AutoML baseado na Teoria de Resposta ao Item

Lucas F. F. Cardoso[1,2], **Regiane S. Kawasaki Francês**[1], **Ronnie C. O. Alves**[2]

[1]Faculdade de Computação – Universidade Federal do Pará (UFPA)
Belém – PA – Brasil

[2]Instituto Tecnológico Vale (ITV)
Belém – PA – Brasil.

lucas.cardoso@icen.ufpa.br, kawasaki@ufpa.br, ronnie.alves@itv.org

Abstract. *Generating machine learning models automatically remains an area of much recent research. However, there is still no definitive way to generate simple models that better generalize and that are immune to underspecification. In this work, we present the preliminary results of a proposed methodology that uses the concepts of Item Response Theory and Genetic Algorithm to create a NAS-type AutoML algorithm capable of generating a competitive Neural Network with the models already known in the literature. In the initial results, it was possible to generate a competitive model with AutoKeras, but with a complexity lower than 5.5% of the total complexity of the AutoKeras model itself.*

Resumo. *Gerar modelos de aprendizado de máquina automaticamente segue sendo uma área de muita pesquisa recente. Porém, ainda não há uma forma definitiva capaz de gerar modelos simples que melhor generalizem e que sejam imunes a subespecificação. Neste trabalho, são apresentado os resultados preliminares de uma metodologia proposta que utiliza os conceitos da Teoria de Resposta ao Item e Algoritmo Genético para criar um algoritmo de AutoML do tipo NAS que seja capaz de gerar uma Rede Neural competitiva com os modelos já conhecidos na literatura. Nos resultados iniciais, foi possível gerar um modelo competitivo com o AutoKeras, porém com complexidade inferior a 5.5% da complexidade total do modelo do próprio AutoKeras.*

1. Introdução e Contexto

Devido à grande variedade de aplicações, sistemas que utilizam Aprendizado de Máquina são cada vez mais comuns, resolvendo diversos problemas do mundo real. Porém, à medida que os problemas a serem resolvidos se tornam cada vez mais complexos, também se torna mais complexo desenvolver um modelo de aprendizado de máquina capaz de resolver suficientemente bem uma dada tarefa. Comumente, grandes modelos de alto desempenho só são alcançados após muito tempo de tentativa e erro investido. Essas condições faz com que seja necessário que se possua um alto nível de conhecimento sobre aprendizado de máquina e o problema em questão para poder desenvolver um modelo apto, elevando ainda mais os custos de produção [He et al. 2021].

Diante disso, foram desenvolvidas soluções chamadas de AutoML que visam automatizar o processo de produção de um modelo e que diminua os custos envolvidos no desenvolvimento. De forma simplificada, o AutoML pode ser entendido como sendo um sistema capaz de gerar modelos de aprendizado de alto desempenho sem a necessidade

de interação humana, comumente utilizando técnicas de computação evolucionário como Algoritmos Genéticos (AG). Dentre os diferentes tipos de AutoML existentes um dos mais estudados é a busca por arquitetura neural (NAS) que é voltado principalmente para o aprendizado profundo. Em poucas palavras o NAS visa criar uma arquitetura neural que alcance o melhor resultado possível sobre um determinado problema [Ren et al. 2021].

Mesmo com diversos avanços no campo do AutoML ainda é comum que os modelos gerados sejam subespecificados. A subespecificação de um modelo ocorre quando um modelo tem bom desempenho no treino e validação, porém não é capaz de manter a mesma performance quando posto no mundo real [D'Amour et al. 2020]. Uma das possibilidades para a ocorrência de modelos subespecificados está relacionado as métricas clássicas de avaliação que são utilizadas para mensurar o desempenho dos modelos. No geral, as métricas de avaliação são quantitativas, i.e., visam selecionar os modelos que melhor acertam e não os que melhor aprendem. Todavia, a busca pelo modelo que melhor aprendeu ou generalizou ainda é uma tarefa difícil dentro do universo aprendizado de máquina. Diante de questões como essas, estudos recentes buscaram conhecimento em outras áreas para preencher essa lacuna. Uma nova abordagem é a aplicação de conceitos psicométricos que são comumente utilizadas para avaliar o aprendizado de indivíduos, entre eles está a Teoria de Resposta ao Item (TRI) [Cardoso et al. 2020].

A TRI é um conjunto de modelos matemáticos que visa calcular a probabilidade de um individuo acertar corretamente um item de um teste, considerando a dificuldade do item e habilidade do indivíduo. Com uma analogia simples, é possível aplicarmos a TRI em aprendizado de máquina ao considerarmos os modelos como sendo os indivíduos, o dataset de teste como sendo o próprio teste e as instâncias como sendo os itens. Dessa forma, é possível avaliar qual foi o desempenho de um modelo considerando a própria complexidade das instâncias para mensurar com maior precisão qual a real habilidade do modelo. Neste artigo, são apresentado os resultados preliminares do desenvolvimento de uma metodologia de AutoML do tipo NAS baseada nos resultados obtidos pela TRI. A seção 2 demonstra a metodologia utilizada e a seção 3 trás os resultados preliminares.

2. Materiais e Métodos

Resumidamente, objetivo é utilizar a capacidade evolucionária dos algoritmos genéticos para criar um algoritmo de AutoML do tipo NAS, onde os indivíduos serão modelos de Redes Neurais sendo os genes os pesos da rede. A avaliação dos indivíduos será feita via o cálculo e conceitos da TRI, além disso os resultados da TRI também são utilizados para realizar a mutação adaptativa. A Tabela 1 resume como é o funcionando do AG proposto para AutoML.

Os indivíduos são uma lista de pesos que compõe a Rede Neural. Todas as redes são simples com apenas 3 camadas (entrada, oculta e saída). A quantidade de neurônios na rede são definidos de acordo com o dataset que se deseja classificar. Dentro do AG todos os genes são valores gerados entre 0 e 1. Para utilizar esses valores como pesos de uma Rede Neural, são calculados limites superior e inferior de cada camada da rede utilizando a inicialização de Xavier. Antes da execução do AG, o dataset é dividido em dados de treino e teste, de forma que somente os dados de treino são utilizados no AG. A Figura 1 apresenta o fluxograma da execução da metodologia proposta.

Como pode ser observado na Figura 1, nesse estudo é utilizado o modelo logístico

Tabela 1. Tabela de Parâmetros do AG.

Parâmetro	**Descrição**
Representação	Cada gene do cromossomo é um número tipo float
Cruzamento	Mutação Gaussiana
Probabilidade de Mutação	Mutação adaptativa pela TRI com queda exponencial
Seleção de Pais	Torneio
Seleção de Sobreviventes	Ficam só os filhos
Número de gerações máximo	Não foi determinado
Tamanho da população	20, 30 e 50
Inicialização	Aleatória
Critério de Parada	Parada por estagnação ou caso atinja um valor máximo
Fitness	O Fitness é calculado sobre a probabilidade de acerto da TRI

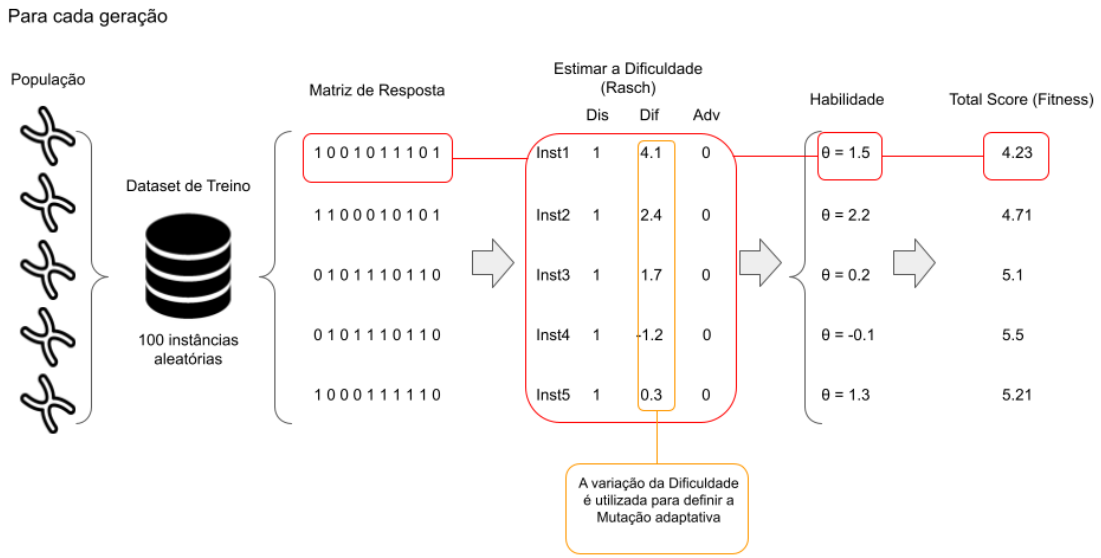

Figura 1. Fluxograma do AG baseado na TRI.

da TRI de um parâmetro, conhecido como modelo Rasch, onde apenas considera-se a dificuldade do item como parâmetro junto com a habilidade estimada do indivíduo. A cada geração do AG sorteia-se 100 instâncias aleatórias do conjunto de treino para testar os indivíduos, o objetivo é avaliar os modelos utilizando conjuntos diferentes de instâncias, que podem ser mais fáceis ou difíceis pela TRI e dessa forma escolher os modelos que melhor aprenderam/generalizaram para seguirem adiante. Cada modelo da população é colocado para classificar as instâncias sorteadas e depois as respostas são utilizadas para calcular a dificuldade e a habilidade estimada das Redes Neurais. Por conseguinte, calcula-se a probabilidade de acerto para cada instância de treino e então obtém-se o Score total que é a soma da probabilidade de acerto sempre que o modelo acertar e a subtração da probabilidade de erro sempre que o modelo errar, sendo 100 o valor máximo. Dessa forma, busca-se tornar mais evidente a separação de modelos mais habilidosos dos menos habilidosos e também para não classificar o indivíduo somente pela quantidade de acertos, mas

também pela qualidade dos acertos. Além disso, pode-se usar os níveis de dificuldade obtidos para entender como está o desempenho total da população. Para isso, a cada geração é calculado a média da dificuldade normalizada entre 0 e 4, então verifica-se se essa média aumentou ou diminuiu. Caso tenha aumentado, então incrementa-se a probabilidade de mutação e o sigma, caso contrário continua o decaimento exponencial.

3. Resultados Preliminares

Como estudo de caso, foi escolhido o dataset Banknote-Atuthentication, obtido na plataforma OpenML. Que foi escolhido por se tratar de um dataset mais simples com somente 4 features e que serve para testar a metodologia proposta. O dataset foi então divido em 70% para treino e 30% para teste, o AG então é executado 5 vezes para cada população, a Tabela 2 apresenta os valores finais de média de fitness obtidos.

Tabela 2. Resultados obtidos pelo AG para o dataset Banknote-Atuthentication.

População	Média do Fitness	Desvio Padrão do Fitness
20	97.2	1.6
30	96.8	1.6
50	97.22	0.95

Para comparar o desempenho da metodologia, as 5 melhores Redes obtidas pelo AG foram comparadas com o desempenho de 5 modelos gerado pelo AutoKeras, um sistema consolidado para NAS. Para isso, os modelos gerados foram avaliados utilizando o dataset de teste e obtida a média das acurácias. Na comparação direta o AutoKeras foi o vencedor, pois obteve uma média 0.993 de acurácia enquanto a metodologia proposta obteve 0.977. Apesar de não alcançar o mesmo nível de acerto, nota-se que a metodologia proposta mantêm-se competitiva com desempenho muito próximo. Entretanto, a melhor rede gerada pelo AutoKeras possui o total de 1258 parâmetros, enquanto a melhor rede obtida pelo AG possui apenas 69 parâmetros que corresponde a apenas 5.5% do total de parâmetros do modelo do AutoKeras. Portanto, o modelo gerado pela metodologia proposta conseguiu alcançar um desempenho médio quase ótimo e com uma Rede Neural muito mais simples e transparente, evidenciando que há futuro na exploração do uso da TRI para geração de modelos via AutoML.

Referências

Cardoso, L. F., Santos, V. C., Francês, R. S. K., Prudêncio, R. B., and Alves, R. C. (2020). Decoding machine learning benchmarks. In *Brazilian Conference on Intelligent Systems*, pages 412–425. Springer.

D'Amour, A., Heller, K., Moldovan, D., Adlam, B., Alipanahi, B., Beutel, A., Chen, C., Deaton, J., Eisenstein, J., Hoffman, M. D., et al. (2020). Underspecification presents challenges for credibility in modern machine learning. *arXiv preprint arXiv:2011.03395*, 1(3).

He, X., Zhao, K., and Chu, X. (2021). Automl: A survey of the state-of-the-art. *Knowledge-Based Systems*, 212:106622.

Ren, P., Xiao, Y., Chang, X., Huang, P.-Y., Li, Z., Chen, X., and Wang, X. (2021). A comprehensive survey of neural architecture search: Challenges and solutions. *ACM Computing Surveys (CSUR)*, 54(4):1–34.