

# Avaliação de Métodos de Seleção de Características de Amostras Android com a Ferramenta FS3E (v2)

Nicolas Neves<sup>2</sup>, Vanderson Rocha<sup>2</sup>, Diego Kreutz<sup>1</sup>  
Hendrio Bragança<sup>2</sup>, Eduardo Feitosa<sup>2</sup>

<sup>1</sup>Universidade Federal do Pampa (UNIPAMPA)  
diegokreutz@unipampa.edu.br

<sup>2</sup>Universidade Federal do Amazonas (UFAM)  
{nicolas.neves, hendrio.luis, efeitosa}@icom.ufam.br  
vanderson@ufam.edu.br

**Resumo.** Neste artigo apresentamos uma avaliação extensiva de oito métodos de seleção de características utilizando a nova versão da F3SE (v2). Para a avaliação dos métodos, utilizamos seis datasets distintos. Os resultados indicam que há uma variabilidade significativa entre os métodos e entre os datasets. Entretanto, podemos observar os métodos FSDroid e SemiDroid apresentam uma maior estabilidade e qualidade geral na seleção de características.

## 1. Introdução

A seleção de características, também conhecida como seleção de variáveis ou seleção de atributos, é uma técnica que permite selecionar as características que tornam um modelo mais simples, mais fácil de interpretar e menos propenso a *overfitting* [HEAVY.AI, 2023]. Os métodos mais comuns de seleção de características são: (1) **filtro**, que é baseado na seleção de características com base em estatísticas; (2) **invólucro**, que considera a seleção de um conjunto de características como um problema de busca; e (3) **embutido**, onde a classificação e a seleção de características são realizadas simultaneamente.

A FS3E [Costa et al., 2022] permite a incorporação e avaliação sistemática de métodos de seleção de características para o domínio de detecção de malwares Android [Soares et al., 2022]. A versão original contém os métodos de seleção de características JOWNDroid, LR, RFG, SigAPI e SigPID. Como podemos observar na Figura 1, a ferramenta é composta por três etapas: (a) **métodos** de seleção de características, (b) **avaliação** com modelos de aprendizado de máquina, e (c) **visualização** dos resultados.

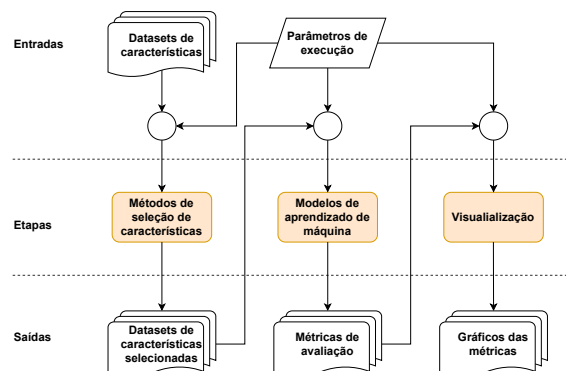


Figura 1. Etapas e fluxo de dados da FS3E [Costa et al., 2022]

Na primeira etapa, os **métodos** de seleção de características escolhidos pelo usuário são aplicados sobre cada *dataset* de **entrada**. A **saída** desta etapa serve de **entrada** para a **avaliação**, que utiliza modelos de aprendizado de máquina. As métricas de avaliação resultantes representam a **entrada** da etapa de **visualização**. É importante ressaltarmos que desconhecemos trabalho similar (vide [Costa et al., 2022]), ou seja, um arcabouço de software para implementação, disponibilização e avaliação sistemática de métodos e técnicas de seleção de características para o domínio de detecção de malwares Android. É importante destacarmos também que assumimos que os *datasets* de entrada da FS3E não contém amostras intencionalmente modificadas por um atacante.

Neste trabalho apresentamos a incorporação e avaliação de quatro novos métodos de seleção de características, o *Artificial Bee Colony* (ABC) [Mohammadi et al., 2020], FSDroid [Mahindru and Sangal, 2021a], SemiDroid [Mahindru and Sangal, 2021b] e *Multi-Tiered* (MT) [Bhat and Dutta, 2022]. Para a avaliação dos métodos utilizamos os *datasets* AndroCrawl, Adroit, Android Permissions, DefenseDroid, Drebin-215 e KronoDroid (Dispositivo Real). Detalhes adicionais sobre a implementação e o código dos métodos, bem como informações técnicas e identificação das fontes dos *datasets* originais e sub-sets reduzidos estão disponíveis no repositório do GitHub<sup>1</sup>.

As principais contribuições do trabalho são: (a) implementação, reprodução e disponibilização de quatro métodos adicionais de seleção de características, (b) uma avaliação extensiva de oito métodos utilizando seis *datasets* distintos, (c) identificação de aspectos conceituais e de implementação que levam alguns métodos a ter resultados superiores aos demais e (d) identificação de anomalias em *datasets* e métodos de seleção.

## 2. Resultados

Utilizamos como entrada os *datasets* detalhados na Tabela 1. Para cada *dataset*, apresentamos um resumo das características e amostras de cada *dataset*, incluindo o número e percentagem de amostras maliciosas e benignas e o total de amostras.

**Tabela 1. Datasets**

Dataset	Características	Amostras				
		Malwares		Benignos		Total
AndroCrawl	141	10170	10.5%	86562	89.5%	96732
Adroit	166	3418	29.8%	8058	70.2%	11476
Android Permissions	151	17787	66,21%	9077	33,79%	26864
DefenseDroid	2938	6000	50.1%	5975	49.9%	11975
Drebin-215	215	5560	37%	9476	63%	15036
KronoDroid	383	41382	53%	36755	47%	78137

Utilizamos a FS3E (v2) para executar os métodos JOWNDroid, LR, RFG, SigPID, ABC, FSDroid, SemiDroid e MT pada cada *dataset*. Cada método gera um novo sub-set que é utilizado como entrada do modelo *Random Forest*<sup>2</sup>. Como resultados temos as métricas de acurácia, precisão, *recall*, F1-Score, MCC e ROC\_AUC. Os testes foram realizados em uma máquina Dell G15, com 16 GB de memória, processador Intel I5-11, rodando Ubuntu 20.04 e Python 3.11.

<sup>1</sup><https://github.com/Malware-Hunter/WRSeg23-FS3E-v2>

<sup>2</sup>Hiperparâmetros utilizados: *n\_estimators=100, max\_depth=2, random\_state=0*

As Tabelas 2, 3, 4, 5, 6, 7 apresentam os resultados para os seis *datasets* escolhidos. A primeira coluna indica o *dataset*, sendo que há um novo *dataset* para cada método (e.g., ABC\_Adroit, i.e., *dataset* gerado pelo método ABC sobre o conjunto original do Adroit). O *dataset* original aparece na primeira linha de cada tabela. As colunas 2 a 7 apresentam as métricas do modelo *Random Forest* executado sobre cada conjunto de dados. Finalmente, as duas últimas colunas apresentam o número de características do *dataset* e a porcentagem de redução em relação ao conjunto de dados original (linha 1).

**Tabela 2. Métricas e redução para o *dataset* Adroit**

Dataset	Accuracy	Precision	Recall	F1_Score	MCC	ROC_AUC	# caract.	Redução (%)
Adroit	81.90	66.31	79.72	72.40	0.59	81.27	167	0.00
ABC_Adroit	81.93	66.36	79.75	72.44	0.60	81.30	166	0.60
FSDroid_Adroit	89.13	91.29	70.22	79.38	0.73	83.69	29	82.63
JOWNDroid_Adroit	70.07	44.37	1.96	3.75	0.04	50.46	47	71.86
LR_Adroit	88.80	87.33	73.00	79.52	0.72	84.25	101	39.52
RFG_Adroit	86.22	76.83	76.95	76.89	0.67	83.55	122	26.95
SemiDroid_Adroit	88.70	89.91	69.89	78.65	0.72	83.28	18	89.22
MT_Adroit	87.26	83.75	71.01	76.85	0.69	82.58	5	97.01
SigPID_Adroit	89.18	94.44	67.64	78.83	0.74	82.98	5	97.01

No caso do *dataset* Adroit (Tabela 2), podemos observar que os métodos de seleção SigPID e MT apresentaram uma alta taxa de redução e um bom desempenho, isto é, uma ROC\_AUC muito próxima aos demais métodos. No caso do SigPID, aspectos do *dataset* como esparsidade, frequência das características e pares com alta concordância contribuem para o bom resultado.

É interessante observarmos que os métodos LR e SemiDroid apresentaram reduções de 101 e 18 características. Entretanto, as métricas resultantes foram similares aos métodos SigPID e MT, que atingiram taxas de redução significativamente maiores. Estes resultados indicam que o SigPID e MT são os mais indicados para *datasets* similares ao Adroit quando o objetivo é atingir uma alta taxa de redução. Já em termos de métricas de classificação, métodos como o FSDroid atingem taxas melhores, ao custo de uma redução ligeiramente menor (e.g., redução de 82% contra 97% do SigPID e MT).

**Tabela 3. Métricas e redução para o *dataset* Androcrawl**

Dataset	Accuracy	Precision	Recall	F1_Score	MCC	ROC_AUC	# caract.	Redução (%)
Androcrawl	98.37	94.57	89.60	92.02	0.91	94.50	142	0.00
ABC_Androcrawl	98.32	94.53	89.16	91.77	0.91	94.28	140	1.41
FSDroid_Androcrawl	98.36	94.33	89.76	91.99	0.91	94.56	29	79.58
JOWNDroid_Androcrawl	89.29	25.93	1.03	1.99	0.03	50.34	31	78.17
LR_Androcrawl	98.00	93.47	87.10	90.17	0.89	93.19	8	94.37
RFG_Androcrawl	91.46	78.77	25.68	38.74	0.42	62.44	22	84.51
SemiDroid_Androcrawl	98.14	95.38	86.51	90.73	0.90	93.01	16	88.73
MT_Androcrawl	89.29	41.56	4.72	8.48	0.11	51.97	5	96.48
SigPID_Androcrawl	98.06	94.14	87.01	90.43	0.89	93.19	6	95.77

No caso do *dataset* Androcrawl (Tabela 3), podemos observar uma tendência similar ao Adroit, exceto para os métodos RFG e MT, que obtiveram métricas significativamente piores. Diferentemente do RFG e MT, métodos como o SigPID selecionam características criadas por especialistas, como "*Contains URL known to be suspicious*" e "*Airpush Included*". Já o RFG e MT selecionam apenas características pouco expressivas para reconhecer malwares, como "*GET\_PACKAGE\_SIZE*", "*EXPAND\_STATUS\_BAR*",

"*Landroid/content/Context;-¿startService*", "*FLASHLIGHT*", e "*FORCE\_BACK*". Interessantemente, algumas dessas características estão indicadas como "*Harmless*" no *dataset*, ou seja, tem baixa relação com aplicativos maliciosos.

Detectamos que o MT possui um *threshold* de 0,8 na frequência de cada característica na primeira etapa do método. Isto nos leva a hipótese de o MT eliminar prematuramente características que são importantes para a detecção de malwares, como aquelas criadas por especialistas. Como resultado, o MT acaba selecionando características menos relevantes em *datasets* como o Androcrawl, o que explica os resultados piores.

**Tabela 4. Métricas e redução para o *dataset* Android\_Permissions**

Dataset	Accuracy	Precision	Recall	F1_Score	MCC	ROC_AUC	# caract.	Redução (%)
Android_Permissions	66.72	69.17	89.76	78.13	0.15	55.67	152	0.00
ABC_Android_Permissions	66.55	68.92	90.14	78.11	0.15	55.24	150	1.32
FSDroid_Android_Permissions	67.13	67.96	95.26	79.33	0.13	53.63	44	71.05
JOWNDroid_Android_Permissions	66.02	66.85	96.56	79.01	0.06	51.37	69	54.61
LR_Android_Permissions	66.77	66.88	98.70	79.73	0.09	51.45	69	54.61
RFG_Android_Permissions	66.09	66.49	98.37	79.35	0.04	50.61	22	85.53
SemiDroid_Android_Permissions	66.48	67.43	95.48	79.04	0.10	52.56	17	88.82
MT_Android_Permissions	66.21	66.21	100.00	79.67	0.00	50.00	4	97.37
SigPID_Android_Permissions	66.61	66.72	98.93	79.69	0.08	51.11	43	71.71

No caso do *dataset* Android Permissions (Tabela 4), podemos observar um desempenho baixo quando comparado com outros *datasets*. A explicação para esse comportamento pode estar fortemente relacionada a dois aspectos do *dataset*. Primeiro, o Android Permissions possui um desbalanceamento inverso de amostras, isto é, mais amostras malignas do que benignas. Segundo, possui diversas características compostas, definidas por especialistas (e.g. "*add or modify calendar events and send email to guests*", "*modify/delete usb storage contents modify/delete sd card contents*"), o que acaba dificultando a seleção de características. Finalmente, é importante observarmos que o *dataset* original (linha 1 da tabela) também apresenta um desempenho ruim, ou seja, isto pode ser um indicativo de má qualidade geral das características definidas no *dataset*. É possível observarmos também que apenas características do tipo permissões não leva a taxas muito boas (i.e., acima de 90%) para classificação de *malwares* Android. Por exemplo, no caso do *dataset* Adroit, cujos resultados podem ser vistos na Tabela 2, o melhor método de seleção levou a taxas de classificação abaixo de 85%.

**Tabela 5. Métricas e redução para o *dataset* Defensedroid\_prs**

Dataset	Accuracy	Precision	Recall	F1_Score	MCC	ROC_AUC	# caract.	Redução (%)
Defensedroid_prs	92.38	93.62	91.00	92.29	0.85	92.39	2878	0.00
ABC_Defensedroid_prs	91.97	93.11	90.70	91.89	0.84	91.98	2789	3.09
FSDroid_Defensedroid_prs	88.56	91.52	85.05	88.17	0.77	88.57	32	98.89
JOWNDroid_Defensedroid_prs	73.54	84.79	57.50	68.53	0.50	73.57	1296	54.97
LR_Defensedroid_prs	88.68	87.27	90.62	88.91	0.77	88.67	2041	29.08
RFG_Defensedroid_prs	73.59	83.87	58.57	68.97	0.50	73.63	22	99.24
SemiDroid_Defensedroid_prs	92.10	94.69	89.23	91.88	0.84	92.11	289	89.96
MT_Defensedroid_prs	81.41	78.69	86.27	82.30	0.63	81.40	6	99.79
SigPID_Defesemsdroid_prs	89.84	93.08	86.12	89.46	0.80	89.84	70	97.57

No caso do *dataset* DefenseDroid (Tabela 5), podemos observar novamente um comportamento similar ao do Androcrawl. É interessante destacarmos que os métodos FSDroid e o SigPID, novamente, atingem taxas de redução acima de 97% e conseguem manter taxas de classificação muito próxima do *dataset* original (linha 1 da tabela).

**Tabela 6. Métricas e redução para o *dataset* Drebin-215**

Dataset	Accuracy	Precision	Recall	F1_Score	MCC	ROC_AUC	Total reduzido	Porcentagem
Drebin215	98.80	99.23	97.50	98.36	0.97	98.53	216	0.00
ABC_Drebin215	98.80	99.23	97.50	98.36	0.97	98.53	216	0.00
FSDroid_Drebin215	93.78	94.77	88.03	91.27	0.87	92.59	20	90.74
JOWNDroid_Drebin215	80.09	82.07	59.01	68.66	0.56	75.73	34	84.26
LR_Drebin215	93.73	95.07	87.56	91.16	0.86	92.45	32	85.19
RFG_Drebin215	96.91	96.84	94.74	95.78	0.93	96.46	42	80.56
SemiDroid_Drebin215	94.52	93.48	91.56	92.51	0.88	93.91	23	89.35
MT_Drebin215	93.29	91.51	90.23	90.86	0.86	92.66	21	90.28
SigPID_Drebin215	94.87	95.34	90.55	92.88	0.89	93.98	21	90.28

No caso do Drebin-215 (Tabela 6) os resultados mostram que todos os métodos conseguiram uma pontuação próxima ao *dataset* original, menos o JOWNDroid, que ficou 23% abaixo da taxa de ROC\_AUC. Há dois aspectos a considerar com relação ao JOWNDroid. Primeiro, ele utiliza modelos pouco apropriados para esses conjuntos de características, como a regressão linear. Segundo, seus modelos de MLP podem ter desempenho ruim em *datasets* pequenos, desbalanceados ou ruidosos.

**Tabela 7. Métricas e redução para o *dataset* KronoDroid\_real.device.**

Dataset	Accuracy	Precision	Recall	F1_Score	MCC	ROC_AUC	# caract.	Redução (%)
KronoDroid_real.device	97.60	98.15	97.30	97.72	0.95	97.62	287	0.00
ABC_KronoDroid_real.device	97.47	98.03	97.17	97.60	0.95	97.48	283	1.39
FSDroid_KronoDroid_real.device	92.65	92.32	93.92	93.12	0.85	92.57	32	88.85
JOWNDroid_KronoDroid_real.device	56.42	55.26	93.07	69.34	0.13	54.11	65	77.35
LR_KronoDroid_real.device	93.66	93.86	94.20	94.03	0.87	93.63	103	64.11
RFG_KronoDroid_real.device	73.20	79.11	67.12	72.62	0.47	73.58	22	92.33
SemiDroid_KronoDroid_real.device	95.68	95.91	95.93	95.92	0.91	95.66	30	89.55
MT_KronoDroid_real.device	88.02	88.08	89.49	88.78	0.76	87.93	10	96.52
SigPID_KronoDroid_real.device	93.60	94.19	93.69	93.94	0.87	93.59	18	93.73

O KronoDroid (Tabela 7) apresenta resultados similares aos do DefenseDroid, ou seja, apenas os métodos JOWNDroid e RFG ficam com métricas abaixo de 80%.

É importante destacarmos que os métodos FSDroid e SemiDroid apresentam bons resultados (e.g., precisão e *recall*), com taxas de redução acima de 80%, para a maioria dos *datasets*. Podemos dizer que os dois métodos apresentam uma boa estabilidade e capacidade de generalização, uma descoberta interessante. Ambos os métodos implementam uma seleção de características em dois estágios. Primeiro, realizam o ranqueamento das características utilizando cinco técnicas distintas, incluindo *gain-ratio*, *information gain*, *chi-square*, OneR, PCA e *logistic regression*. O ranqueamento resulta em cinco sub-sets distintos. A partir de um ponto de corte (e.g., 10 melhores características), é realizado uma união dos cinco sub-sets, resultando em um novo sub-grupo de características que é processado utilizando três técnicas distintas, incluindo técnicas de correlação, análise do conjunto aproximado e avaliação de subconjunto de consistência. Como os valores de corte dos dois métodos são distintos, sendo 10 características para o FSDroid e as 10% melhores para SemiDroid, os resultados são ligeiramente diferentes (e.g., FSDroid e SemiDroid selecionam respectivamente 29 e 18 características do *dataset* Adroit).

Finalmente, cabe ressaltar que se o conjunto de dados tiver um desbalanceamento de classes, os modelos poderão ter dificuldade em detectar a classe minoritária, resultando em baixo *recall*, como pode ser observado nos resultados do Adroit e Androcrawl, os dois *datasets* com maior nível de desbalanceamento de classes. Isso gera uma discussão

em torno da seleção de características de cada método, que também pode ser afetada pelo viés do desbalanceamento do *dataset*. Por exemplo, o MT utiliza um *threshold* de seleção baseado na frequência da característica, o que pode implicar em não selecionar características que são importantes para detectar *malwares*.

### 3. Considerações Finais

Os resultados experimentais com a FS3E-v2 nos permitem apresentar *insights* importantes em relação aos métodos de seleção de características para o domínio de detecção de *malwares* Android. Por exemplo, os métodos FSDroid e SemiDroid apresentaram a maior estabilidade e capacidade de generalização para os seis *datasets*, atingindo taxas de redução muito boas (e.g., acima de 80%). Um segundo exemplo é o fato de métodos como o RFG e MT serem pouco adequados para *datasets* que contêm características criadas por especialistas, como foi o caso do conjunto de dados Androcrawl.

Como trabalhos futuros podemos destacar: (a) incorporação de mais métodos de seleção de características; (b) inclusão de *datasets* modernos e atualizados na avaliação, como o MH-100K [Bragança et al., 2023a]; (c) utilização de técnicas de explicabilidade (XAI) para entender o comportamento dos métodos de seleção em determinados conjuntos de dados [Bragança et al., 2023b]; (d) avaliar os métodos de seleção em um ambiente com coleta de dados em tempo real; e (e) investigar como ataques adversariais podem influenciar a seleção das características.

**Agradecimentos.** Esta pesquisa foi parcialmente financiada, conforme previsto nos Arts. 21 e 22 do decreto no. 10.521/2020, nos termos da Lei Federal no. 8.387/1991, através do convênio no. 003/2021, firmado entre ICOMP/UFAM, Flextronics da Amazônia Ltda e Motorola Mobility Comércio de Produtos Eletrônicos Ltda, e apoiada pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Código de Financiamento 001.

### Referências

- Bhat, P. and Dutta, K. (2022). A multi-tiered feature selection model for android malware detection based on feature discrimination and information gain. *Journal of King Saud University*, 34(10, Part B).
- Bragança, H., Rocha, V., Souto, E., Kreutz, D., and Feitosa, E. (2023a). Capturing the behavior of android malware with mh-100k: A novel and multidimensional dataset. In *XXIII SBSeg*.
- Bragança, H., Rocha, V., Souto, E., Kreutz, D., and Feitosa, E. (2023b). Explaining the effectiveness of machine learning in malware detection: Insights from explainable AI. In *XXIII SBSeg*.
- Costa, E., Kreutz, D., Rocha, V., Leão, L., Sabóia, S., Neves, N., and Feitosa, E. (2022). FS3E: uma ferramenta para execução e avaliação de métodos de seleção de características para detecção de malwares android. In *XXII SBSeg*, pages 151–158. SBC.
- HEAVY.AI (2023). Feature selection definition? <https://www.heavy.ai/technical-glossary/feature-selection>.
- Mahindru, A. and Sangal, A. (2021a). FSDroid: A feature selection technique to detect malware from android using machine learning techniques. *Multimedia Tools and Applications*, 80(9):13271–13323.
- Mahindru, A. and Sangal, A. (2021b). SemiDroid: a behavioral malware detector based on unsupervised machine learning techniques using feature selection approaches. *Int. J. of ML and Cybernetics*, 12.
- Mohammadi, F. G., Shenavarmasouleh, F., Amini, M. H., and Arabnia, H. R. (2020). Malware detection using artificial bee colony algorithm. In *ACM Int. Conf. on Pervasive and Ubiquitous Computing*. ACM.
- Soares, T., Kreutz, D., Rocha, V., Costa, E., Leão, L., Pontes, J., Assolin, J., Rodrigues, G., and Feitosa, E. (2022). Uma análise de métodos de seleção de características aplicados à detecção de malwares android. In *XXII SBSeg*, pages 288–301. SBC.