**Título**

Geração automática de dados de teste por meio de aprendizado de máquina.

**Tema**

Área: Engenharia de Software  
Tema: Geração Automática de Casos de Teste

**Motivação**

Produtos de software de diferentes tamanhos e complexidades são utilizados todos os dias em atividades profissionais ou de entretenimento. Em qualquer circunstância, entretanto, a falta de qualidade dos produtos pode caracterizar uma situação preocupante para seus produtores [1][10], uma vez que cada vez mais os níveis aceitáveis de qualidade de um software estão aumentando, tanto no que se refere ao seu comportamento observável externamente quanto ao seu processo de desenvolvimento e estrutura interna [2,4,15].  
O teste de software [9] é a maneira mais popular de verificar se um software atende às especificações descritas e cumpre o papel desejado pelos interessados [5]. Testes são costumeiramente vistos durante o processo como atividades dispendiosas no ciclo de vida de software [8], requisitando tempo, esforço, planejamento e execução [9], além de recursos financeiros, como demonstrado pelo estudo de impacto econômico dos testes efetuado pelo NIST (National Institute of Standards and Technology), onde o custo nos Estados Unidos para infraestrutura de testes no ano de 2002 é estimado em um valor entre 22 e 59 bilhões de dólares. Aliados aos altos custos, a complexidade e a criticidade de sistemas orientados a objetos são ampliadas [10], ou seja, os sistemas tornam-se maiores e mais complexos ao passo que os métodos convencionais de validação levam mais tempo e ficam mais caros [9, 10].

Com o aumento da complexidade e do tamanho do software, ficou mais evidente a necessidade de que os projetos de software devam possuir características que os tornem mais fáceis de serem testados [11], já que compreender e reduzir o esforço de testes são elementos principais de interesse tanto no ramo acadêmico quanto industrial [9]

**Lacuna/Problema**

Ao longo dos anos diversas técnicas têm sido propostas para o propósito de de geração automática de dados de teste, estas técnicas abrangem conceitos que vão desde análise de estrutura de código até análise dos modelos de dados utilizados em aplicações, porém a área carece de estudos que evidenciem tanto a prioritização de técnicas em detrimento de outros em casos específicos de geração assim como a seleção ou combinação destas técnicas para atingir a geração de melhores casos de teste e assim maior cobertura de código.

**Objetivo**

Construir um modelo capaz, a partir da análise de características do código em estudo, de selecionar a melhor técnica de geração automática de dados de teste para o software em questão afim de atingir maior cobertura de código.

**Hipóteses**

Hipótese 1 : Métricas de testabilidade de software são os melhores indicadores para seleção de técnicas de geração automática de dados de teste.

Hipótese 2: É possível escolher a melhor técnica de geração automática de dados de teste baseado somente nas métricas de testabilidade extraídas deste código e assim atingir maior cobertura de código com os casos de teste gerados se comparado com as técnicas de geração automática de dados de teste isolodamente.

**Justificativa**

**Método**

1. Fazer um mapeamento sistemático das métricas de testabilidade existentes na literatura.
2. Selecionar as melhores métricas ,a partir do mapeamento sistemático , baseado tanto no uso quanto na sua relevância para seleção, caracteriza-se relevância como o quanto uma métrica de software pode descrever uma característica para seleção.
3. Selecionar softwares candidatos à análise, a partir de softwares open-source disponibilizados por meio de repositórios de código aberto, tais como: GitHub, Bitbucket, entre outros.
4. Extrair as métricas selecionadas dos softwares selecionados para análise.
5. Treinar um modelo afim de selecionar a melhor técnica de geração de dados de teste, baseado nas características do software em questão, características estas representadas por métricas de testabilidade.
6. Gerar os casos de teste a partir do modelo para cada um dos softwares candidatos à análise.
7. Gerar os casos de teste para cada um dos softwares candidatos à análise em cada uma das técnicas de geração automática de dados de teste e também para um modelo de seleção randômica de técnicas de geração automática de dados de teste.
8. Analisar a cobertura de código pontuada por cada um dos conjuntos de casos de teste gerados.
9. Testar a hipótese de maior cobertura de código gerada pelo modelo contra as estratégias de geração de casos de teste isoladamente.

**Avaliação**

**Limitações**

**Contribuições**

Espera-se com este trabalho, contribuir para a área de testes de software, afim de prover não somente uma metodologia para seleção das melhores técnicas de geração automática de teste, mas também prover conhecimento sobre as qualidades presentes no software capazes de influenciar a geração automática de dados de teste.