplicações *web* é o *CAPTCHA*; actualmente, existem variantes aprimoradas desse teste, como o *reCAPTCHA* proposto por Ahn et al. (2008) (actualmente na versão 3 e de propriedade da Google), o *BeCAPTCHA* proposto por Acien et al. (2021) e o *CAPTURE* em Hitaj et al. (2020).

Honeypots é uma abordagem que pode ser utilizada para detecção de *Bots online* devido aos seus padrões de funcionalidade. Considerada em Han et al. (2012) como a melhor técnica de Sistemas de Detecção de Intrusões (*IDS*, do inglês, *Intrusion Detection Systems*), a abordagem consiste em criar um sistema com certas vulnerabilidades atractivas para *Bots*. A detecção depende da atractividade dos recursos estabelecidos, como computadores, dados, parte de uma rede, aplicações *web*, etc.

As abordagens baseadas em ML são de múltiplos propósitos, uma vez que se enquadram nas abordagens de detecção descritas nos parágrafos anteriores. Apesar da frequência de implementação destas, quando implementadas de forma convencional (baseadas em características simples) seu desempenho tende a reduzir, tal como Hayawi et al. (2023) sugere que os modelos baseados em características, como os previstos no estudo de Varol et al. (2017), podem nem sempre ser os mais adequados, por exemplo, para detectar ataques coordenados. Já as abordagens baseadas em mineração de dados complexos são muito específicas para cada contexto, pois, muitas delas centram-se na geração de características de fontes de dados complexos, o que introduz alguma componente adicional de processamento, com implicações ao nível da infra-estructura de cômputo necessária.

Material e Métodos

A proposta da presente investigação é elaborada com base no fluxo da Figura 1 (com destaque das actividades coloridas a verde). Dado o conjunto de dados CTU-13, uma vez pré-processados os dados (particionados segundo a representação da Figura 2), o conjunto de treinamento é usado para a Criação da Base de Factos, tendo este processo como saída um meta-dataset, que é a Base de Factos propriamente dita. A construção do modelo multiclassificar vem a seguir, tendo como alimentação principal a Base de Factos (dados de descrição para conjuntos de *Bots* e conjuntos sem *Bots*).

Para a segunda etapa, após o pré-processamento dos dados de entrada, uma vez desconhecidos os rótulos dos exemplos de dados, é realizada a rotulagem dos dados. As medidas de complexidade de classificação da Tabela 4 correspondentes à amostra ou conjunto seleccionado são calculadas utilizando os resultados da rotulagem. As medidas de complexidade de classificação calculadas são classificadas pelo modelo multiclassificador criado anteriormente, resultando num rótulo que indicará se o conjunto contém exemplos de *Bots* ou é composto apenas por utilizadores humanos.

Descrição dos dados

O CTU-13 é um conjunto de dados de tráfego de *Botnet* que foi capturado na Universidade CTU, na República Tcheca, em 2011. O objectivo do conjunto de dados era ter uma grande captura do tráfego real de *Botnet* misturado com tráfego normal e tráfego em segundo plano. O conjunto de dados CTU-13 consiste em treze capturas (chamadas de cenários) de diferentes amostras de *Botnet*. Possui 15 atributos (ver Tabela 1) em cada cenário, se executa um *Malware* específico, que usava vários protocolos e realizava diferentes acções. A Figura 2 mostra as características dos cenários de *Botnet*. Cada cenário foi capturado em um arquivo de extesão .pcap que contém todos os pacotes dos três tipos de tráfego. A característica distintiva do conjunto de dados CTU-13 é que analisamos e rotulamos manualmente cada cenário. O processo de rotulagem foi feito dentro dos arquivos do NetFlows (Garcia et al., 2014).

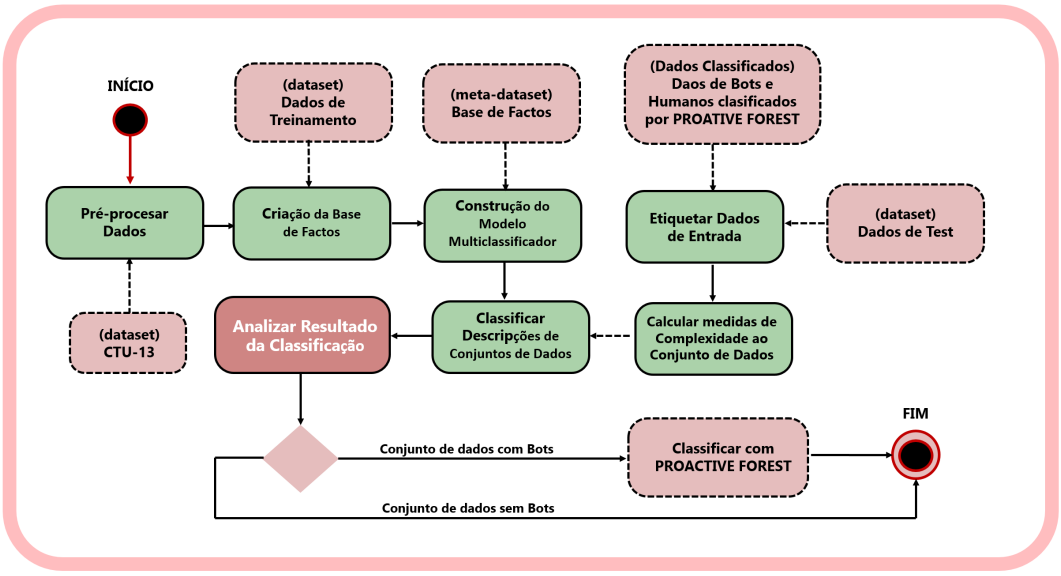

Figura 1. Fluxo de actividades.
Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Tabela 1.
Atributos ou características do Dataset CTU-13.

ATRIBUTO	DESCRIÇÃO	TIPO
Dir	Sentido da conexão	Categórico
Dport	Porta Destino	Categórico
DstAddr	Endereço IP destino	Categórico
Dur	Duração da conexão	Numérico
Proto	Protocolo de capa de transporte empregue na conexão	Categórico
Sport	Porta origem	Categórico
SrcAddr	Endereço IP origem	Categórico
SrcBytes	Quantidade de bytes enviados desde a origem	Numérico
StartTime	Timestamp correspondente ao inicio da conexão	Datetime
State	Estado da conexão	Categórico
TotBytes	Quantidade de bytes transmitidos através da conexão	Numérico
TotPkts	Quantidade de pacotes transmitidos através da conexão	Numérico
dTos	Tipo de serviço do destino	Numérico
sTos	Tipo de serviço da origem	Numérico
Label	Classe do fluxo (Botnet, Normal, Background)	Categórico

Fonte: (Garcia et al., 2014).

Table 2 – Characteristics of the botnet scenarios. (CF: ClickFraud, PS: Port Scan, FF: FastFlux, US: Compiled and controlled by us.)

Id	IRC	SPAM	CF	PS	DDoS	FF	P2P	US	HTTP	Note
1	√	√	√							
2	√	√	√							
3	√			√				√		
4	√				√			√		UDP and ICMP DDoS.
5		√		√					√	Scan web proxies.
6				√						Proprietary C&C. RDP.
7									√	Chinese hosts.
8				√						Proprietary C&C. Net-BIOS, STUN.
9	√	√	√	√						
10	√				√			√		UDP DDoS.
11	√				√			√		ICMP DDoS.
12							√			Synchronization.
13		√		√					√	Captcha. Web mail.

Figura 2. Características dos cenários de CTU-13.
Fonte: (Garcia et al., 2014).

Pré-processamento dos Dados

Para adequar os dados ao processo de mineração, foram aplicados um conjunto de procedimentos com vista a eliminar irregularidades, inconsistências e ruído nos dados, converter, escalar e reduzir dimensionalidade. Como o conjunto de dados possui instâncias de classe de tráfego de *Botnet*, *Normal* e de *Background*, as duas últimas foram tratadas como uma única classe, uma vez que, a classe *Normal* corresponde ao tráfego de utilizadores humanos e a classe *Background*, ao tráfego próprio da rede quando não existem actividades humanas e de *Bots*. Se pode observar desde a Tabela 2, que os cenários de CTU-13 têm um desequilíbrio de classes severo, facto que motivou a adopção de uma conduta conservadora no tratamento das instâncias com valores faltantes, substituindo-os com o valor 0, evitando assim a perda de informação, fundamentalmente para a classe minoritária. A redução de dimensionalidade foi realizada por intermédio da técnica de Análise de Componentes Principais (*PCA*, do inglês, *Principal Component Analysis*) considerando uma variância explicada acumulada mínima correspondente a 99%, resultando em 7 componentes principais para todos os cenários. Na Tabela 3, podem ser visualizados os resultados das transformações.

Tabela 2.
Dimensões dos cenários de CTU-13 antes do pré-processamento.

CENÁRIO	BOTNET (1)	NORMAL (0)	TOTAL DE INSTÂNCIAS	ATRIBUTOS	PERCENTAGEM DA CLASSE MINORITÁRIA
1	40961	2783675	2824636	14	1.45 %
2	20941	1787181	1808122	14	1.15 %
3	26822	4683816	4710638	14	0.57 %
4	2580	1118496	1121076	14	0.23 %
5	901	128931	129832	14	0.70 %
6	4630	554289	558919	14	0.83 %
7	63	114014	114077	14	0.05 %
8	6127	2948103	2948103	14	0.20 %
9	184987	1902521	2087508	14	8.45 %
10	106352	1203439	1309791	14	8.11 %
11	8164	99087	107251	14	7.61 %
12	2168	323303	325471	14	0.67 %
13	40003	1885146	1925149	14	2.07 %

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Tabela 3.
Dimensões dos cenários de CTU-13 após o pré-processamento.

CENÁRIO	LIMPEZA DE DADOS	TRANSFORMAÇÃO E SELECÇÃO			
		COMPONENTES PRINCIPAIS	EQUILÍBRIO DE CLASSES		
	INSTÂNCIAS		BOTNET (1)	NORMAL (0)	TOTAL
1	2824636	7	2783675	2783675	5567350
2	1808122	7	1787181	1787181	3574362
3	4710638	7	4683816	4683816	9367632
4	1121076	7	1118496	1118496	2236992
5	129832	7	128931	128931	257862
6	558919	7	554289	554289	1108578
7	114077	7	114014	114014	228028
8	2948103	7	2948103	2948103	5896206
9	2087508	7	1902521	1902521	3805042
10	1309791	7	1203439	1203439	2406878
11	107251	7	99087	99087	198174
12	325471	7	323303	323303	646606
13	1925149	7	1885146	1885146	3770292

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Equilíbrio de classes

Segundo Han et al. (2012) os multiclassificadores funcionam relactivamente bem para o problema de desequilíbrio de classes, fundamentalmente em classificação binária, pois, os classificadores individuais que compõem o conjunto podem incluir versões de abordagens de sobreamostragem e ajuste do limiar. No entanto, a classe positiva (classe de tráfego de *Botnet*) é minoritária, com a máxima percentagem inferior a 10%. Tendo em conta a severidade do desequilíbrio de classes dos cenários de CTU-13, foi implementada a técnica de sobreamostragem sintética da classe minoritária (*SMOTE*, do inglês, *Synthetic Minority Over-sampling Technique*). Na Tabela 3 se pode observar as transformações realizadas no conjunto de dados.

Criação da Base de Factos e divisão dos dados

A Base de Factos representa os diferentes episódios de aprendizagem pelos quais o algoritmo *Proactive Forest* passou. É composta por meta-conhecimento que descreve os dados dos utilizadores Bot e Humanos descritos por intermédio das medidas de complexidade de classificação da Tabela 4. Basicamente, a construção da Base de Factos foi realizada nos passos seguintes:

1. Com base a Figura 3, 70% do conjunto de dados foram seleccionados para o treinamento.
2. Os dados de treinamento seleccionados no ponto anterior, são novamente divididos de forma estratificada, considerando o equilíbrio de classe alcançado e a quantidade de dados disponíveis, 50% dos dados é utilizado para a construção e validação do modelo classificador baseado no algoritmo *Proactive Forest*.
3. Uma vez validado o modelo classificador ora construído, com os dados já classificados por *Proative Forest*, são usados para contruir árvores de decisão para o processo de etiquetagem.
4. A outra metade do conjunto de treinamento (50%) passa por um processo de amostragem aleatória, onde são constituidos subconjuntos com instâncias de *Bots* e sem instâncias de *Bots* (subconjuntos de utilizadores humanos).

Os referidos subconjuntos são classificados pelo modelo de árvore de decisão, posteriormente calculadas as medidas de complexidade de classificação com a consequente atribuição da etiqueta 1 se o subconjunto tiver pelos uma instância de *Bot* e 0 se formado apenas por instâncias de utilizadores humanos. Este processo decorre para cada subconjunto criado, tendo no final a descrição de todos os subconjuntos com as respectivas etiquetas, constituindo desta forma a Base de Factos.

Uma vez que o ceneários 3 e 11, prefazem um valor próximo aos 30% do conjunto de dados, foram seleccionados como a parte dos dados reservada para o teste. Por outro lado, tal selecção tem alguma implicação com a geralização dos modelos ora criados, pois, os cenários de CTU-13 não têm o mesmo tipo de tráfego de *Botnet*. Os cenários seleccionados foram fundidos para o posterior emprego, em um processo aleatório de amostragem para a conformação de 100 subconjuntos de dados com Bots e Humanos para a correspondente caracterização.

Construção de Multiclassificadores

Neste estudo, foram consideradas as quatro (4) variantes de métodos de combinação de classificadores, apresentadas na Tabela 5. A Figura 1, representa um fluxo de trabalho genérico para a construção de multiclassificadores, tendo em conta cada método seleccionado. Para este efeito, foram utilizados os algoritmos supervisionados da Tabela 6. A construção do modelo multiclassificador baseia-se num processo de treino a que o algoritmo seleccionado é submetido, utilizando a Base de Factos e os algoritmos supervisionados. Cada método de combinação de classificadores tem as suas próprias particularidades que influenciam os processos de criação (Duin, 2002).

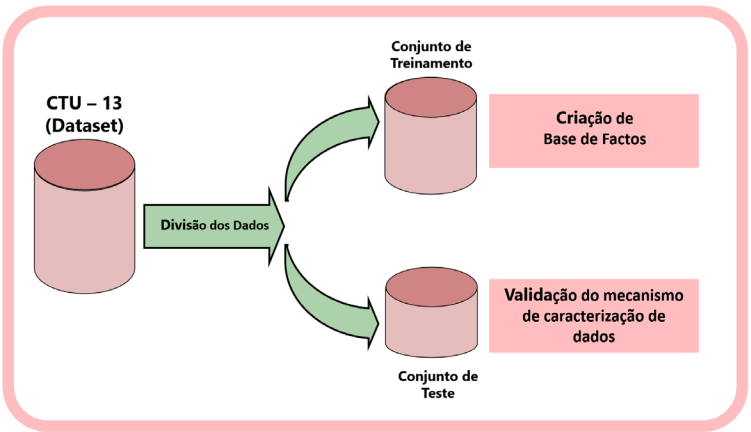

Figura 3. Divisão dos dados para treino (utilizado para construção de modelo classificador com Proactive Forest e criação da base de factos) e teste (para validação da proposta de caracterização dos dados).
Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Tabela 4.
Medidas de Complexidade de Classificaçãoento.

N	MEDIDA	ID	TIPO
01	Relacção máxima do discriminante de Fisher	F1	**Basadas em Características**
02	Volume da região sobreposta	F2	
03	Máxima eficiência de funções individuais	F3	
04	Eficiência de características colectivas	F4	
05	Soma da distância do Erro por Programação Linear	L1	**Linearidade**
06	Taxa de erro do classificador linear	L2	
07	Não linearidade de um classificador linear	L3	
08	Fracção de pontos limite	N1	**Vizinhança**
09	Proporção de distância do vizinho mais próximo	N2	
10	Taxa de erro do classificador vizinho mais próximo	N3	
11	Não linearidade do classificador vizinho mais próximo	N4	
12	Fracção de hiper-esferas que cobrem os dados	T1	**Dimensionalidade**
13	Valor médio de características por pontos	T2	
14	Valor médio de dimensões de PCA	T3	
15	Relação entre a dimensão PCA e a dimensão original	T4	
16	Entropía de proporsões de classe	C1	**Desequilíbrio de classe**
17	Relação de desequilíbrio	C2	

Fonte: (Lorena et al., 2019).

Tabela 5.
Medidas de Complexidade de Classificaçãoento.

TIPO	MÉTODO	IDENTIFICADOR
Homogéneos	Bagging	BG
	Boosting	BS
Hibridos	Stacking	STK
	Voting	VTG

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Tabela 6.
Algoritmos de Aprendizagem supervisionado seleccionados segundo a frequência de seu emprego no estado da arte (Chissingui et al., 2022; Karataş & Şahin, 2017; Latah, 2020; Orabi et al., 2020).

CATEGORIA DO ALGORITMO	VARIANTE	IDENTIFICADOR
Modelos Lineares	Regresão Logística	LR
	SGD	SGD
Árvore de Decisão	ExtraTree	EXTRA
	Cart	CART
Bayesiano	Naives Bayes	NAIVE
Baseado em Instâncias	Vizinhos mais próximos	KNN
Máquina de Suporte Vectorial	SVC(*Kernel*=RBF)	SVM
Rede Neural	Perceptron Multi-capas	MLP

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Caracterização dos Dados

O processo de detecção de potenciais utilizadores de *Bots* envolve a análise de um conjunto de utilizadores de cada vez. Para descrever estes utilizadores, são utilizadas dezassete (17) medidas de complexidade de classificação mencionadas na Tabela 4. Estas medidas dependem dos rótulos das classes. Por isso, antes da descrição, os dados são submetidos ao processo de rotulagem. Considerando a possibilidade de um elevado fluxo de utilizadores aceder ao sistema ao mesmo tempo, para além da sua influência no cálculo das medidas de complexidade de classificação, a etiquetagem requer rapidez. Uma vez que o processo mencionado tem uma influência considerável na caracterização dos conjuntos de dados, é necessário que os erros de previsão (falsos positivos e negativos) sejam os menores possíveis. Para cumprir os requisitos mencionados, é utilizado um modelo classificador baseado em Árvores de Decisão, construído com dados previamente classificados pelo *Proactive Forest*, algoritmo que apresenta melhores resultados de diversidade em relação ao *Random Forest*, conforme estudo Cepero-Pérez et al.(2018).

A Figura 4 representa o fluxo de actividades para rotular ou etiquetar os dados de entrada. Utilizando dados previamente classificados pelo *Proactive Forest*, é criada uma árvore de decisão, esta é utilizada para rotular os dados de entrada. Após a conclusão do fluxo de actividades da Figura 4, os dados estão prontos para o cálculo das medidas de complexidade de classificação da Tabela 4.

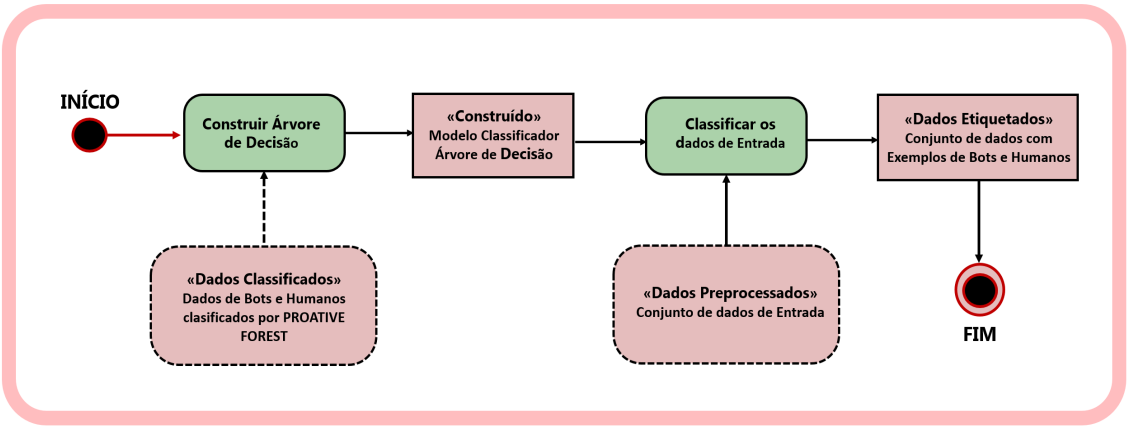

Figura 4. Actividades relacionadas ao processo de etiquetagem dos dados de entrada.
Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Descrição dos experimentos

Dois experimentos factoriais completos foram conduzidos, ambos com o objectivo de avaliar como os factores seleccionados influenciam a percentagem de instâncias correctamente caracterizadas para os multiclassificadores homogêneos e híbridos. A classificação dos factores e a determinação dos níveis são apresentadas na Tabela 7.

Tabela 7.
Planificação dos experimentos.

EXPERIMENTO 1 - PARA MODELOS HOMOGÉNEOS				
Rendimento : % de instâncias correctamente caracterizadas (% de ICC)				
Factores	**Tipo**	**Nível Alto**	**Nível Baixo**	**Identificador**
Tipo de Modelo	Não qualitativo	Bagging	Adaboost	A
Conjunto de Dados	Não qualitativo	Humanos	Bots	B
RENDIMENTO : % DE INSTÂNCIAS CORRECTAMENTE CARACTERIZADAS (% DE ICC)				
Rendimento : % de instâncias correctamente caracterizadas (% de ICC)				
Factores	**Tipo**	**Nível Alto**	**Nível Baixo**	**Identificador**
Tipo de Modelo	Não qualitativo	Voting	Stacking	A
Conjunto de Dados	Não qualitativo	Humanos	Bots	B

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Configuração dos Experimentos

Na tabela 8 são apresentadas as características do ambiente de desenvolvimento para a implementação desta pesquisa. Baseado em Python, se podem destacar as *Scikit-learn* (Pedregosa et al., 2011) e *Problexity* (Komorniczak & Ksieniewicz, 2022).

Tabela 8.
Características do Ambiente de implementação da pesquisa.

SISTEMA OPERATIVO	PROCESSADOR	MEMÓRIA RAM	SUPORTE
Windows 10	Intel(R) Core (TM) i5-3320M	6GB	Anaconda Scikit-learn Problexity

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Resultados

Análise Estatística

Esta secção apresenta os resultados obtidos a partir da análise estatística com as métricas de desempenho, que determinarão a escolha do algoritmo mais adequado ao problema, em cada métrica de desempenho utilizada.

Comparação dos multiclassificadores homogéneos

Para avaliar os multiclassificadores homogêneos (*Bagging* e *Adaboosting*), foram realizadas 10 execuções, analisando 100 conjuntos de dados de diferentes tamanhos, com e sem *Bots*, mas no mesmo banco de dados. Os dados foram seleccionados aleatoriamente. Os resultados são obtidos pela média das métricas de cada multiclassificador (dependendo do tipo de algoritmo usado como estimador) que compõe *Bagging* e *Adaboosting*. Os resultados são apresentados na Tabela 9. Primeiramente, foi realizado o teste de normalidade de *Shapiro-Wilk*, como resultado, todas as amostras analisadas seguiam uma distribuição normal. Pois, o valor de p-value é sempre superior ao nível de significância de α = 0,05. A única amostra para a qual nenhum teste é realizado ou comparado é a métrica de precisão, uma que no *Bagging*, todos os resultados têm o valor 1, que é o valor máximo da métrica, valor que o *Adaboosting* não atinge em nenhuma de suas execuções.

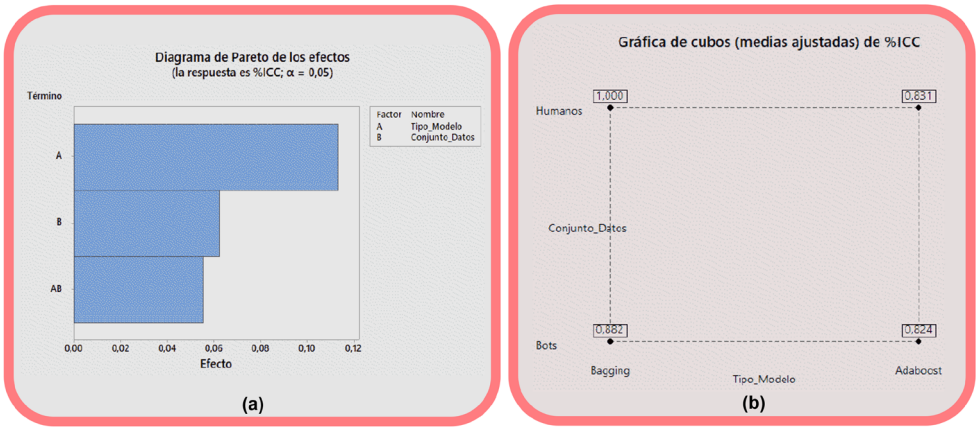

Figura 5. (a) Diagrama de pareto de efeitos e (b) Gráficos de Cubo da Percentagem de instâncias correctamente caracterizadas para os Modelos Homogéneos.
Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Como a amostra segue uma distribuição normal, testes paramétricos devem ser realizados. Foi aplicado o teste *t-student* para comparar as médias das métricas (Acurácia, *Recall*, F1, AUC) dos dois tipos de multiclassificadores. Em cada comparação, se avalia a existência de diferenças significativas nas médias e qual delas apresenta melhor desempenho. Por isso são aplicadas as hipóteses das equações (1) e (2).

$$H_0 : \mu_{BG} - \mu_{BS} = 0 \quad (1)$$

$$H_1 : \mu_{BG} - \mu_{BS} > 0 \quad (2)$$

Como resultado, o *p-value = 0, ou seja*, foi sempre menor que o valor de significância α = 0,05, rejeitando-se a hipótese nula, e pode-se afirmar que há diferenças significativas nos resultados das métricas desses multiclassificadores. Os multiclassificadores de *Bagging* apresentaram desempenho significativamente melhor.

Tabela 9.
Médias das métricas de rendimento para os modelos Bagging e Adaboosting nas 10 execuções.

EXECUÇÕES	BAGGING (BG)					ADABOOST (BS)				
	Acc	Pre	Rec	F1	AUC	Acc	Pre	Rec	F1	AUC
1	0.942	1.00	0.885	0.914	0.943	0.747	0.617	0.668	0.615	0.747
2	0.946	1.00	0.891	0.915	0.946	0.747	0.783	0.67	0.618	0.747
3	0.946	1.00	0.893	0.92	0.946	0.743	0.612	0.67	0.618	0.743
4	0.94	1.00	0.883	0.909	0.94	0.748	0.65	0.673	0.623	0.748
5	0.944	1.00	0.889	0.915	0.945	0.752	0.695	0.673	0.625	0.752
6	0.951	1.00	0.901	0.928	0.951	0.748	0.647	0.672	0.622	0.748
7	0.943	1.00	0.886	0.914	0.943	0.75	0.823	0.675	0.627	0.75
8	0.94	1.00	0.881	0.906	0.94	0.747	0.805	0.673	0.625	0.747
9	0.944	1.00	0.886	0.909	0.944	0.745	0.617	0.67	0.618	0.745
10	0.945	1.00	0.889	0.915	0.945	0.743	0.613	0.67	0.618	0.743

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Comparação dos multiclassificadores híbridos

Aplicando o mesmo de análise dos multiclassificadores homogêneos. Foram realizadas as 10 execuções para avaliação dos modelos *Stacking* e *Voting*, onde 100 conjuntos de dados de diferentes tamanhos foram analisados, estes contendo ou não dados de *Bots*, mas pertencendo ao mesmo banco de dados. Os dados foram seleccionados aleatoriamente. Os resultados são obtidos pela média dos resultados das métricas de cada multiclassificador, como descritos na Tabela 10.

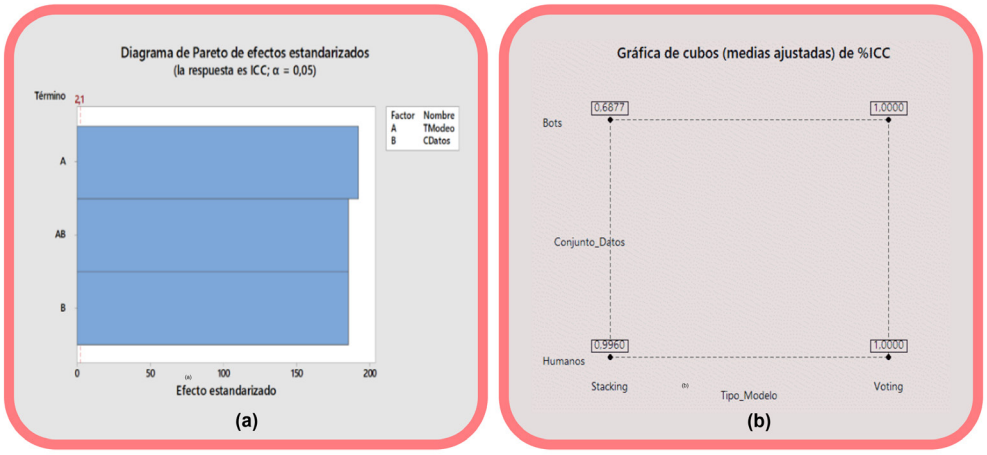

Figura 6. (a) Diagrama de pareto de efeitos e (b) Gráficos de Cubo da Percentagem de instâncias correctamente caracterizadas para os Modelos híbridos.
Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Utilizando o teste de normalidade de *Shapiro-Wilk*, constatou-se que todas as amostras analisadas seguiram uma distribuição normal, pois *p-value* é sempre maior que o nível de significância α = 0,05. A única amostra para a qual nenhum teste ou comparação é realizada é a métrica Precisão no rendimento do modelo *Voting* tem resultado 1 em todas as execuções, valor máximo que não é atingido pelo modelo *Stacking*.

$$H_0 : \mu_{BG} - \mu_{BS} = 0 \quad (3)$$

$$H_1 : \mu_{BG} - \mu_{BS} > 0 \quad (4)$$

Tal como nos modelos homogéneos, a amostra dos híbridos segue uma distribuição normal, pelo que, se adopta o mesmo procedimento, onde as hipóteses aplicadas correspondem as equações (3) e (4). Como resultado, o *p-value = 0*, ou seja, foi sempre menor que o valor de significância α = 0,05, rejeitando-se a hipótese nula, e pode-se afirmar que há diferenças significativas nos resultados das métricas desses multiclassificadores. Os multiclassificadores de votação apresentaram desempenho significativamente superior.

Tabela 10.
Médias das métricas de rendimento para os modelos *Stacking* e *Voting* nas 10 execuções.

EXECUÇÕES	STACKING (STK)					VOTING (VTG)				
	Acc	Pre	Rec	F1	AUC	Acc	Pre	Rec	F1	AUC
1	0.843	0.733	0.685	0.707	0.843	0.994	1.00	0.988	0.993	0.994
2	0.853	0.866	0.709	0.725	0.853	0.997	1.00	0.993	0.997	0.997
3	0.843	0.733	0.696	0.712	0.865	0.995	1.00	0.991	0.995	0.995
4	0.841	0.733	0.693	0.712	0.841	0.995	1.00	0.99	0.995	0.995
5	0.851	0.733	0.699	0.715	0.851	0.996	1.00	0.991	0.995	0.996
6	0.852	0.733	0.707	0.72	0.852	0.997	1.00	0.993	0.997	0.997
7	0.843	0.85	0.693	0.715	0.843	0.995	1.00	0.989	0.994	0.995
8	0.845	0.861	0.696	0.717	0.845	0.995	1.00	0.99	0.995	0.995
9	0.849	0.733	0.707	0.72	0.849	0.997	1.00	0.993	0.997	0.997
10	0.848	0.733	0.699	0.715	0.848	0.996	1.00	0.991	0.995	0.996

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Multiclassficadores homogéneos versus híbridos

Nas comparações estatísticas dos modelos homogêneos e híbridos nas secções anteriores, são mencionados os modelos que apresentaram melhor desempenho.

Em seguida, são comparados o melhor multiclassificador híbrido (*Voting*) e o melhor multiclassificador homogêneo (*Bagging*). O objectivo é determinar qual tem a maior influência nos resultados das métricas de avaliação. Como resultado do teste *t-student*, *p-value* = 0, inferior ao valor de significância α = 0,05. Portanto, a hipótese nula é rejeitada, podendo-se afirmar que há diferenças significativas nos resultados métricos para os multiclassificadores híbridos e homogêneos. Os multiclassificadores híbridos apresentam desempenho significativamente superior. Na Figura 7 representam-se as métricas de desempenho de classificação dos modelos, homogêneo e híbrido, utilizando um gráfico de barras.

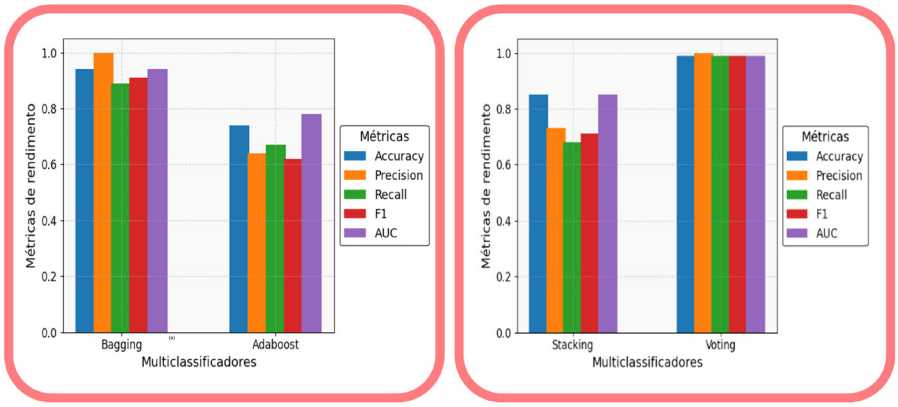

Figura 7. Comparação das valor médio das métricas de desempenho dos modelos multiclassificadores.
Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Tabela 11.
Média das métricas dos modelos homogéneos (*Voting*) e híbridos (*Bagging*).

EXECUÇÕES	HÍBRIDO (VOTING, VTG)					HOMOGÊNEO (BAGGING, BG)				
	Acc	Pre	Rec	F1	AUC	Acc	Pre	Rec	F1	AUC
1	0.994	1.00	0.988	0.993	0.994	0.942	1.00	0.885	0.914	0.943
2	0.997	1.00	0.993	0.997	0.997	0.946	1.00	0.891	0.915	0.946
3	0.995	1.00	0.991	0.995	0.995	0.946	1.00	0.893	0.92	0.946
4	0.995	1.00	0.99	0.995	0.995	0.94	1.00	0.883	0.909	0.94
5	0.996	1.00	0.991	0.995	0.996	0.944	1.00	0.889	0.915	0.945
6	0.997	1.00	0.993	0.997	0.997	0.951	1.00	0.901	0.928	0.951
7	0.995	1.00	0.989	0.994	0.995	0.943	1.00	0.886	0.914	0.943
8	0.995	1.00	0.99	0.995	0.995	0.94	1.00	0.881	0.906	0.94
9	0.997	1.00	0.993	0.997	0.997	0.944	1.00	0.886	0.909	0.944
10	0.996	1.00	0.991	0.995	0.996	0.945	1.00	0.889	0.915	0.945

Fonte: Elaboração própria do autor a partir de dados da pesquisa.

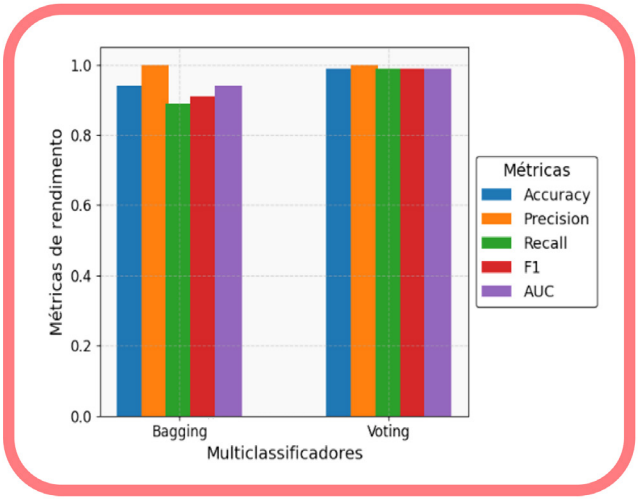

Figura 8. Comparação das valor médio das métricas de desempenho dos melhores modelos multiclassificadores.
Fonte: Elaboração própria do autor a partir de dados da pesquisa.

Tabela 12
Comparação as abordagens do estudo da arte.

REFERÊNCIA	ABORDAGEM	ALGORITMO	MÉTRICAS
Cresci et al. (2018)	Mineração de sequências	DNA-Digital	P=0.95, A=0.95, F1=0.95, R=0.96
Gezer et al. (2019)	Aprendizagem Supervisionada	Random Forests	A = 0.99
Suchacka & Iwanski (2020)	Aprendizagem não Supervisionada	Algoritmo de Gargalo de Informação Aglomerativo	Para k=139, P=0.98,F1=0.98, R = 0.99
Alkadi et al. (2021)	CIDS, Aprendizagem Profundo	Redes Neurais Recorrentes	A=0.99
Presente Estudo	Meta-Aprendizagem, Aprendizagem supervisionada	Multiclassificadores hibridos	A=0.99, F1=0.99, R=0.99, Au=0.99, P=1

Nota: A:Acurácia, P:Precisão, F1:F1-score, R: Recall, Au:AUC

Discussão

Os diagramas de Pareto nos dois experimentos (Figuras 5 e 6) mostram que os factores seleccionados influenciam significativamente a percentagem de instâncias classificadas correctamente, mas para modelos homogêneos, o factor tipo de modelo é mais significativo, enquanto para modelos híbridos, os factores e suas interações têm níveis aproximados de significância na percentagem de instâncias correctamente caracterizadas.

Para determinar quais níveis de factores influenciam a maximização do desempenho, os diagramas de cubo fornecem informações interessantes. A Figura 5 mostra que o desempenho máximo é alcançado ao utilizar o multiclassificador *Bagging* e conjuntos de dados humanos para modelos homogêneos. Para modelos híbridos, a Figura 6 mostra comportamentos diferentes, onde para o multiclassificador *Voting*, a percentagem de instâncias correctamente caracterizadas atinge seu valor máximo em ambos os conjuntos de dados (*Bots* e Humanos). Considerando o objectivo dos experimentos e a análise estatística, ambos os testes mostram uma correlação marcante em seus resultados, mesmo tendo sido realizados com métodos diferentes. Além disso, nos experimentos, a avaliação é realizada considerando apenas a percentagem de instâncias correctamente caracterizadas.

A análise estatística realizada com base na comparação dos valores das métricas de desempenho, concentrou-se em testes estatísticos de normalidade e testes paramétricos aplicados aos valores das métricas nas 10 execuções do multiclassificador. Os resultados da comparação dos modelos homogêneos demonstram que, com base nos valores das métricas de desempenho, há uma diferença significativa entre os valores das métricas para os dois multiclassificadores, o que valida a selecção do método com os melhores resultados, que neste caso é o multiclassificador *Bagging*. Da mesma forma, para os modelos híbridos, o *Voting* obteve os melhores resultados.

Para a abordagem proposta neste trabalho, e com base no que foi apresentado, é necessária uma análise detalhada das métricas F1-*score* e *Recall*, pois ambas lidam com falsos positivos e falsos negativos, dois tipos de erros que afectam a disponibilidade do serviço. Os multiclassificadores *Bagging*, *Adaboost* e *Stacking* apresentam comportamento instável nas métricas mencionadas, um indicador importante, visto que há evidências de conjuntos

de *Bots* sendo classificados como humanos ou vice-versa. Com base nisso, o resultado do algoritmo *Voting* é o mais estável, pois é evidente que as métricas são separadas por um intervalo muito pequeno.

Na Tabela 12 se pode verificar que os valores alcançados pelas abordagens do estado da arte estão próximos e noutra casos superados pela abordagem proposta neste estudo.

A abordagem proposta afecta a disponibilidade de serviço de um sistema com base em falsos positivos. Além disso, o facto de a caracterização dos dados ser realizada considerando um conjunto de instâncias de utilizador, pode evitar a sobrecarga do sistema em cenários onde um número excessivo de solicitações de utilizadores ao sistema é evidente.

Conclusões

A pesquisa realizada permite-nos chegar às seguintes conclusões:

- Embora a abordagem de aprendizagem automática seja a mais utilizada para a detecção de *Bots*, a sua exploração requer uma actualização constante dos modelos e a utilização de novos dados disponíveis ou capturados no sistema. Ao mesmo tempo, o conjunto de características que podem discriminar entre o comportamento humano e o comportamento cada vez mais complexo dos *Bots* deve ser continuamente evoluído.
- A meta-aprendizagem é uma abordagem importante para a evolução dos modelos convencionais de aprendizagem automática, onde as experiências de episódios anteriores também são utilizadas para melhorar o desempenho do modelo actual, e a sua contribuição é também evidente em condições de dados escassos.
- A utilização de algoritmos multiclassificadores permite uma resposta de classificação precisa, uma vez que é considerado um conjunto específico de classificadores individuais, o que reduz o impacto dos problemas de sobreajuste do modelo.
- O mecanismo proposto neste trabalho, para além de suportar a detecção de *Bots* numa fase posterior, pode ser utilizado para lidar com ameaças de negação de serviço, considerando que um determinado número de utilizadores pode ser bloqueado após a caracterização dos dados. Além disso, o processo de caracterização é essencial para lidar com *Bots* que alteram o seu comportamento.
- Os multiclassificadores híbridos apresentaram o melhor desempenho nos cenários de teste modelados. Para o processo de caracterização de dados de utilizadores de *Bot* e humanos, com base nas experiências e testes estatísticos realizados, o multiclassificador *Voting* é o mais adequado, com uma média de 99,6% de instâncias correctamente caracterizadas.

Trabalhos futuros

1. Trabalhar na melhoria da base de factos para que exemplos específicos estejam disponíveis, como o caso particular em que todos os utilizadores que acedem ao sistema são considerados *Bots*.
2. As medidas de descrição de dados utilizadas neste trabalho dependem dos rótulos das classes. Dadas as suas desvantagens, recomenda-se procurar e utilizar outras métricas de descrição de dados que não dependam de rótulos de classe e comparar as duas abordagens.

Referências

Acien, A., Morales, A., Fierrez, J., Vera-Rodriguez, R., & Delgado-Mohatar, O. (2021). BeCAPTCHA: Behavioral bot detection using touchscreen and mobile sensors benchmarked on HuMIdb. *Engineering Applications of Artificial Intelligence*, 98, 104058. https://doi.org/10.1016/j.engappai.2020.104058

Ahn, L. Von, Maurer, B., McMillen, C., Abraham, D., & Blum, M. (2008). reCAPTCHA: Human-Based Character Recognition via Web Security Measures. *Science*, 321(5895), 1465–1468. https://doi.org/10.1126/science.1160379

Albanese, M., Jajodia, S., & Venkatesan, S. (2018). Defending from Stealthy Botnets Using Moving Target Defenses. *IEEE Security Privacy*, 16(1), 92–97. https://doi.org/10.1109/MSP.2018.1331034

Alkadi, O., Moustafa, N., Turnbull, B., & Choo, K.-K. R. (2021). A Deep Blockchain Framework-Enabled Collaborative Intrusion Detection for Protecting IoT and Cloud Networks. *IEEE Internet of Things Journal*, 8(12), 9463–9472. https://doi.org/10.1109/JIOT.2020.2996590

Cepero-Pérez, N., Denis-Miranda, L. A., Hernández-Palacio, R., Moreno-Espino, M., & García-Borroto, M. (2018). Proactive Forest for Supervised Classification. In Y. Hernández Heredia, V. Milián Núñez, & J. Ruiz Shulcloper (Eds.), *Progress in Artificial Intelligence and Pattern Recognition* (pp. 255–262). Springer International Publishing. https://doi.org/10.1007/978-3-030-01132-1_29

Chen, H., He, H., & Starr, A. (2020). An Overview of Web Robots Detection Techniques. *IEEE Xplore*.

Chissingui, H. J., Pando, H. D., Espino, M. M., & Peréz, N. C. (2022). Bot detection algorithms: A systematic literature review. *Revista Cubana de Ciencias Informáticas*, 16(4), 1–26. https://rcci.uci.cu/index.php/RCCI/article/view/2548

Chissingui, H. J., Perez, N. C., Pando, H. D., & Espino, M. M. (2023). Multiclasificador homogeneo para detección de bots en el comercio electrónico. *Revista Cubana de Transformación Digital*, 4(1) e200. https://rctd.uic.cu/rctd/article/view/200/

Cresci, S., Pietro, R. Di, Petrocchi, M., Spognardi, A., & Tesconi, M. (2018). Social Fingerprinting: Detection of Spambot Groups Through DNA-Inspired Behavioral Modeling. *IEEE Transactions on Dependable and Secure Computing*, 15(4), 561–576. https://doi.org/10.1109/TDSC.2017.2681672

Duin, R. P. W. (2002). The combining classifier: to train or not to train? *2002 International Conference on Pattern Recognition*, 2, 765–770 vol.2. https://doi.org/10.1109/ICPR.2002.1048415

Garcia, S., Grill, M., Stiborek, J., & Zunimo, A. (2014). An empirical comparison of botnet detection methods. *Computers and Security Journal*, *Elsevier*, 45, 100–123. http://dx.doi.org/10.1016/j.cose.2014.05.011

Gezer, A., Warner, G., Wilson, C., & Shrestha, P. (2019). A flow-based approach for Trickbot banking trojan detection. *Computers & Security*, 84, 179–192. https://doi.org/10.1016/j.cose.2019.03.013

Han, J., Kamber, M., & Pei, J. (2012). *Data mining concepts and techniques, third edition*. Morgan Kaufmann Publishers. https://booksite.elsevier.com/9780123814791/

Hayawi, K., Saha, S., Masud, M. M., Mathew, S. S., & Kaosar, M. (2023). Social media bot detection with deep learning methods: a systematic review. *Neural Computing and Applications*, 35(12), 8903–8918. https://doi.org/10.1007/s00521-023-08352-z

Hitaj, D., Hitaj, B., Jajodia, S., & Mancini, L. V. (2020). Capture the Bot: Using Adversarial Examples to Improve CAPTCHA Robustness to Bot Attacks. *IEEE Intelligent Systems*. https://doi.org/10.1109/MIS.2020.3036156

Imperva. (2022). *2022 Imperva Bad Bot Report - Evasive Bots Drive Online Fraud*. Disponível em : https://www.imperva.com/resources/resource-library/reports/bad-bot-report/

Imperva. (2025). *2025 Bad Bot Report. The Rapid Rise of Bots and the Unseen Risk for Business*. https://www.imperva.com/resources/resource-library/reports/2025-bad-bot-report/

Karataş, A., & Şahin, S. (2017). *A Review on Social Bot Detection Techniques and Research Directions*.

Komorniczak, J., & Ksieniewicz, P. (2022, July 14). *problexity -- an open-source Python library for binary classification problem complexity assessment*. https://doi.org/10.48550/arXiv.2207.06709

Latah, M. (2020). Detection of malicious social bots: A survey and a refined taxonomy. *Expert Systems with Applications*, 151. https://doi.org/10.1016/j.eswa.2020.113383

Lorena, A. C., Garcia, L. P. F., Lehmann, J., Souto, M. C. P., & Ho, T. K. (2019). How Complex Is Your Classification Problem? A Survey on Measuring Classification Complexity. *ACM Comput. Surv.*, 52(5) 1 - 34. https://doi.org/10.1145/3347711

Maeda, S., Kanai, A., Tanimoto, S., Hatashima, T., & Ohkubo, K. (2019). A Botnet Detection Method on SDN using Deep Learning. *2019 IEEE International Conference on Consumer Electronics (ICCE)*, 1–6. https://doi.org/10.1109/ICCE.2019.8662080

Orabi, M., Mouheb, D., Al Aghbari, Z., & Kamel, I. (2020). Detection of Bots in Social Media: A Systematic Review. *Information Processing & Management*, 57(4), 102250. https://doi.org/10.1016/j.ipm.2020.102250

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in {P}ython. *Journal of Machine Learning Research*, 12, 2825–2830. https://jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf

Rahman, R. U., & Tomar, D. S. (2020). A new web forensic framework for bot crime investigation. *Forensic Science International: Digital Investigation*, 33, 300943. https://doi.org/10.1016/j.fsidi.2020.300943

Rheault, L., & Musulan, A. (2021). Efficient detection of online communities and social bot activity during electoral campaigns. *Journal of Information Technology and Politics*, 18(3), 324–337. https://doi.org/10.1080/19331681.2021.1879705

Rovetta, S., Suchacka, G., & Masulli, F. (2020). Bot recognition in a Web store: An approach based on unsupervised learning. *Journal of Network and Computer Applications*, 157, 102577. https://doi.org/10.1016/j.jnca.2020.102577

Stassopoulou, A., & Dikaiakos, M. D. (2009). Web robot detection: A probabilistic reasoning approach. *Computer Networks*, 53(3), 265–278. https://doi.org/10.1016/j.comnet.2008.09.021

Suchacka, G., Cabri, A., Rovetta, S., & Masulli, F. (2021). Efficient on-the-fly Web bot detection. Knowledge-Based Systems, 223, 107074. https://doi.org/10.1016/j.knosys.2021.107074

Suchacka, G., & Iwanski, J. (2020). Identifying legitimate Web users and bots with different traffic profiles — an Information Bottleneck approach. *Knowledge-Based Systems*, 197, 105875. https://doi.org/10.1016/j.knosys.2020.105875

Turing, A. M. (1950). Computing Machinery and Intelligence. *Oxford University Press on Behalf of the Mind Association*, 59(236), 433–460. http://www.jstor.org/stable/2251299

Varol, O., Ferrara, E., Davis, C. A., Menczer, F., & Flammini, A. (2017). Online Human-Bot Interactions : Detection , Estimation , and Characterization. *Proceedings of the Eleventh International AAAI Conference on Web and Social Media* (ICWSM 2017), Icwsm, 280–289. Disponível em: https://doi.org/10.1609/icwsm.v11i1.14871

Venkatesan, S., Albanese, M., Cybenko, G., & Jajodia, S. (2016). A Moving Target Defense Approach to Disrupting Stealthy Botnets. *Proceedings of the 2016 ACM Workshop on Moving Target Defense*, 37–46. Disponível em: https://doi.org/10.1145/2995272.2995280

Zha, Z., Wang, A., Guo, Y., Montgomery, D., & Chen, S. (2019). BotSifter: An SDN-based Online Bot Detection Framework in Data Centers. *2019 IEEE Conference on Communications and Network Security (CNS)*, 142–150. Disponível em: https://doi.org/10.1109/CNS.2019.8802854