

INCORPORAÇÃO DE DIFERENTES ALGORITMOS DE APRENDIZAGEM DE MÁQUINA NO MÓDULO DE APRENDIZAGEM DA FERRAMENTA OPLA-TOOL

Fernando Henrique Kuviatkovski (PIBIC/CNPq/FA/Uem), Aline Maria Malachini Miotto Amaral, ra100285@uem.br.

Universidade Estadual de Maringá / Centro de Tecnologia / Maringá, PR.

Ciência da Computação / Engenharia de Software

Palavras-chave: Linha de Produto de Software, Otimização Iterativa, Aprendizagem de Máquina.

Resumo

Linha de Produto de Software (LPS) é uma estratégia de Engenharia de Software utilizada para promover o reuso de componentes de software. A MOA4PLA (*Multi-Objective Approach for Product-Line Architecture Design*) suportada pela ferramenta OPLA-Tool é uma abordagem para otimização de Arquiteturas de LPS por meio de algoritmos evolutivos multiobjetivos, como o NSGA-II. Em sua versão atual, a MOA4PLA permite ao tomador de decisão (*Decision Maker* - DM) a interação durante todo o processo, tornando-a uma abordagem de otimização interativa: no início, com a configuração de parâmetros; durante, com a avaliação das arquiteturas; ao fim, com a escolha das melhores soluções. Para isto, várias modificações foram realizadas nos algoritmos evolutivos, bem como um modelo de Aprendizagem de Máquina (AM) foi incorporado na ferramenta OPLA-Tool em sua versão 2.0. Este modelo de AM tem como objetivo reduzir a fadiga que o processo avaliativo causa no DM, por se tratar de uma tarefa que demanda grande esforço cognitivo e ocorre diversas vezes. Nesse contexto, com base em resultados obtidos em trabalhos anteriores que avaliaram diferentes algoritmos de AM através de métricas de acurácia e tempo, este trabalho propôs a implementação daqueles que apresentaram os melhores desempenhos, a fim de aprimorar a qualidade das arquiteturas otimizadas.

Introdução

Linha de Produto de Software (LPS) é uma estratégia de Engenharia de Software utilizada para promover o reuso, tal objetivo é atingido principalmente pela elaboração e reaproveitamento da Arquitetura de Linha de Produto de Software (ALPS), definida como um conjunto de componentes que representam características comuns e pontos de variação dentro de um domínio bem definido.

A MOA4PLA (*Multi-Objective Approach for Product-Line Architecture Design*), presente na ferramenta OPLA-Tool v2.0 (FREIRE. 2021), diz respeito a uma abordagem para otimização de ALPS's com base em diversas métricas arquiteturais de forma simultânea por meio de algoritmos evolutivos multiobjetivos, como o NSGA-II. Em sua versão interativa a ferramenta OPLA-Tool suportava inicialmente

apenas de um único algoritmo de AM (MLP (2500)) para diminuir o processo de fadiga causado durante as interações realizadas pelo DM.

No trabalho de Kuviatkovski et al. (2022) foram testados diferentes algoritmos de AM no contexto da otimização de ALPS. Nesse contexto, com base nos resultados obtidos em Kuviatkovski et al. (2022) este trabalho propõe a incorporação na versão interativa da ferramenta OPLA-Tool v2.0 os seguintes algoritmos de AM: K*, Het-ensemble e Hom-ensemble. Tal resultado foi obtido através da comparação de diversas métricas de acurácia e tempo de execução com o algoritmo MLP (2500), já presente na ferramenta OPLA-Tool v2.0.

A Figura 1 apresenta a comparação desses modelos com a *baseline* através de métricas de acurácia de classificação, como Precision, Recall e F1-Score. Percebe-se que apesar de apresentarem um desempenho próximo de acurácia, porém melhores que a *baseline*, os algoritmos propostos apresentam grande vantagem em outros aspectos também relevantes, como o tempo necessário para aprendizagem, já que durante o processo iterativo o DM deve aguardar o treinamento do modelo. Os modelos *ensembles* propostos são compostos por outros algoritmos AM, sendo suas saídas combinadas por meio de um critério pré-determinado, tornando necessária a implementação dos seguintes: *Least Median of Squares Regression* (LMS), *Support-vector Machines* (SVM), *Random Trees* e *Random Forest*.

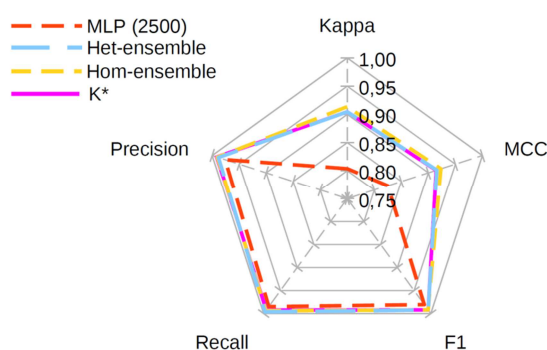


Figura 1 - Comparação de acurácia entre os modelos propostos e *baseline*. (KUVIATKOVSKI, 2022).

Materiais e Métodos

Como o objetivo principal deste trabalho é a implementação dos algoritmos na ferramenta OPLA-Tool v2.0, é necessário compreender o estado atual da arquitetura e as tecnologias já utilizadas na ferramenta, principalmente as seguintes:

- **Angular:** Trata-se de um *framework* para desenvolvimento web baseado em componentes, a linguagem de programação utilizada é TypeScript.
- **Spring:** Trata-se de um *framework* para construir a infraestrutura de de aplicações desenvolvidas na linguagem de programação Java. Na OPLA-Tool v2.0, é utilizada para modelar a API da ferramenta.

- **Weka:** A ferramenta Weka disponibiliza em sua biblioteca a implementação de todos os algoritmos de AM propostos, acessíveis através de sua API.

A Figura 2 apresenta a arquitetura da ferramenta OPLA-Tool v2.0. Neste trabalho, as modificações foram realizadas principalmente no módulo OPLA-Front, com a criação de novos componentes para a interação do usuário, no módulo OPLA-API, para lidar com novas requisições geradas pelo módulo anterior, e no módulo OPLA-Core, para a construção dos modelos de AM com os algoritmos propostos.

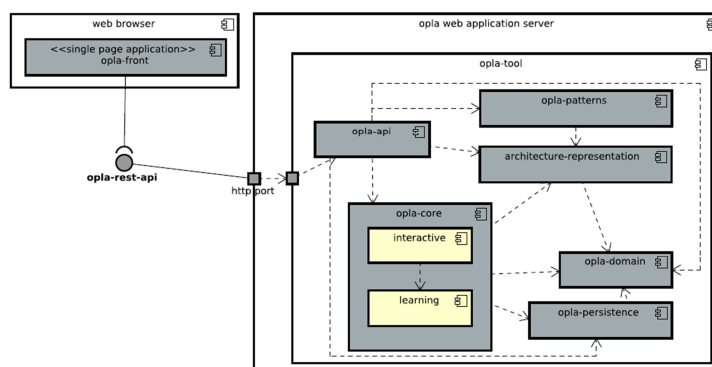


Figura 2 - Arquitetura da ferramenta OPLA-Tool v2.0. (FREIRE. 2021).

Resultados e Discussão

Com as implementações realizadas, a OPLA-Tool v2.0 possibilita o uso de sete algoritmos de AM, bem como a configuração dos parâmetros pertinentes à cada algoritmo. Para isso, a Figura 3 apresenta dois componentes implementados para configurações gerais dos modelos de AM. No componente à esquerda, é possível: habilitar o uso de modelos *ensembles*; carregar ou salvar um arquivo de configurações; selecionar um algoritmo AM e editar seus parâmetros. Caso optado por um modelo *ensemble*, o componente à direita permite a visualização e remoção dos modelos que o compõem, bem como a definição da regra de combinação.

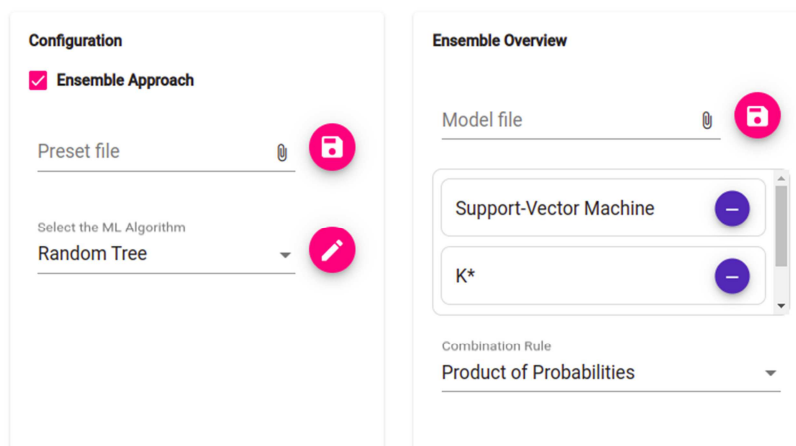


Figura 3 - Componentes implementados para as configurações do modelo de AM.

Já a Figura 4 apresenta um dos componentes implementados para a configuração dos parâmetros dos algoritmos. Os parâmetros de configuração utilizados são conforme as implementações da biblioteca Weka.

Algorithm Parameters		
kValue	breakTiesRandomly	maxDepth
0	False	0
minNum	minVariancePop	numFolds
1.0	0.001	0

Figura 4 - Componente para configuração de parâmetros do algoritmo *Random Tree*.

Conclusões

Podemos afirmar que o objetivo do trabalho foi atingido, visto que as funcionalidades e os algoritmos de AM propostos foram implementados na ferramenta OPLA-Tool v2.0, possibilitando ao DM uma maior customização na otimização interativa, e consequentemente soluções de melhor qualidade.

Agradecimentos

Gostaríamos de agradecer à Fundação Araucária pelo apoio financeiro para o desenvolvimento dessa pesquisa.

Referências

KUVIATKOVSKI, F. H.; FREIRE, W. M.; AMARAL, A. M. M.; COLANZI, T. E.; FELTRIM, V. D. **Evaluating Machine Learning Algorithms in Representing Decision Makers in PLA Design Optimization**. 19th IEEE International Conference on Software Architecture (ICSA), Honolulu, Hawaii (USA), 2022.

FREIRE, WILLIAN MARQUES. **Otimização interativa de arquitetura de linha de produto de software utilizando congelamento de elementos arquiteturais**. 2021. Dissertação (Mestrado) - Curso de Ciência da Computação, Departamento de Informática, Universidade Estadual de Maringá, Maringá, 2021.