Utilizando Algoritmos de *Machine Learning* Caixa Branca para Avaliar a Manutenibilidade dos Modelos de *Features*

Públio Silva[1], **Carla Bezerra**[1], **Ivan Machado**[2]

[1]Universidade Federal do Ceará (UFC) – Campus Quixadá

[2]Universidade Federal da Bahia (UFBA)

publio.blenilio@gmail.com, carlailane@gmail.com, ivan.machado@ufba.br

Abstract. *This work aimed to use white box Machine Learning (ML) models to classify the Feature Model (FM) maintainability, based on 15 maintainability measures. As a result, the decision tree algorithm was used, which obtained accuracy, precision and recall of 0.81, F1 of 0.79 and AUC-ROC of 0.91. Using this model, there was a reduction in the number of measures needed to assess the maintainability of the FM from 15 to 9 measures. The decision tree generated by the algorithm was used to create a mechanism capable of providing suggestions for changes in maintainability measures so that the maintainability of the artifact improves. A tool with an algorithm chosen for automating the classification of the FM was also implemented.*

Resumo. *O objetivo deste trabalho foi utilizar modelos de Machine Learning (ML) caixa branca para classificar a manutenibilidade do Modelo de Features (MF), com base em 15 medidas de manutenibilidade. Como resultados, foi utilizado o algoritmo de árvore de decisão que obteve acurácia e precisão de 0,81, F1 de 0,79 e AUC-ROC de 0,91. Utilizando este modelo, houve redução da quantidade de medidas necessárias para avaliar a manutenibilidade do MF de 15 para 9 medidas. A árvore de decisão gerada pelo algoritmo foi utilizada para criar um mecanismo capaz de fornecer sugestões de mudança nas medidas de manutenibilidade para que a manutenibilidade do artefato melhore. Também foi implementada uma ferramenta com algoritmo escolhido para automação da classificado do MF.*

1. Introdução

Um dos ativos importantes de uma Linha de Produto de Software (LPS) é o modelo de *features* (MF), que representa as *features* do domínio e a variabilidade de uma LPS [Bezerra et al. 2017]. Uma das características de qualidade mais críticas do MF é a manutenibilidade. Isso ocorre devido às LPSs se modificarem constantemente, o que gera a mudança da estrutura do MF, impactando fortemente na sua manutenibilidade [Bagheri 2011, Bezerra et al. 2015, Bezerra et al. 2017]. A evolução da linha de produto afeta diretamente a complexidade e a manutenção do MF, uma vez que há inserção, exclusão e alteração de *features* ao longo da evolução da linha [Passos et al. 2015]. De acordo com o estudo de [Bagheri 2011], algumas medidas estruturais podem ser consideradas como uma boa forma de prever a complexidade dos MFs para avaliação da capacidade de manutenção de uma LPS. [Bezerra et al. 2015] realizaram um mapeamento

sistemático com o objetivo de identificar medidas de qualidade para MF e apresentaram um catálogo resultante com 32 medidas. Contudo, ainda é difícil avaliar a qualidade geral de um MF utilizando-se dessa estratégia, pois cada medida tem foco em uma característica específica do modelo e não na qualidade do modelo todo. Além disso, em geral as faixas de valores das medidas são amplas e não há um indicativo claro de quais valores podem ser considerados adequados ou inadequados [Oliveira e Bezerra 2019].

Um modo de superar os problemas supracitados consiste na agregação de várias medidas em um valor único que indique a manutenibilidade geral de um MF. [Oliveira e Bezerra 2019] utilizaram tal estratégia, ao aplicar lógica *fuzzy* a um conjunto de 15 medidas de qualidade, de modo a produzir um índice único capaz de indicar o grau de manutenibilidade de um MF. Em um estudo anterior [Silva et al. 2020], aplicamos algoritmos de *clustering* para agrupar os MF de acordo com os valores de um conjunto de medidas de manutenibilidade. A manutenibilidade foi classificada na escala: *muito ruim*, *ruim*, *moderada*, *boa* e *muito boa*. Uma grande dificuldade da abordagem de ML é obter um *dataset* de MFs pré-classificados para ser utilizado no treinamento dos algoritmos. Um modo de classificar os dados seria utilizando especialistas em LPS, mas para conseguir um *dataset* de tamanho razoável seria necessário um número relativamente grande de especialistas, o que torna a opção inviável. Por isso, optou-se neste trabalho por utilizar uma abordagem de classificação automática da manutenibilidade do MF, como a de [Oliveira e Bezerra 2019] ou a proposta em nosso trabalho [Silva et al. 2020] anterior para pré-classificar o *dataset* de treinamento dos algoritmos.

Neste contexto, o objetivo deste trabalho consiste na utilização de algoritmos de ML caixa branca para classificar a manutenibilidade de MFs e utilizar os dados obtidos da execução dos modelos de classificação para fornecer aos engenheiros de domínio indicativos de melhoria do MF por meio de sugestões de mudança nas medidas de manutenibilidade. Para isso, foi realizada uma comparação entre as abordagens de classificação da manutenibilidade do MF proposta por [Oliveira e Bezerra 2019] e [Silva et al. 2020]. O resultado da comparação foi utilizado como base para escolher uma das duas abordagens para pré-classificar o *dataset*, que foi utilizado no treinamento dos algoritmos de ML. Os modelos de classificação obtidos também foram comparados e um deles foi escolhido para criar um mecanismo de sugestão de melhoria da manutenibilidade do MF.

Este artigo corresponde uma versão resumida do trabalho de Iniciação Científica (IC) do aluno, obtendo resultados que foram publicados em um simpósio nacional [Silva et al. 2020] e na principal conferência internacional da área de LPS [Silva et al. 2021], além de outras publicações fora do contexto deste trabalho [Bezerra et al. 2021, Uchôa et al. 2020]. Além disso, o aluno foi agraciado com a menção honrosa *(summa cum laude)* no curso de graduação de Engenharia de Software da UFC Quixadá, pelo seu excelente desempenho acadêmico.

2. Trabalhos Relacionados

[El-Sharkawy et al. 2019] realizaram uma revisão sistemática da literatura para identificar medidas de variabilidade desenvolvidas especificamente para as necessidades de LPS, considerando modelos de variabilidade e artefatos de código. Os autores analisaram 42 estudos primários, nos quais foram identificadas 52 medidas para modelos de variabilidade, 80 para artefatos de código e 10 para ambos. No presente trabalho, foram utilizadas

medidas do MF para avaliar a manutenibilidade da LPS, um *gap* identificado no trabalho de [El-Sharkawy et al. 2019]. Além disso, as medidas de manutenibilidade também foram agregadas visando gerar um único valor para indicar a manutenibilidade de um MF.

[Oliveira e Bezerra 2019] desenvolveram um índice de manutenibilidade do MF através da agregação de um subconjunto das medidas propostas por [Bezerra et al. 2015], utilizando lógica *fuzzy*. O índice foi aplicado a um conjunto do MF e foi possível verificar que esse era capaz de medir se um modelo tinha alta ou baixa manutenibilidade. O presente trabalho utilizou o mesmo subconjunto de medidas utilizado por [Oliveira e Bezerra 2019], pois em [Bezerra et al. 2017] foi feita uma análise que revelou a existência de correlações entre as medidas do catálogo COfFEE e que é possível avaliar a manutenibilidade do MF apenas com as 15 medidas utilizadas por [Oliveira e Bezerra 2019]. O catálogo do subconjunto de medidas foi denominado MiniCOfFEE. A principal diferença do presente trabalho para o trabalho de [Oliveira e Bezerra 2019] é o tipo de técnica utilizada para agregar as medidas para permitir avaliar a manutenibilidade do MF.

[Silva et al. 2020] investigaram a utilização de técnicas de ML na avaliação de LPS. Os autores propuseram uma abordagem para classificação da manutenibilidade dos MFs que utiliza algoritmos de *clustering* para agrupar os MFs por similaridade. Foi utilizado o método Vale [Vale et al. 2019] de derivação de *thresholds* para analisar cada grupo dos MFs e atribuir a cada um deles um rótulo que indicasse a manutenibilidade dos membros do grupo. A abordagem foi avaliada em termos de acurácia e precisão, com valores 61,9% e 70,2%, respectivamente. O *dataset* utilizado no estudo considerava o catálogo de medidas de manutenibilidade proposto por [Bezerra et al. 2017]. Neste trabalho, também é utilizado o mesmo catálogo de medidas. No entanto, são utilizados modelos de aprendizado supervisionado para avaliar a manutenibilidade do MF, ao contrário do trabalho anterior, onde foram utilizados modelos de aprendizado não-supervisionado.

3. Comparando as Abordagens de Avaliação da Manutenibilidade do MF

Um dos objetivos do presente trabalho é utilizar modelos de ML caixa branca para classificar a manutenibilidade dos MFs. Este é um problema de classificação, um típico problema de aprendizado supervisionado. Este tipo de problema requer um *dataset* com os valores das variáveis independentes e da variável dependente para cada amostra. Portanto, para o presente trabalho é necessário um *dataset* contendo os valores das 15 medidas de manutenibilidade do catálogo MiniCOfFEE. Essas medidas representam as variáveis independentes do problema e a classificação da manutenibilidade de cada MF, que representa a variável dependente do problema. Identificamos duas abordagens na literatura, propostas por [Silva et al. 2020] (Abordagem Silva) e [Oliveira e Bezerra 2019] (Abordagem Oliveira). Ambas recebem como entrada as 15 medidas do catálogo MiniCOfFEE e a partir disso classificam a manutenibilidade de um MF. Nesta seção, descrevemos os procedimentos realizados para comparar as duas abordagens e decidir qual delas será utilizada para a pré-classificação do *dataset* de treinamento dos algoritmos de ML.

3.1. Questões de Pesquisa

Para guiar a comparação das duas abordagens, foram definidas três questões de pesquisa:

- **QP3.1:** *Com que frequência cada abordagem classifica a manutenibilidade de um MF da mesma forma que especialistas em LPS?*

- **QP3.2:** *As abordagens de avaliação da manutenibilidade dos MFs são mais pessimistas ou otimistas em relação às classificações feitas por especialistas em LPS?*
- **QP3.3:** *Qual a variação das classificações de cada abordagem em relação às classificações feitas por especialistas em LPS?*

3.2. Resultados

3.2.1. Classificação dos MFs

A avaliação da manutenibilidade dos MFs foi realizada por 15 especialistas. Utilizamos 50 MFs neste estudo. Dentre os MFs que foram avaliados pelos especialistas, selecionamos apenas aqueles que houveram consenso entre os 3 especialistas quanto a classificação da manutenibilidade, ou aqueles onde 2 especialistas concordaram e o terceiro classificou o MF com uma classificação vizinha a escolhida pelos outros 2 especialistas. Ao utilizar estes procedimentos foram selecionados 28 MFs para formar o oráculo de comparação. A Tabela 1 apresenta o resultado das classificações das abordagens Silva e Oliveira e as classificações feitas pelos especialistas em LPS. Com os resultados, foi possível calcular as 3 métricas que definimos anteriormente para comparar as abordagens Silva e Oliveira. A Tabela 2 apresenta os valores das 3 métricas para cada abordagem.

Tabela 1. Comparação entre classificações feitas por especialistas, pela abordagem Silva e pela abordagem Oliveira

Modelo de *Features*	Especialistas	Abordagem Silva	Abordagem Oliveira	Número de Features
Lucas - pnp	Boa	Boa	Muito Boa	25
Test Collaboration System	Muito Boa	Ruim	Muito Boa	16
PGE-FeatureModel_V1	Moderada	Boa	Ruim	48
FQAs v2 7ctcs	Moderada	Muito Boa	Muito Ruim	178
Alarm System	Boa	Moderada	Muito Boa	13
Experiment environment	Moderada	Ruim	Ruim	35
Bike Shop	Boa	Ruim	Boa	21
Toko	Ruim	Boa	Ruim	72
ProfileItems	Muito Ruim	Ruim	Muito Ruim	78
Documentation_Generation	Ruim	Ruim	Ruim	44
WebCollaborativeApplication	Boa	Boa	Ruim	40
e-foramation	Boa	Ruim	Boa	26
Warren	Moderada	Ruim	Ruim	37
Thread	Moderada	Ruim	Moderada	44
Buzzard Radio play DB	Moderada	Moderada	Moderada	14
MayaModel	Moderada	Boa	Ruim	57
ModelTransformation	Muito Ruim	Muito Boa	Muito Ruim	88
SmartPhone Device	Boa	Ruim	Muito Boa	17
Hastane Randevu Sistemi	Boa	Ruim	Muito Boa	17
ScrollingText	Ruim	Ruim	Muito Boa	27
Student Attendance System	Moderada	Moderada	Muito Boa	23
Simple Drawing Tools	Boa	Moderada	Muito Boa	14
Speech Recognition	Moderada	Ruim	Muito Ruim	75
Windows70_Contrastn	Moderada	Ruim	Muito Ruim	59
Owncloud	Moderada	Ruim	Ruim	37
WebApp	Muito Boa	Ruim	Muito Boa	10
VMTools-RA	Moderada	Moderada	Muito Ruim	40
Jogo de Tiros	Moderada	Moderada	Moderada	37

Tabela 2. Comparação das métricas entre as abordagens Silva e Oliveira

Métrica	Abordagem Silva	Abordagem Oliveira
Acurácia das classificações de uma abordagem em relação às classificações dos especialistas em LPS	0,28	0,39
Taxa de otimismo e pessimismo de uma abordagem	-0,28	-0,10
Média das distâncias das classificações de uma abordagem para as classificações dos especialistas em LPS	1,25	0,89

3.2.2. QP3.1: Frequência da Classificação da Manutenibilidade do MF

Para responder a QP3.1, calculamos a acurácia das classificações de uma abordagem em relação às classificações dos especialistas em LPS. Para obter o valor desta métrica, somamos a quantidade de vezes que cada abordagem classificou um MF com a mesma classe definida pelos especialistas e dividimos pela quantidade total dos MFs no oráculo de comparação. Os dados da Tabela 1 indicam que a abordagem Silva classificou 8 MFs com a mesma classe definida pelos especialistas e a abordagem Oliveira classificou 11 MFs. Portanto, como é possível observar na Tabela 2, a acurácia da abordagem Silva em relação às classificações dos especialistas em LPS é 0,28 e a da abordagem Oliveira 0,39. O resultado mostra que a acurácia da abordagem Oliveira em relação às classificações dos especialistas em LPS é maior que a da abordagem Silva, o que significa que a abordagem Oliveira classifica mais vezes com a mesma classe definida por especialistas em LPS.

3.2.3. QP3.2: Otimismo ou Pessimismo das Abordagens em Relação às Classificações dos Especialistas

Para responder a QP3.2, calculamos a métrica da taxa de otimismo e pessimismo de uma abordagem. Para obter valor da métrica contamos a quantidade de vezes em que cada abordagem classificou com otimismo em relação a classificação dos especialistas (por exemplo, os especialistas classificaram um MF como moderado e a abordagem classificou como bom), subtraímos pela quantidade de vezes em que a abordagem classificou com pessimismo em relação a classificação dos especialistas (por exemplo, os especialistas classificaram um MF como moderado e a abordagem classificou como ruim) e dividimos pela quantidade total do MF no oráculo de comparação. Ao analisar os dados da Tabela 1, é possível observar que a abordagem Silva apresenta 14 classificações pessimistas e 6 classificações otimistas; na abordagem Oliveira, 10 classificações foram pessimistas e 6 foram otimistas. A Tabela 2 indica que o valor da métrica para a abordagem Silva foi -0,28 e para a abordagem Oliveira foi -0,10. O resultado indica que ambas as abordagens classificam em geral com pessimismo em relação à classificação feita por especialistas em LPS, o que favorece as duas abordagens, pois, ao fornecer um resultado pessimista os engenheiros de domínio serão alertados de que o MF precisa ser melhorado.

3.2.4. QP3.3: Variação das Classificações das Abordagens e dos Especialistas

Para responder a QP3.3, calculamos a terceira métrica da média das distâncias das classificações de uma abordagem para as classificações dos especialistas em LPS. Para obter o valor da métrica somamos as distâncias entre as classificações de cada abordagem e as classificações dos especialistas para cada MF (por exemplo, se os especialistas classificaram um MF como ruim e uma abordagem classificou como muito bom a distância é 3) e dividimos pela quantidade total dos MFs no oráculo de comparação. Como é possível observar na Tabela 2, a média das distâncias na abordagem Silva foi 1,25 e na abordagem Oliveira foi 0,89. O resultado mostra que em geral as classificações da abordagem Oliveira estão mais próximas da classificação dos especialistas em LPS que a abordagem Silva. Este valor é influenciado também pela quantidade de vezes em que a abordagem Oliveira classificou com a mesma classe definida pelos especialistas.

4. Avaliando a Manutenibilidade do MF Utilizando Modelos de ML

Um dos grandes problemas encontrados quando se faz uso de modelos de ML para resolver os mais diversos tipos de problemas é a chamada "maldição da dimensionalidade". Este termo faz referência a relação existente entre a quantidade de variáveis envolvidas em um determinado problema e o custo computacional que é necessário para resolvê-lo [Han et al. 2017]. No estudo realizado por [Bezerra et al. 2017] foi feita a redução de catálogo com 32 medidas para um com 15 medidas que é o MiniCOfFEE. As 15 medidas de manutenibilidade do MF do catálogo MiniCOfFEE serão utilizadas como as variáveis independentes do problema de ML abordado no presente trabalho. Apesar do catálogo MiniCOfFEE já se tratar de uma redução de um catálogo de medidas, desejamos verificar se é possível reduzir ainda mais o número de medidas necessárias para classificar a manutenibilidade dos MFs.

4.1. Questões de Pesquisa

Para guiar o processo de criação dos modelos de ML para avaliação da manutenibilidade dos MFs, foram definidas duas questões de pesquisa:

- **QP4.1:** É possível reduzir a quantidade de medidas de manutenibilidade necessárias para avaliar a manutenibilidade de um MF?
- **QP4.2:** Qual o melhor modelo de ML para avaliação da manutenibilidade do MF?

4.2. Resultados

4.2.1. QP4.1: Redução de Medidas de Manutenibilidade para Avaliar o MF

Para responder a QP4.1 utilizamos duas abordagens de seleção de variáveis independentes[1]. Na primeira abordagem utilizamos o coeficiente de *Spearman* [Schober et al. 2018] para analisar a correlação entre as 15 medidas do catálogo MiniCOfFEE e a classificação da manutenibilidade dos MFs. Selecionamos apenas as medidas para as quais o valor absoluto do coeficiente de correlação fosse maior ou igual a 0,6, ficando assim apenas com as medidas com correlação forte ou muito forte [Salkind e Rainwater 2006] com a classificação da manutenibilidade dos MFs. Utilizando este procedimento foram selecionadas as medidas NF, NLeaf, DTMax, CogC, FEX e NVC.

Na segunda abordagem de seleção de variáveis independentes, utilizamos um atributo de importância das variáveis independentes que é fornecido por alguns algoritmos de ML, como é o caso dos algoritmos de regressão logística e árvore de decisão. Com o algoritmo de regressão logística, obtivemos um conjunto de medidas de manutenibilidade formado pelas medidas NF, NM, NTop, NLeaf DTMax, CogC e FEX. Com o algoritmo de árvore de decisão obtivemos um conjunto de medidas de manutenibilidade formado pelas medidas SCDF, CogC, RDen, NM, FoC, NTop, NLeaf, NF e DTMax.

Os três conjuntos de medidas foram utilizados para treinar modelos de ML. O modelo de ML que obteve melhor taxa de acerto, foi o que utilizou o algoritmo de árvore de decisão. Este modelo de ML utilizou o conjunto de medidas 4 que é composto por 9 das 15 medidas de manutenibilidade que compõem o catálogo MiniCOfFEE. Desse modo, concluímos que é possível reduzir a quantidade de medidas de manutenibilidade necessárias para avaliar a manutenibilidade de um MF.

[1] `https://bit.ly/3JIN2kZ`

4.2.2. QP4.2: Melhor Modelo de ML para Avaliação da Manutenibilidade do MF

Para responder a QP4.2 realizamos validação cruzada com 10 etapas para os 3 modelos de ML criados (utilizando os algoritmos *Naive Bayes*, regressão logística e árvore de decisão). Calculamos as 5 métricas (acurácia, precisão, recall, F1 e AUC-ROC) em cada etapa, e ao final obtivemos o valor mínimo, o valor máximo, a média e o desvio padrão de cada métrica nas 10 etapas[2] [3] [4]. O modelo que obteve melhores resultados para todas as métricas foi o que utilizou o algoritmo de árvore de decisão, seguido pelos algoritmos *Naive Bayes* e regressão logística. Todas as métricas ficaram próximas ou acima de 0,80 para o modelo de árvore de decisão, inclusive as métricas de acurácia e precisão. Isso quer dizer que a taxa de acerto deste modelo de ML é maior que a do classificador da manutenibilidade dos MFs gerado em nosso trabalho anterior [Silva et al. 2020].

Ao comparar os valores das métricas precisão, *recall* e F1 dos 3 modelos de ML, é possível afirmar que o modelo de árvore é o menos suscetível a erros por falso positivo e falso negativo. Com os valores da métrica AUC-ROC, é possível afirmar que o modelo de árvore de decisão é também o que melhor separa as 5 classes de manutenibilidade. A Figura 1 mostra como os valores das 5 métricas de classificação variam nas 10 etapas da validação cruzada nos 3 modelos de ML. É possível observar que praticamente todas as etapas o modelo de árvore de decisão se mantém com bons valores para todas as métricas de classificação, e geralmente acima dos valores dos outros dois modelos. Isso indica que o modelo de árvore de decisão tem boa taxa de acerto em *datasets* compostos por MFs com diferentes características e é, portanto, generalizável.

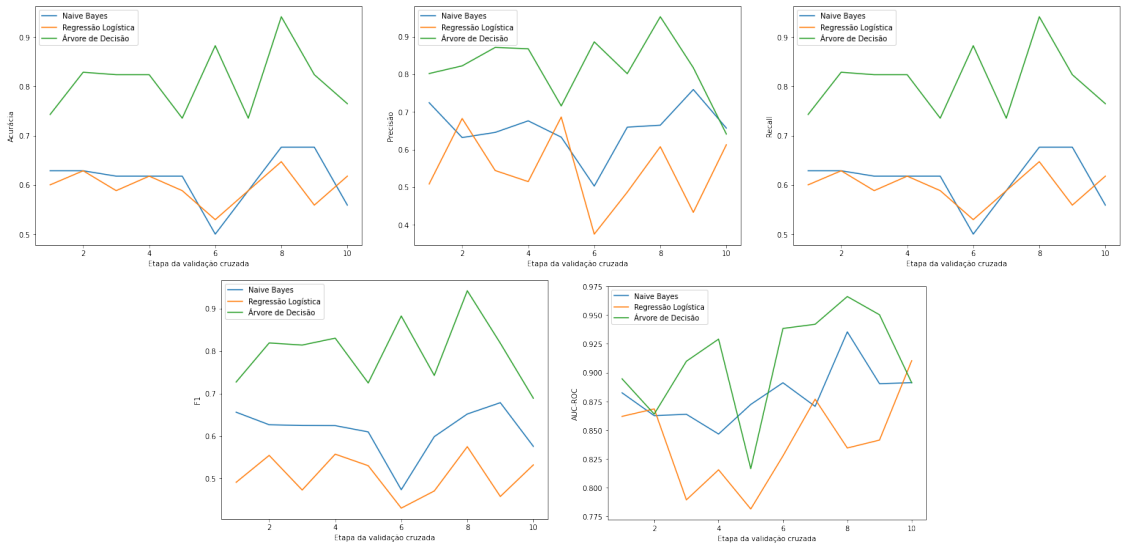


Figura 1. Valores das 5 métricas de classificação variam nas 10 etapas da validação cruzada nos 3 modelos de ML

Com isso, concluímos que dentre os 3 modelos de ML para classificação da manutenibilidade do MF gerados, o modelo de ML que utilizou o algoritmo de árvore de decisão é o melhor, pois este obteve os melhores valores em todas as 5 métricas de

[2] https://bit.ly/35bOvkC
[3] https://bit.ly/3IFTgRm
[4] https://bit.ly/3IHfMZS

classificação utilizadas. Além disso, o modelo de árvore de decisão mostrou ser generalizável pois obteve bons valores para todas as 5 métricas de classificação em praticamente todas as 10 etapas da validação cruzada.

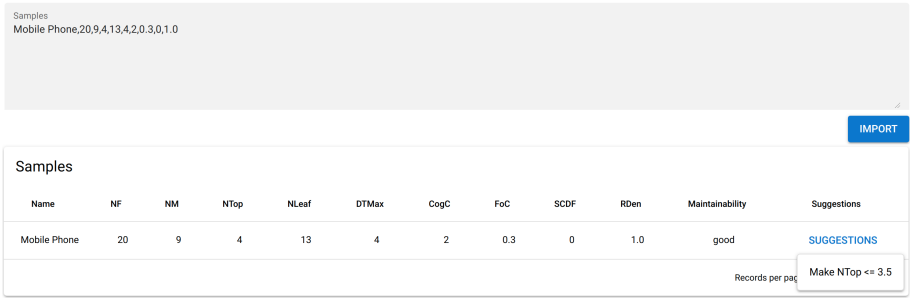
4.2.3. Criação do Mecanismo para Fornecer Indicativos de como Melhorar a Manutenibilidade do MF

Como o algoritmo de árvore de decisão é de caixa branca é possível extrair dados que permitem interpretar e explicar os resultados obtidos. Para classificar uma amostra do *dataset*, o algoritmo de árvore de decisão gera uma estrutura de árvore binária, onde cada nó folha contém uma possível classe e os nós restantes contém condições que determinam o próximo nó a ser verificado. A estrutura de árvore gerada pelo modelo de ML criado no presente trabalho, que utilizou o algoritmo de árvore de decisão.

Foi criada uma função em JavaScript que implementa o processo de classificação da manutenibilidade dos MFs considerando a árvore de decisão gerada[5]. A árvore de decisão também foi utilizada para criar uma outra função em JavaScript para encontrar um nó folha (o mais próximo) que contenha uma classificação de manutenibilidade melhor que a obtida inicialmente. Após encontrar esse nó, a função retorna as regras que precisam ser atendidas para que o MF receba a nova classificação. Desse modo, o retorno da função indica o valor para o qual as medidas de manutenibilidade precisam aumentar ou diminuir para que a manutenibilidade do MF aumente.

As duas funções foram integradas a uma página web que permite a importação dos valores de medidas de manutenibilidade de um ou mais MFs e exibe a classificação da manutenibilidade dos MFs e indicativos de como melhorar a manutenibilidade do artefato[6]. A Figura 2 mostra um MF classificado pelo modelo de ML de árvore de decisão como bom. Foi dada a sugestão que o valor da medida NTop fosse reduzido de 4 para um valor menor ou igual a 3,5. A Figura 3 mostra o mesmo MF com o valor da medida NTop sendo 3 e agora sendo classificado como muito bom.

Figura 2. MF antes das sugestões fornecidas pelo mecanismo serem aceitas.

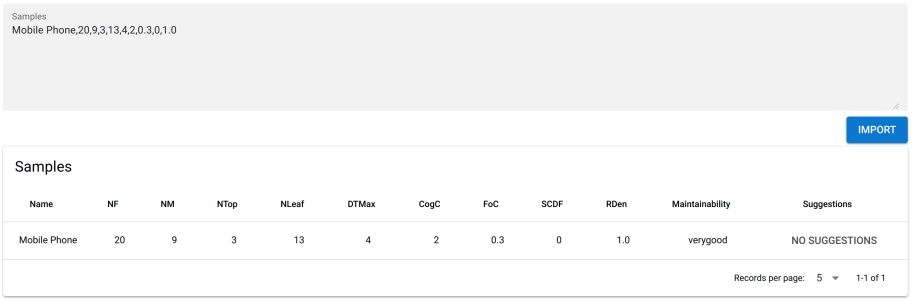


5. Conclusão

Neste trabalho, utilizamos algoritmos de ML caixa branca para criar modelos capazes de classificar a manutenibilidade de MFs a partir de um conjunto de medidas. Selecionamos

[5] https://bit.ly/3izIul0
[6] https://bit.ly/3Rfqmvy

Figura 3. MF após as sugestões fornecidas pelo mecanismo serem aceitas.



duas abordagens de classificação automática (Oliveira e Silva) e as comparamos a partir de um oráculo de comparação composto por 28 MFs avaliados de forma manual por 15 especialistas em LPS. A abordagem Oliveira obteve melhores resultados na comparação, e foi então utilizada para pré-classificar o *dataset*.

Verificamos a possibilidade de avaliar a manutenibilidade de um MF com um subconjunto de medidas menor do que o conjunto formado pela 15 medidas de manutenibilidade. No total, foram criados 3 modelos de ML para classificação da manutenibilidade dos MFs: (1) *Naive Bayes*; (2) regressão logística; e, (3) árvore de decisão. O modelo de ML que obteve os melhores resultados foi o que utilizou o algoritmo de árvore de decisão que ficou com acurácia média de 0,81, precisão de 0,81, *recall* de 0,81, F1 de 0,79 e AUC-ROC de 0,91. Esses valores indicam que o modelo de ML é pouco suscetível a erros por falso negativo e falso positivo e que consegue separar bem as 5 classes de manutenibilidade do MF. Com o modelo de árvore de decisão foi possível também realizar uma redução de 6 variáveis independentes. O algoritmo de árvore de decisão gera uma estrutura de árvore que é utilizada no processo de classificação. Essa estrutura foi utilizada como base para a criação de um mecanismo capaz de fornecer indicativos sobre o que precisa ser feito para melhorar a manutenibilidade de um MF. Os indicativos fornecidos dizem um valor para o qual um conjunto de medidas de manutenibilidade deve aumentar ou diminuir para que a manutenibilidade de um MF melhore.

O modelo de classificação proposto permite que engenheiros de domínio avaliem rapidamente a manutenibilidade de um MF, facilitando a tomada de ações corretivas no início do ciclo. Como trabalhos futuros, podemos apontar: (i) validar o mecanismo criado no presente trabalho, (ii) investigar a conversão de sugestões de mudanças nos valores das medidas de manutenibilidade em sugestões de mudanças no próprio MF; (iii) avaliar o esforço necessário para alterar um MF de modo que as sugestões de mudanças nos valores das medidas de manutenibilidade sejam atendidas; (iv) realizar mais ajustes e otimizações nos hiperparâmetros dos algoritmos de ML visando aumentar a qualidade dos modelos de ML obtidos; e, (v) utilizar na seleção de variáveis independentes abordagens agnósticas de modelo para medir a importância das variáveis.

Referências

Bagheri, E. e Gasevic, D. (2011). Assessing the maintainability of software product line feature models using structural metrics. *Software Quality Control*, 19(3):579–612.

Bezerra, C., Lima, R., and Silva, P. (2021). Dymmer 2.0: A tool for dynamic modeling and evaluation of feature model. In *Proceedings of the XXXV Brazilian Symposium on*

Software Engineering, SBES '21, page 121–126, New York, NY, USA. Association for Computing Machinery.

Bezerra, C. I., Andrade, R., and Monteiro, J. M. S. (2015). Measures for quality evaluation of feature models. In *International Conference on Software Reuse*, pages 282–297. Springer.

Bezerra, C. I., Andrade, R. M., and Monteiro, J. M. (2017). Exploring quality measures for the evaluation of feature models: a case study. *Journal of Systems and Software*, 131:366–385.

El-Sharkawy, S., Yamagishi-Eichler, N., and Schmid, K. (2019). Metrics for analyzing variability and its implementation in software product lines: A systematic literature review. *Information and Software Technology*, 106:1–30.

Han, J., Jentzen, A., and Weinan, E. (2017). Overcoming the curse of dimensionality: Solving high-dimensional partial differential equations using deep learning. *arXiv preprint arXiv:1707.02568*, pages 1–13.

Oliveira, D. C. S. and Bezerra, C. I. M. (2019). Development of the maintainability index for spls feature models using fuzzy logic. In *Proceedings of the XXXIII Brazilian Symposium on Software Engineering*, SBES 2019, page 357–366, New York, NY, USA. Association for Computing Machinery.

Passos, L., Teixeira, L., Dintzner, N., Apel, S., Wasowski, A., Czarnecki, K., Borba, P., and Guo, J. (2015). Coevolution of variability models and related software artifacts. *Empirical Software Engineering*, pages 1–50.

Salkind, N. J. and Rainwater, T. (2006). *Exploring research*. Pearson Prentice Hall Upper Saddle River, NJ.

Schober, P., Boer, C., and Schwarte, L. A. (2018). Correlation coefficients: appropriate use and interpretation. *Anesthesia & Analgesia*, 126(5):1763–1768.

Silva, P., Bezerra, C. I., and Machado, I. (2021). A machine learning model to classify the feature model maintainability. In *Proceedings of the 25th ACM International Systems and Software Product Line Conference-Volume A*, pages 35–45.

Silva, P., Bezerra, C. I. M., Lima, R., and Machado, I. (2020). Classifying feature models maintainability based on machine learning algorithms. In *Proceedings of the 14th Brazilian Symposium on Software Components, Architectures, and Reuse*, SBCARS '20, page 1–10, New York, NY, USA. Association for Computing Machinery.

Uchôa, A., Barbosa, C., Oizumi, W., Blenilio, P., Lima, R., Garcia, A., and Bezerra, C. (2020). How does modern code review impact software design degradation? an in-depth empirical study. In *2020 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 511–522.

Vale, G., Fernandes, E., and Figueiredo, E. (2019). On the proposal and evaluation of a benchmark-based threshold derivation method. *Software Quality Journal*, 27(1):275–306.