IWSHAP: Uma Ferramenta para Seleção Incremental de Características utilizando IWSS e SHAP

Felipe H. Scherer[1], Felipe N. Dresch[1], Silvio E. Quincozes[1],[2]
Diego Kreutz[1], Vagner E. Quincozes[3]

[1]LEA, PPGES, Universidade Federal do Pampa (UNIPAMPA) – Alegrete, Brasil.

[2]PPGCO, Universidade Federal de Uberlândia (UFU) – Uberlândia, Brasil

[3]Instituto de Computação, Universidade Federal Fluminense (UFF) – Niterói, Brasil

{felipedresch,felipescherer}@aluno.unipampa.edu.br

{silvioquincozes, diegokreutz}@unipampa.edu.br

vequincozes@midiacom.uff.br

Resumo. *Este trabalho apresenta a ferramenta IWSHAP, que combina o algoritmo Iterative Wrapper Subset Selection (IWSS) com valores SHAP (SHapley Additive exPlanations) para encontrar o melhor conjunto de características e maximizar o desempenho de algoritmos de aprendizado de máquina. Os resultados em um cenário de detecção de intrusões em redes veiculares indicam que a ferramenta é eficaz na redução de datasets por meio da seleção de características, alcançando taxas elevadas de redução (e.g., 90%) e mantendo altas métricas de desempenho do modelo (acima de 90%).*

1. Introdução

Diversas técnicas para a seleção de características têm sido propostas na literatura para endereçar desafios de escalabilidade e qualidade de Sistemas de Detecção de Intrusões (*Intrusion Detection Systems* — IDSs), incluindo métodos de filtro, *wrapping* e *embedded*. Métodos de filtro são rápidos e baseiam-se na ordenação das características por relevância, mas frequentemente apresentam limitações na captura de interações complexas entre características. Os métodos *wrapping*, por outro lado, oferecem maior precisão ao considerar a interação entre características durante a seleção, mas apresentam um custo computacional elevado, o que pode inviabilizar seu uso em sistemas que precisam processar dados em tempo real. Métodos *embedded* integram a seleção de características diretamente no processo de aprendizagem, sendo específicos para cada algoritmo [Chandrashekar and Sahin 2014].

Recentemente, ferramentas de Inteligência Artificial Explicável, do inglês *eXplainable Artificial Intelligence* (XAI), como a técnica SHAP (*SHapley Additive exPlanations*), têm sido empregadas como recurso complementar para entender melhor a contribuição de cada característica [Quincozes et al. 2024]. A SHAP utiliza valores baseados na teoria dos jogos para fornecer explicações sobre a influência de cada característica no modelo, oferecendo novas perspectivas para a seleção de características. Especificamente, com a SHAP é possível criar um *ranking* das características mais influentes, permitindo uma seleção mais informada e precisa das características mais relevantes. Na literatura existente, trabalhos têm aplicado valores SHAP para a seleção de características,

inspirados principalmente por abordagens de filtragem utilizando o *ranking* dos valores gerados [Nazat et al. 2024, E. L. Asry et al. 2024, Setitra et al. 2023]. Essas abordagens, no entanto, enfrentam desafios semelhantes aos métodos tradicionais de filtro, como a incapacidade de considerar interações entre características.

Neste artigo é apresentada uma nova ferramenta de seleção de características, denominada IWSHAP [Scherer et al. 2024], que combina o algoritmo *Incremental Wrapper Subset Selection* (IWSS) [Bermejo et al. 2009] com valores SHAP. O objetivo é alcançar um equilíbrio entre o desempenho do modelo de aprendizado de máquina e o tempo de otimização do conjunto de características analisado. Na versão atual da ferramenta é possível gerar gráficos das métricas de desempenho do modelo *F1-Score*, *Recall* e *Precision*. A ferramenta também gera um gráfico de resumo, conhecido como *summary plot*, que destaca a relevância das características na detecção de ataques. A partir das relevâncias das características busca-se encontrar o melhor conjunto de características. Ainda, o IWSHAP gera registro geral de toda a execução, no qual mantém as informações de cada rodada de execução, como métricas de cada conjunto analisado e as características utilizadas na respectiva rodada. Por fim, o IWSHAP tem a capacidade de gerar *datasets* reduzidos em mais de 90%, mantendo a qualidade preditiva equivalente ao *baseline*.

Os resultados demonstram que o IWSHAP pode contribuir significativamente para a detecção de intrusões em redes CAN (*Controller Area Network*), oferecendo uma solução mais balanceada e eficiente para a seleção de características. Eles também indicam que o método se destaca tanto em capacidade de redução (*i.e.*, menor número de características selecionadas) quanto em termos de tempo de execução.

2. Ferramenta IWSHAP

Nesta seção é detalhado o fluxo de execução (Seção 2.1) e a implementação (Seção 2.2) do algoritmo central da ferramenta.

2.1. Fluxo de execução

Conforme observado na Figura 1, o fluxo de execução da ferramenta inicia com a **entrada de dados**. Primeiro, a ferramenta recebe os *datasets* benigno e maligno (*i.e.*, sem ataques e com ataques). Na sequência, os *datasets* são concatenados e os valores categóricos são convertidos para numéricos.

No estágio de **processamento inicial** a ferramenta treina o modelo inicial com todo o conjunto de dados, realiza o cálculo dos valores SHAP e apresenta o *ranking* de relevância de cada característica, que leva à criação de um novo *dataframe* ordenado pelo *ranking*; Já o **processo IWSHAP**, demarcado pela linha pontilhada, é responsável pela lógica central da ferramenta. Primeiramente, inicializa-se um novo *dataframe* onde serão testadas as características a partir de iterações com base no conjunto das características mais relevantes. Na sequência, o conjunto que obtiver o melhor *F1-Score* é mantido. Esse ciclo é repetido várias vezes, até que não hajam mais características a serem avaliadas. Por fim, o estágio de **documentar e analisar resultados** registra os resultados das ações realizadas durante o **processo IWSHAP**. Neste estágio inclui-se a geração de arquivos de registro geral, gráficos das métricas de desempenho do modelo, um gráfico *summary plot* e a geração de um *dataset* reduzido com as melhores características encontradas.

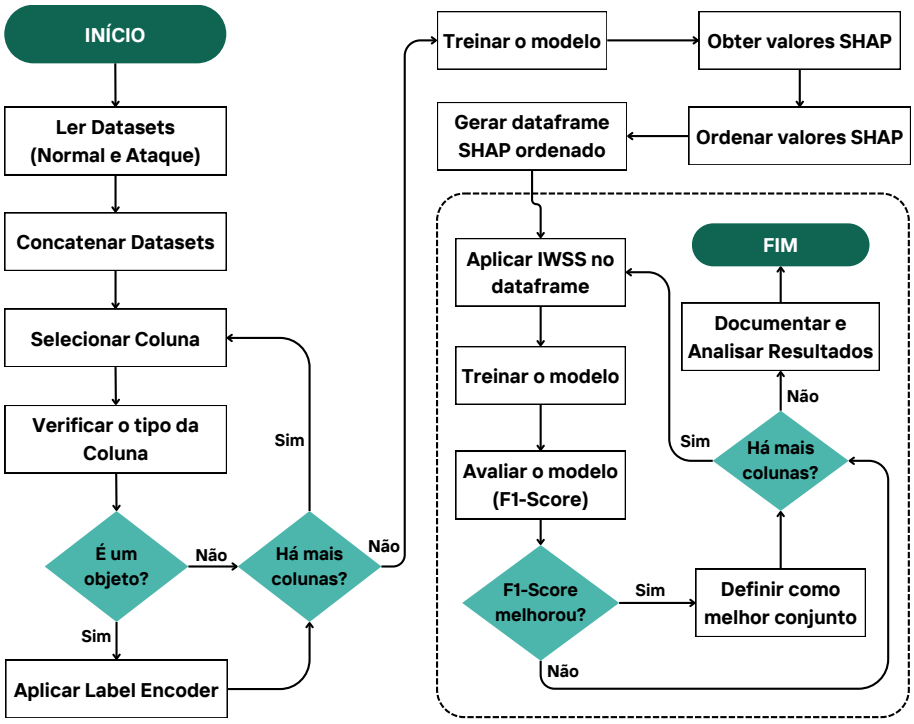

Figura 1. Fluxo de execução da ferramenta da ferramenta.

2.2. Implementação da Ferramenta

Com relação à implementação e execução da ferramenta, foi utilizada a linguagem de programação Python (versão 3.12.4) em conjunto com as bibliotecas Pandas (versão 2.2.2), NumPy (versão 1.26.4), SHAP (versão 0.45.1), XGBoost (versão 2.0.3), SciKit-learn (versão 1.5.0), MatPlotLib (versão 3.9.0) e PyArrow (versão 16.1.0).

A Tabela 1 resume os parâmetros de entrada da ferramenta, incluindo a *flag* abreviada, uma breve descrição e o tipo (obrigatório ou não). O parâmetro `--safe-path` (`-s`) indica o caminho para o *dataset* benigno, enquanto o parâmetro `--attack-path` (`-a`) indica o caminho para o *dataset* maligno. Os formatos de *datasets* aceitos são arquivos estruturados do tipo `CSV` ou `Parquet`. O parâmetro `--explainable` (`-x`) é utilizado para definir a criação do gráfico SHAP *summary plot*. Ademais, o parâmetro opcional `--newdata-reduced` (`-n`) pode ser utilizado para gerar *datasets* reduzidos baseados no melhor conjunto de características encontrado.

3. Matérias e métodos

Para facilitar a instalação, utilização e execução da IWSHAP, foi criado e disponibilizado um *Dockerfile* para utilização do sistema de conteinerização Docker em ambientes Linux, MacOS X, Windows, entre outros. O *Dockerfile* discrimina os requisitos de sistema operacional e dependências necessárias para instanciação e execução da ferramenta.

O ambiente Docker também contribui para a reprodução. Para efetivamente garantir a reprodução, disponibilizamos todas as entradas e parâmetros utilizados na execução dos experimentos com a IWSHAP, apresentados na Seção 4. Seguindo a documentação da ferramenta disponível no GitHub, o usuário será capaz de reproduzir os mesmos resultados apresentados neste trabalho.

Tabela 1. Parâmetros de entrada da ferramenta.

Parâmetro	Flag	Descrição	Tipo
–safe-path	-s	Indica o caminho de entrada do *dataset* benigno	Obrigatória
–attack-path	-a	Indica o caminho de entrada do *dataset* maligno	Obrigatória
–log-path	-l	Indica o caminho do diretório onde serão mantidos os registros gerais	Opcional
–graphics-path	-g	Indica o caminho do diretório onde os gráficos serão armazenados	Opcional
–newdata-reduced	-n	Define a geração de um novo *dataset* reduzido com as melhores características	Opcional
–explainable	-x	Define a geração de um gráfico *summary plot* com as características do melhor conjunto	Opcional

Para a realização dos experimentos, foi utilizado um servidor com processador AMD Ryzen 7 5800x de 8 *cores* e 64 GB de memória RAM. O sistema operacional do servidor é o Ubuntu Server, versão 22.04.

Para fins de avaliação da ferramenta IWSHAP, foram utilizados dados do *dataset* X-CANIDS [Jeong et al. 2024]. Tal *dataset* contém amostras de mensagens CAN e abrange diversos tipos de ataques, incluindo suspensão, fabricação, *Masquerade*, repetição e *Fuzzing*. No total, cada amostra do *dataset* original contém 392.387 instâncias.

Os testes foram conduzidos utilizando duas amostras distintas do *dataset*: uma contendo apenas dados livres de ataques e outra contendo dados com ataques. Isso resultou em um total de 784.774 instâncias analisadas. Ademais, para a verificação da eficácia da ferramenta, foram utilizados especificamente os ataques de suspensão e fabricação.

Como parte do processo do IWSHAP, os dados são divididos em conjuntos de teste e treinamento e, por padrão, a ferramenta utiliza de 20% dos dados para teste e 80% para treinamento. Nos testes conduzidos a divisão resultou em 627.819 mil instâncias para treinamento e 156.955 mil de teste.

4. Resultados

Nesta seção são apresentados e discutidos a geração de *datasets* reduzidos, os registros da execução e os gráficos de explicabilidade da ferramenta.

4.1. Registro de Execução (Logs)

Para uma análise mais completa do processo de seleção conduzido pelo IWSHAP, é elaborado um registro abrangente da execução, no qual são registradas as métricas de desempenho do modelo referentes a *baseline* e as características selecionadas em cada iteração, juntamente com as métricas de desempenho do modelo correspondente. Além disso, o registro inclui o resultado final do conjunto ótimo de características e suas respectivas métricas.

Esse registro global permite a análise das características que possam ter impactado negativamente o modelo, bem como proporcionar uma compreensão aprofundada sobre aquelas que mais contribuíram para o desempenho. A Figura 2 apresenta um recorte do registro geral, o qual informa o número da rodada e seu tempo de execução, bem como

as características atuais e as métricas de desempenho do modelo. É importante destacar também que, no início do registro geral, são apresentadas as métricas de desempenho do modelo da *baseline*, para fins de comparação. Para análises mais aprofundadas, gráficos podem ser gerados a partir do registro geral para visualizar as variações ao longo das iterações, facilitando a compreensão da relevância de cada característica.

```
Rodada: 687
Features atuais:
 ['2B0_MsgCount', '2B0_SAS_Angle', '2B0_CheckSum', '5B0_CF_Clu_Odometer', '260_AliveCounter', '220_ESP12
_Checksum', '080_CF_Ems_Alive', '164_CF_Esc_AliveCnt', '111_CF_Tcu_Alive1', '164_CF_Esc_Chksum', '081
_CR_Ems_IndAirTemp', '220_CYL_PRES', '556_PID_05h', '386_WHL_SPD_FR', '4F1_CF_Clu_DetentOut', '081
_CF_Ems_ChkSum2', '5FA_CR_Wcs_ClassStat']
F1 Score: 0.9186191167323243, Recall: 0.9230208333333333, Precision: 0.914259182831201, Tempo:
0.9878129959106445 segundos
Melhores features finais:
 ['2B0_MsgCount', '2B0_SAS_Angle', '2B0_CheckSum', '5B0_CF_Clu_Odometer', '260_AliveCounter', '220_ESP12
_Checksum', '080_CF_Ems_Alive', '164_CF_Esc_AliveCnt', '111_CF_Tcu_Alive1', '164_CF_Esc_Chksum', '081
_CR_Ems_IndAirTemp', '220_CYL_PRES', '556_PID_05h', '386_WHL_SPD_FR', '4F1_CF_Clu_DetentOut', '081
_CF_Ems_ChkSum2']
Melhor F1 Score: 0.9186191167323243
```

Figura 2. Exemplo de recorte do registro geral da execução (Log).

4.2. Gráficos das Métricas

Como artefato de saída, o IWSHAP disponibiliza um gráfico contendo as métricas *F1-Score*, *Recall* e *Precision* obtidas para o melhor conjunto de características, conforme apresentado na Figura 3. O gráfico permite uma análise comparativa das métricas de desempenho do modelo em relação à sua *baseline*.

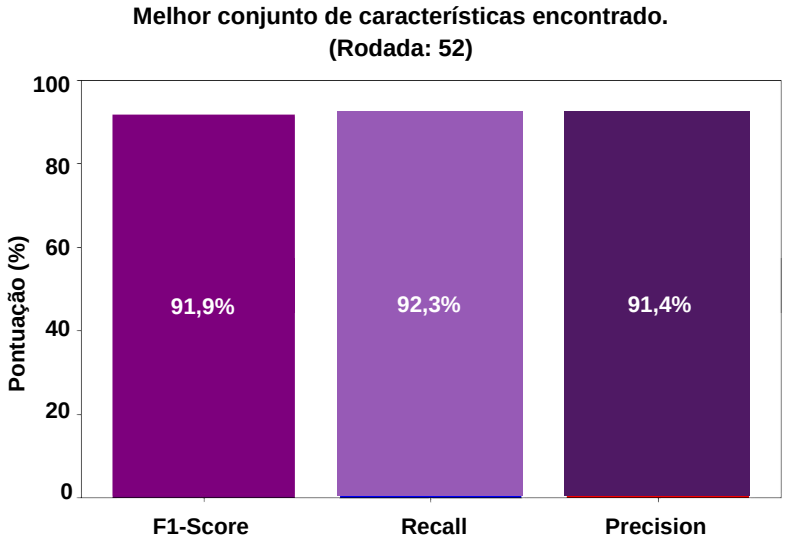

Figura 3. Exemplo de gráfico de métricas para o *dataset* de ataque de suspensão.

Inicialmente, na sua *baseline*, o modelo alcançou um *F1-Score* de 90,17% utilizando 688 características. Ainda, conforme analisado na Figura 3, com a aplicação da ferramenta IWSHAP o classificador teve um aumento nas taxas de *F1-Score*, *Recall* e *Precision*, obtendo as pontuações de 91,9%, 92,3% e 91,4%, respectivamente. Portanto, o método se mostra superior ao *baseline*, o qual obteve as pontuações de 90,17%, 90,44% e 89,90% em termos de *F1-Score*, *Recall* e *Precision*, respectivamente.

Essa análise comparativa é essencial para validar a eficácia da ferramenta IWSHAP na redução dimensional sem perda significativa de desempenho. A capacidade de manter ou até melhorar as métricas de desempenho com um conjunto de características significativamente menor demonstra a eficiência da ferramenta proposta na seleção das características mais relevantes.

Ademais, a interpretação dessas métricas permite uma compreensão mais profunda do comportamento do modelo em diferentes cenários. A análise detalhada do *Recall* e da *Precision* proporciona uma compreensão valiosa sobre a capacidade do modelo de detectar corretamente os ataques e de minimizar os falsos alarmes. Com essas informações, é possível ajustar e otimizar o modelo para melhor adequá-lo às necessidades específicas do sistema de segurança em questão.

4.3. Gráfico de Explicabilidade

A partir do melhor conjunto de características selecionado, a IWSHAP também permite a geração de um gráfico do tipo *summary plot*, que apresenta a relevância de cada característica em conjunto com o seu respectivo valor SHAP. Com base nesse gráfico, é possível desenvolver soluções específicas, uma vez que ele permite identificar tanto o local do ataque (*i.e.*, onde) quanto a natureza do ataque (*e.g.*, negação e sobrecarga).

A Figura 4 ilustra a explicabilidade[1] gerada a partir do ataque de suspensão. Esse gráfico revela que a característica mais relevante na identificação de ataques de suspensão é a `2B0_MsgCount`. Essa informação é crucial, pois permite focar em aspectos específicos do sistema que são mais vulneráveis a esse tipo de ataque e assim desenhar uma solução mais eficiente para lidar com ataques.

Por fim, é importante destacar que essa metodologia de análise e explicabilidade pode ser aplicada a diversos contextos e tipos de ataques, tornando a ferramenta versátil e aplicável a diferentes domínios. A capacidade de identificar e entender as características mais relevantes proporciona uma boa base para a implementação de soluções de segurança mais direcionadas.

4.4. Discussão

A IWSHAP demonstrou a capacidade de reduzir os *datasets* sem comprometer as taxas das métricas de desempenho. Em avaliações realizadas, a ferramenta foi capaz de diminuir em até 99,17% a quantidade de características no conjunto de dados do ataque de suspensão e em 95,93% no ataque de fabricação. Ademais, no que tange ao tempo de execução, o IWSHAP destacou-se por necessitar apenas de 1,01 segundos de treinamento para alcançar seu melhor *F1-Score*, enquanto o *baseline* demandou 59,48 segundos para atingir o mesmo objetivo.

É importante ressaltar que, apesar da significativa redução no número de características e tempo de execução, a qualidade preditiva dos modelos de detecção de intrusões manteve-se robusta, sendo equivalente e até superior a *baseline*. Esta constatação substancia a eficácia do IWSHAP em preservar a precisão do sistema enquanto otimiza a eficiência operacional através da redução de complexidade nos conjuntos de dados analisados.

[1] O texto contido no gráfico aparece em inglês, conforme o *output* original da ferramenta SHAP.

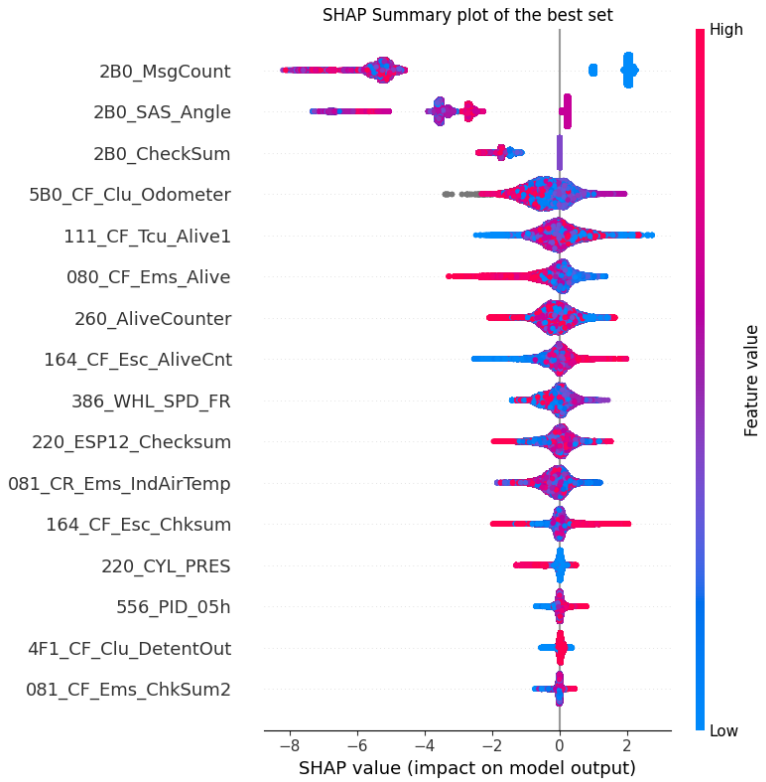

Figura 4. Exemplo de gráfico de explicabilidade com base nos valores SHAP gerado a partir da execução com o *dataset* de ataque de suspensão.

Além disso, o IWSHAP apresenta uma alta taxa de *recall*, essencial para a detecção eficaz de intrusões em sistemas ciber-físicos. Uma alta taxa de *recall* indica que o modelo é eficaz em identificar a maioria dos eventos ou instâncias relevantes, minimizando a ocorrência de falsos negativos. Isso significa que o sistema tem uma capacidade elevada de detectar verdadeiros positivos, assegurando que a maioria dos ataques ou intrusões sejam identificados corretamente mesmo com as características reduzidas.

Os resultados obtidos corroboram a viabilidade e a relevância da IWSHAP como uma ferramenta eficiente para a seleção de características em problemas de detecção de intrusões. Ao simplificar e reduzir o número de características sem comprometer a qualidade dos resultados, a IWSHAP não apenas oferece uma solução mais balanceada e eficiente, mas também promove avanços significativos na segurança e na eficiência operacional de sistemas de rede.

5. Considerações Finais

Este trabalho introduziu a ferramenta IWSHAP para a seleção de características e entendimento de ataques a partir dos artefatos gerados (*i.e.*, gráfico de métricas de desempenho do modelo e de explicabilidade). Os resultados experimentais indicam que a ferramenta é capaz de atingir altas taxas de redução sem comprometer as taxas das métricas de desempenho, o que é importante para ambientes críticos e/ou de tempo real. Ademais, a partir dos resultados da IWSHAP, é importante destacar que é possível modelar soluções de detecção de ataques mais eficientes, contribuindo para uma maior robustez e resiliência das defesas contra ameaças cibernéticas.

O código fonte, documentação, *datasets* utilizados, artefatos de exemplo e instruções de instalação estão disponíveis no repositório da IWSHAP[2]. A demonstração da ferramenta será realizada através de um ambiente Docker em dispositivo próprio dos autores. O funcionamento da ferramenta será demostrado através dos seguintes passos: (a) apresentação dos parâmetros de execução da ferramenta; (b) apresentação das funcionalidades; (c) apresentação da execução e resultados gerados.

Como trabalhos futuros podem ser destacados: (a) inclusão de novos gráficos de explicabilidade e desempenho do modelo; (b) inclusão de diferentes modelos de classificação; (c) aprimoramento da seleção de características a partir da inserção e/ou mesclagem de abordagens; (d) incorporação de maior configurações por parte do usuário.

Agradecimentos. Esta pesquisa foi parcialmente financiada, com apoio da CAPES – Código de Financiamento 001 e FAPERGS, através dos editais 08/2023 e 09/2023.

Referências

Bermejo, P., Gámez, J. A., and Puerta, J. M. (2009). Incremental wrapper-based subset selection with replacement: An advantageous alternative to sequential forward selection. In *2009 IEEE Symposium on Computational Intelligence and Data Mining*, pages 367–374. IEEE.

Chandrashekar, G. and Sahin, F. (2014). A survey on feature selection methods. *Computers & electrical engineering*, 40(1):16–28.

E. L. Asry, C., Benchaji, I., Douzi, S., and E. L. Ouahidi, B. (2024). A robust intrusion detection system based on a shallow learning model and feature extraction techniques. *PLOS ONE*, 19(1):1–31.

Jeong, S., Lee, S., Lee, H., and Kim, H. K. (2024). X-CANIDS: Signal-aware explainable intrusion detection system for controller area network-based in-vehicle network. *IEEE Transactions on Vehicular Technology*, 73(3):3230–3246.

Nazat, S., Li, L., and Abdallah, M. (2024). XAI-ADS: An explainable artificial intelligence framework for enhancing anomaly detection in autonomous driving systems. *IEEE Access*, 12:48583–48607.

Quincozes, V. E., Quincozes, S. E., Kazienko, J. F., Gama, S., Cheikhrouhou, O., and Koubaa, A. (2024). A survey on IoT application layer protocols, security challenges, and the role of explainable AI in IoT (XAIoT). *International Journal of Information Security*, 23(3):1975–2002.

Scherer, F. H., Dresch, F. N., Quincozes, S. E., Kreutz, D., and Quincozes, V. E. (2024). IWSHAP: Um método de seleção incremental de características para redes CAN baseado em Inteligência Artificial Explicável (XAI). In *Anais do XXIV Simpósio Brasileiro de Segurança da Informaçao e de Sistemas Computacionais*. SBC.

Setitra, M. A., Fan, M., and Bensalem, Z. E. A. (2023). An efficient approach to detect distributed denial of service attacks for software defined internet of things combining autoencoder and extreme gradient boosting with feature selection and hyperparameter tuning optimization. *Transactions on Emerging Telecommunications Technologies*, 34(9):e4827.

[2]Disponível em: `https://github.com/sf24-iwshap/sf24-iwshap`