

Informação e criatividade computacional

Ricardo Ribeiro Gudwin

Como citar: GUDWIN, Ricardo Ribeiro. Informação e criatividade computacional. *In:* ALMEIDA, Carlos Cândido de; VITTI-RODRIGUES, Mariana (org.). **Estudos pluridisciplinares da informação:** filosofia, tecnologia e semiótica. Marília: Oficina Universitária; São Paulo: Cultura Acadêmica, 2025. p. 255-281. DOI: <https://doi.org/10.36311/2025.978-65-5954-633-6.p255-281>.



All the contents of this work, except where otherwise noted, is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 (CC BY-NC-ND 4.0).

Todo o conteúdo deste trabalho, exceto quando houver ressalva, é publicado sob a licença Creative Commons Atribuição-NãoComercial-SemDerivações 4.0 (CC BY-NC-ND 4.0).

Todo el contenido de esta obra, excepto donde se indique lo contrario, está bajo licencia de la licencia Creative Commons Reconocimiento-No comercial-Sin derivados 4.0 (CC BY-NC-ND 4.0).

9

INFORMAÇÃO E CRIATIVIDADE COMPUTACIONAL

INFORMATION AND COMPUTATIONAL CREATIVITY

Ricardo Ribeiro GUDWIN

UNICAMP

gudwin@unicamp.br

<https://orcid.org/0000-0002-2189-1055>

Resumo: Existe um debate perene na filosofia, questionando se máquinas podem ser criativas, ou se podem somente se limitar a reproduzir tarefas para as quais foram programadas, sem exercer nenhum poder criativo. Neste trabalho, fazemos uma análise dos fundamentos teóricos por trás de processos criativos, sob a perspectiva da teoria da informação e da computação, e também da filosofia Peirceana.

Palavras-chave: informação, criatividade computacional, semiótica, processos criativos.

Abstract: There is a permanent philosophical debate about whether machines can be creative, or whether they can simply reproduce the tasks they have been programed to perform without exerting any creative power. In this chapter, we analyze the theoretical foundations underlying creative processes from the perspective of information and computing theory, as well as Peircean philosophy.

Keywords: information, computational creativity, semiotics, creative processes.

1 INTRODUÇÃO

Sistemas computacionais são máquinas capazes de seguir instruções, previamente armazenadas em programas disponibilizados em sua memória. Tipicamente, um programa de computador é executado sempre da mesma maneira, sendo que para um dado conjunto de entradas, as saídas devem sempre ser as mesmas, uma vez que o conjunto de instruções, em princípio não muda. Apesar de ser possível criar programas que modificam a si mesmos, fazendo com que seu comportamento se modifique durante sua própria execução, a grande maioria dos programas não possui essa característica, sendo que o código executado permanece o mesmo. Assumindo essa premissa, é razoável nos perguntarmos se programas desse tipo (que não se auto-modificam) podem ser criativos, ou exercer atividade criativa. De acordo com (Runco; Jaeger, 2012), o conceito de criatividade advém da combinação de dois fatores: *originalidade* (também muitas vezes chamado de novidade) e *efetividade*, que está relacionado ao valor daquilo que é criado. Em muitas definições de criatividade, essa efetividade está relacionada à *utilidade* daquilo que é gerado. Algumas vezes, a palavra *adequabilidade* (*fitness*) a um determinado propósito ou situação é também empregada. O termo *efetividade* se mostra talvez mais interessante, pois permite uma abrangência mais ampla do que *utilidade*, ou *adequabilidade*. Isso aconteceria, por exemplo, na geração de obras de arte, que podem possuir um valor efetivo, sem possuir uma utilidade evidente ou uma adequabilidade a um critério canônico. Além desses dois fatores, Boden (1998) acrescenta um terceiro fator, que seria o *fator surpresa*, ou seja, o fato de que aquilo que foi criado seja *surpreendente*, *inesperado* ou *imprevisível*. Este terceiro fator está menos ligado à própria criatividade em si, mas mais à questão da intensidade, ou do nível de criatividade. Assumindo que a criatividade é uma capacidade que as pessoas (e talvez os sistemas) possam ter em diferentes níveis, altos níveis de criatividade estão ligados à capacidade de surpreender com suas criações. Isso não significa que o produto da ação criativa necessite desse fator surpresa, mas que, em havendo esse fator surpresa, ele implica na consideração de que o agente criador é bastante criativo.

Levando-se em conta a definição de criatividade de Runco e Jaeger (2012) (desconsiderando, em princípio, o fator surpresa como uma necessidade para a atividade criativa, mas aquiescendo que sua manifestação pode implicar em altos níveis de criatividade), para que um processo computacional seja considerado criativo, ele precisaria ser capaz de duas coisas: gerar estruturas originais e estimar o valor daquilo que foi gerado. Vamos analisar cada uma dessas pré-condições. Em primeiro lugar a questão da originalidade: Boden (2005) nos alerta que se quisermos nos ater à definição de dicionário para originalidade¹, isso seria algo impossível. Em princípio, tudo o que é original, gerado pela mente humana, acaba tendo alguma influência em elementos previamente assimilados pela percepção. Da mesma forma, a própria Boden (1998) nos exorta a diferenciar entre o que ela chama de P-criatividade (ou criatividade *personal*, ao nível de indivíduo) e H-criatividade (ou criatividade histórica). A P-criatividade envolve uma originalidade que ocorre somente ao nível do indivíduo, algo que é novo para ele, mas que pode ter sido já criado por outros indivíduos na história da humanidade. Para o próprio indivíduo, seria uma atividade criativa. Já do ponto de vista histórico, somente valeria a H-criatividade, ou seja, a geração de algo que nunca foi gerado anteriormente na história da humanidade. Apesar de um conceito interessante, esse conceito de H-criatividade é bastante criticável. O que significaria algo completamente original, do ponto de vista da H-criatividade? Vamos pensar em uma nova teoria científica, por exemplo. Será que quando uma nova teoria científica é publicada em um artigo científico, em um periódico de amplo conhecimento, podemos dizer que ela possui H-criatividade? Quem nos garante que essa mesma teoria não poderia ter circulado anteriormente em um círculo restrito de pessoas, talvez transmitida somente verbalmente, como um comentário momentâneo em uma conversa particular, da qual não nos lembramos com exatidão o autor? Boden (1998) também nos sugere os meios pelos quais essa geratividade pode ocorrer. Ela sugere que existem três meios principais que podem dar origem a novas estruturas (ou ideias): a *combinação* de elementos anteriores, gerando algo novo, a *exploração* de um espaço conceitual previamente delimitado, e a *transformação*

¹ ‘to bring into being or form out of nothing’, ou seja, gerar algo a partir do nada.

de uma ou mais dimensões desse espaço conceitual, levando a abstrações de um novo tipo, que em certos casos demanda toda uma reestruturação do conhecimento obtido até então. Ela chama a essas criatividades de criatividade combinacional, criatividade exploratória e criatividade transformacional. Nas seções subsequentes, vamos analisar um pouco mais profundamente essa questão da originalidade e suas possibilidades em termos computacionais.

Mas, como também nos exorta Boden (2005), não necessariamente alguma coisa original pode ser considerada criativa. Diversas coisas extremamente originais são completamente inúteis, e portanto, despidas de valor e, sendo assim, de criatividade. O segundo requisito para a criatividade é a necessidade de que aquilo que foi gerado tenha valor, ou seja, que tenha *efetividade*. Talvez esse seja o requisito mais importante para que máquinas, dispositivos ou algoritmos computacionais sejam considerados criativos. A grande dificuldade em estimar o valor daquilo que é gerado é o fato de que nem mesmo seres humanos são plenamente capazes de fazer essa estimativa, de maneira efetiva. O valor de alguma coisa é algo extremamente subjetivo, podendo variar de pessoa para pessoa, ou até mesmo dependendo de circunstâncias momentâneas em que cada pessoa se encontra. Com isso, qualquer estimativa do valor de alguma coisa original que tenha sido gerada está sujeita a imprecisões, e está limitada aos critérios utilizados na estimação desse valor, que podem ser de maior ou menor qualidade. De uma certa forma, assumindo que algoritmos computacionais possam exibir criatividade, já temos aqui uma restrição: a qualidade da capacidade criativa desses algoritmos será maior ou menor, dependendo da qualidade na estimativa do valor daquilo que é gerado.

Alguns algoritmos computacionais (programas de computador), como os algoritmos genéticos (Dennis; Stella, 2011; Vega *et al.*, 2013), parecem ser capazes de satisfazer essas duas condições para que sejam criativos: originalidade e efetividade. A originalidade é conseguida às custas dos assim chamados algoritmos geradores de números pseudoaleatórios, além dos operadores de *crossover* e mutação, que simulam um processo evolutivo em estruturas computacionais. A efetividade é conseguida a partir das assim chamadas funções de *fitness*, que avaliam a qualidade das estruturas

geradas por meio dos operadores de *crossover* e mutação. Da mesma forma, algoritmos mais recentes, tais como as redes neurais dos tipos Autoencoder, redes gerativas adversariais (GANs), LSTM, redes de convolução, modelos de difusão (Franceschelli; Musolesi, 2021), os Transformers (Franceschelli; Musolesi 2021; Wertz; Kuhn, 2022), ferramentas para síntese de imagens hiper-realísticas, como o MidJourney e o DALL-E (Akyıldız, 2023; Borji, 2022; Hanna, 2023; Jaruga-Rozdolska, 2022; Pearson, 2023; Radhakrishnan, 2023) e até mesmo os LLMs (Large Language Models) como o ChatGPT, o Llama 2, o Google Bard ou o Claude 2 (Franceschelli; Musolesi 2023; Chakrabarty *et al.*, 2023; Chakrabarty *et al.* 2023; Zhang, 2023) parecem também exibir características de criatividade.

Nas seções a seguir, faremos uma análise mais detalhada de como tais algoritmos, como processos computacionais de transformação de informação, podem desenvolver criatividade. Para isso, nos serviremos de ideias já apresentadas nesta introdução, referentes ao trabalho de Boden (1998), reinterpretadas segundo a teoria da semiótica de Charles Sanders Peirce.

2 PRINCÍPIOS DA TEORIA PEIRCEANA

O filósofo norte-americano Charles Sanders Peirce (1839-1914) é hoje considerado como um dos maiores filósofos americanos, tendo desenvolvido uma teoria ao mesmo tempo inovadora e complexa que modela o pensamento humano. Sua teoria, também chamada de Semiótica, explica como o pensamento pode ser modelado em termos de signos, e como esses signos podem representar a realidade. Neste trabalho, vamos analisar a questão da criatividade computacional, utilizando como base alguns elementos da teoria Peirceana.

Diversos filósofos, dentro de uma tradição da filosofia chamada filosofia da mente, antes de Peirce, tentaram categorizar os diferentes tipos de pensamentos que habitam a mente humana. Por exemplo, Aristóteles (2021) entendia que os pensamentos podem ser enquadrados entre dez diferentes categorias: substâncias, quantidades, qualidades, relações, lugares, períodos de tempo, posições, estados, ações ou efeitos. Essas categorias, de acordo com Aristóteles (2021), se referiam aos tipos de palavras que

podem ser utilizadas para dizer coisas sobre o mundo. Já Kant (2024), desenvolveu uma teoria de categorias que englobava doze diferentes categorias, enquadradas em quatro grandes grupos: em termos de quantidades: Unidade, Pluralidade, Totalidade; em termos de qualidades: Realidade, Negação, Limitação, em termos de relações: Subsistência/Inerência, Causalidade/dependência, comunalidade (reciprocidade entre agente e paciente) e em termos de modalidade: Possibilidade/Impossibilidade, Existência/Inexistência, Necessidade/Contingência. As categorias de Kant, ao contrário de Aristóteles, dizem respeito aos tipos de juízos que se podem fazer sobre o mundo. Peirce (1931), tentando fazer uma síntese sobre as categorias de Kant, chegou à conclusão de que todos os pensamentos podem ser reduzidos e enquadrados em três grandes categorias: Primeiridade, Segundidade e Terceiridade. Essas categorias de Peirce estão relacionadas aos tipos de relações que podem ocorrer no mundo: relações monádicas, diádicas e triádicas. Dessa forma, os pensamentos, segundo Peirce (1931), funcionam de maneira análoga a compostos químicos, ou seja, possuem uma valência, que indica como esses compostos podem se coligar com outros compostos formando compostos mais elaborados.

As categorias de Peirce podem ser vistas como arranjos filosóficos, concepções derivadas da análise lógica do pensamento e presumidamente aplicáveis a todos os fenômenos do mundo (Peirce, 1931, CP 1.300)². Assim define Peirce suas três categorias (Peirce, 1931, CP 1.295, CP 1.296, CP 1.297, 1958, CP 8.328):

- **Primeiridade:** É tudo aquilo que é assim como é, ou seja, um primeiro, independente de um segundo ou terceiro.
- **Segundidade:** É tudo aquilo que é o que é, somente em relação a um segundo, mas de maneira independente de um terceiro (outridade).
- **Terceiridade:** É tudo aquilo que é o que é, em função de colocar um segundo e um terceiro, em relação um com o outro (composição).

² Citações para as obras de Peirce seguem o formato ABNT acrescido do formato tradicional usado por estudiosos peirceanos. Para parágrafos dos *Collected Papers of Charles Sanders Peirce*, a citação aparece como (Peirce, ano, CP x.y), onde “ano” indica o ano de publicação do volume, CP indica *Collected Papers*, x indica o volume e y indica o parágrafo – veja (Wikipedia contributors, 2023)

Segundo Peirce, a ideia de *Primeiro* é predominante nas ideias de aleatoriedade, novidade, criação, liberdade, originalidade, potencialidade (Peirce, 1931, CP 1.302). Dentre os exemplos fornecidos por Peirce como exemplos de primeiro estão as sensações e os sentimentos. A ideia de Segundo é predominante nas ideias de causação e reação (forças estáticas ocorrem sempre aos pares), comparação, oposição, polaridade, diferenciação, existência (oposição ao resto do mundo) (Peirce, 1931, CP 1.325). As ideias de segundo nascem da comparação entre percepção (sensação) e ação (vontade). A ideia de Terceiro é predominante nas ideias de mediação, meio, intermediário, continuidade, representação, generalidade, infinitude, difusão, crescimento, inteligência (intencionalidade) (Peirce, 1931, CP 1.340). É interessante, entretanto, nos exemplos de Peirce, que ele sempre fala em “predominância” de primeiridade, segundidade e terceiridade. Isso porque segundo ele, primeiridade, segundidade e terceiridade formam uma tecitura intrincada na constituição da realidade. Por exemplo, nas ideias de criação, liberdade e originalidade, a primeiridade é predominante. Mas isso não significa que não existam traços de segundidade ou de terceiridade, como veremos mais à frente.

De uma certa forma, Peirce em sua teoria faz uma mistura do assim chamado Tychismo (de Tyche – chance: acaso), a doutrina do acaso (ideia de que pode existir o acaso absoluto - nem tudo é fonte de um determinismo absoluto) com o assim chamado Sinequismo, a doutrina que prega que tudo que existe é contínuo. Segundo o sinequismo, o discreto é somente uma percepção simplificada de certos tipos de comportamento contínuo. Ou seja, o sinequismo é fundado na noção de que a coalescência, o tornar-se contínuo, o ser governado por leis não é nada mais do que fases de um único e mesmo processo de crescimento e desenvolvimento. Em outras palavras, no pensamento, ou seja, o fluxo das ideias na mente, as ideias somente podem ser conectadas por continuidade. Para Peirce (1931), o acaso é primeiridade (tudo aquilo que é assim como é, ou seja, um primeiro independente de um segundo ou terceiro) e o contínuo é a terceiridade (tudo aquilo que é o que é, em função de colocar um segundo e um terceiro, em relação um com o outro).

Um outro construto teórico importante, vindo de Peirce, é a distinção que ele faz entre Existência e Realidade. A Existência é um fenômeno de segundidade (Lane, 2024; Silveira, 2000). A distinção que Peirce faz entre Existência e Realidade é que as leis não fazem parte da Existência, embora operem sobre ela. Dessa forma, a Realidade corresponde à Existência, acrescentada das leis que operam sobre ela e do possível. Para Peirce (1931, 1934) as leis não existem, mas são reais. Cabe lembrar que, para Peirce (1931, 1934), leis são terceiridades, o que implica que os três elementos constituintes da realidade são: originalidade (primeiridade), obsistência (segundidade) e transuação (terceiridade). Obsistência tem a ver com o fluxo de mudanças observáveis na Existência, e a transuação com a identificação da operação de leis sobre esse fluxo, dando origem a seu entendimento.

3 PROCESSOS

Considerando que nosso objetivo é entender melhor o que seja o processo criativo, e avaliar se processos desse tipo podem ocorrer em dispositivos computacionais, é importante primeiramente compreendermos o que é um processo. O conceito de processo, entretanto, não é simples. A palavra processo é utilizada em diversos contextos diferentes, podendo muitas vezes ter um significado muito particular dentro de cada contexto. Por exemplo, em negócios e sistemas de produção, um processo é uma série de atividades realizadas por diferentes agentes, sobre um conjunto de matérias-primas, que acabam por resultar em um produto. No serviço público, um processo é uma pasta com documentos e anotações que registram o histórico de eventos relacionados a uma determinada pessoa ou empresa. Na computação, um processo é um programa de computador em execução em um dispositivo computacional. Na engenharia de software, um processo é um método padronizado para detectar os requisitos de usuários e promover o desenvolvimento de um programa ou sistema de computador, seguindo uma metodologia. No sistema legal, um processo é o registro histórico temporal dos eventos e decisões judiciais relacionados a uma demanda ou queixa encaminhada à justiça. Há, entretanto, uma noção pervasiva que colige os múltiplos significados da palavra processo, que tentaremos apreciar nessa seção.

Em sua versão mais abstrata, um processo pode ser entendido como um conjunto (ou coleção) ordenado de itens. Esse conjunto pode ser discreto (finito ou infinito) ou contínuo. Em processos discretos, temos uma sequência. Em processos contínuos, temos um fluxo. Existe uma correlação, que se caracteriza em todos os tipos de processos, entre o tempo e a ordenação dos itens no processo, em que cada item está relacionado a um instante de tempo. Esse tempo não precisa ser um tempo absoluto (embora muitas vezes o seja), mas pode ser simplesmente um tempo ordinal, que indica a ordem em que os itens aparecem no processo, ao longo do tempo. Para deixar a coisa um pouco mais complexa, a palavra processo é utilizada, de maneira intercambiável, tanto a processos particulares (instâncias), como a processos gerais (leis). Da mesma forma, a palavra processo pode se referir tanto ao conjunto completo, tomado como (um todo, quanto ao procedimento para a construção do conjunto, a partir de alguma função. Processos são tipicamente utilizados para representar as mudanças que ocorrem no mundo real, ao longo do tempo, em diferentes escopos.

Existe toda uma área de investigação, conhecida como filosofia dos processos, que busca estudar a natureza dos diferentes processos e como eles acontecem no mundo. A filosofia dos processos propõe uma ontologia da realidade onde os processos são os blocos elementares constituintes da realidade. Nessa visão, a realidade pode ser vista como uma grande teia de processos, conectados uns aos outros, constituindo aquilo que chamamos de Existência. Essa ontologia se coloca em oposição à assim chamada ontologia das substâncias, onde entidades, que sofrem mudanças no tempo, são os blocos básicos elementares. Os processos, como elementos básicos constituintes da realidade, são estudados por exemplo nos trabalhos de Whitehead (2010) e Rescher (1996).

No intuito de compreender melhor o que seria um processo criativo, vamos agora assumir que o conjunto ordenado que constitui um processo seja um conjunto numérico e analisar possíveis procedimentos para sua construção. Aplicando as categorias de Peirce a esse procedimento, poderíamos conceber três situações: cada novo elemento independe completamente dos elementos anteriores (primeiridade); cada novo elemento depende de maneira determinística (somente) dos elementos anteriores (se-

gundidade); cada novo elemento depende, de maneira determinística (função) dos elementos anteriores, e também de um alvo ou meta, para o qual desejamos que haja uma convergência na sequência de itens (terceiridade). Os processos construídos a partir desses procedimentos nos levariam, com certeza, a três diferentes tipos de sequências. O processo de primeiridade é claramente não-determinístico. Os processos de segundidade e terceiridade, como definidos aqui, são totalmente determinísticos. Entretanto, esses tipos não seriam os únicos possíveis que poderíamos conceber. Tanto os processos de segundidade como os de terceiridade, que são determinísticos, poderiam ter alguma influência não-determinística, simultânea à influência determinista em sua constituição. Poderíamos entender esses processos, portanto, como sofrendo uma degeneração na pureza de sua constituição. De fato, esses são os casos que Peirce chama de segundidade e terceiridade degeneradas, pois sofrem, de algum modo, influências dos outros modos, em nosso caso aqui, da primeiridade. Dessa forma, querendo analisar os casos em que poderíamos encontrar algum tipo de originalidade, podemos encontrar três opções: processos completamente aleatórios (primeiridade pura), processos transformativos não-determinísticos (segundidade degenerada, com influências de primeiridade), e processos combinacionais não-determinísticos (terceiridade degenerada, com influências de segundidade e primeiridade). É interessante confrontar aqui essa análise Peirceana dos processos e os três tipos de criatividade apontados por Boden (1998): criatividade exploratória, criatividade transformacional e criatividade combinacional, como faremos mais a frente, na seção 5.

4 GERAÇÃO DE NÚMEROS PSEUDOALEATÓRIOS

Um dos algoritmos mais requisitados na construção de simuladores computacionais é o assim chamado gerador de números pseudoaleatórios (Knuth, 2014). O propósito desse algoritmo é produzir sequências de números cujo comportamento seja muito difícil de distinguir de sequências “verdadeiramente aleatórias”, pelo menos para aplicações de interesse (Deng; Lin, 2000). Apesar desse algoritmo ser um algoritmo determinístico, diversas estratégias são utilizadas para tornar as sequências geradas indistinguíveis de uma sequência realmente aleatória. Algumas dessas es-

estratégias são melhores do que outras, quando critérios mais estritos são utilizados para julgar essa aleatoriedade. Bons geradores de números aleatórios podem ser difíceis de encontrar, quando as exigências são maiores (Hellekalek, 1998), como por exemplo em algoritmos de criptografia. Uma estratégia muito utilizada, principalmente em aplicações mais recentes (James; Moneta, 2020) são os algoritmos baseados em mapas caóticos (Naik; Singh, 2024). Esses algoritmos se baseiam em uma propriedade dos sistemas caóticos de serem virtualmente indistinguíveis de processos aleatórios, quando operando na zona de caos, apesar de serem determinísticos.

O determinismo de geradores de números pseudoaleatórios pode ser relativizado, utilizando-se como uma das entradas desses algoritmos um número que seja realmente aleatório. Muitos dos geradores de números pseudoaleatórios necessitam de uma (ou mais) semente(s), que são utilizadas para gerar os demais números da sequência. Um tipo de semente muito utilizada em simuladores é uma medida do instante inicial em que o programa é inicializado, num determinado uso do simulador. Assim, detecta-se o milissegundo exato em que o programa é inicializado, e esse número, que é de fato aleatório, é utilizado como semente para o gerador de números aleatórios. Com isso, cada vez que o programa seja executado, toda uma sequência diferente será gerada, a partir dessa semente aleatória, tornando a similaridade com uma sequência verdadeiramente aleatória ainda mais próxima.

Assim, assumindo que utilizamos um algoritmo gerador de números pseudoaleatórios, a partir de uma semente verdadeiramente aleatória, podemos ter um gerador de originalidade que será muito útil na construção de criatividade computacional. Na falta de um gerador de números aleatórios verdadeiro, os geradores de números pseudoaleatórios podem ser vistos como os motores de primeiridade utilizados em sistemas de criatividade computacional.

5 PROCESSOS GERADORES DE ORIGINALIDADE

Em computadores, toda informação fica armazenada em estruturas de dados. Essas estruturas podem ser de diversas formas, tais como tuplas,

vetores, matrizes, tensores, árvores, grafos, etc. Essas estruturas são utilizadas para representar informações dos mais variados tipos, tais como dados de sensores, atuadores e estados internos de agentes, imagens gráficas, e até mesmo estruturas 3D como objetos. Tipicamente, quando pensamos em um processo criativo, gerado por computadores, estamos falando da criação de uma estrutura de dados como essas já mencionadas que seja, ao mesmo tempo, original e efetiva, ou seja, que tenha algum valor. Analisando a questão sob a perspectiva das categorias Peirceanas, estamos falando de um processo de primeiridade.

Vamos agora retomar a proposta de Boden (1998), de que existem três mecanismos geradores de originalidade, mas vamos fazê-lo agora sob uma perspectiva da semiótica Peirceana. Uma das características do pensamento Peirceano é aplicar de maneira recursiva os conceitos de primeiridade, segundidade e terceiridade, a si próprios. Assim, apesar de estarmos interessados em mecanismos geradores de originalidade, vamos categorizá-los como mecanismos com características de primeiridade, segundidade e terceiridade. O primeiro candidato para um processo gerador de originalidade, com características de primeiridade, é o próprio gerador de números pseudoaleatórios. Por meio deste processo, podemos gerar estruturas de dados completamente originais, simplesmente utilizando o processo de geração de números pseudoaleatórios para gerar cada item de uma estrutura de dados que queremos criar. Vamos chamar esse processo de um processo de **geração**. Ele é um processo de primeiridade, porque segundo Peirce, uma primeiridade é “tudo aquilo que é assim como é, ou seja, um primeiro, independente de um segundo ou terceiro” (Peirce, 1958, p. 221, CP 8.328, tradução nossa). O processo de geração, entretanto, não nos leva muito longe, pois as estruturas de dados que podem ser geradas vão se assemelhar, no máximo, a um ruído branco. A probabilidade de se conseguir gerar uma estrutura efetiva, necessária para um processo criativo, é muito baixa, utilizando-se somente processos de geração. Para conseguir estruturas originais e efetivas, faremos uso de um segundo processo, onde também utilizamos o gerador de números pseudoaleatórios, mas fazendo uso de uma outra estrutura de dados. Com isso, temos um exemplo de um processo de segundidade, pois segundo Peirce, uma segundidade

é “tudo aquilo que é o que é, somente em relação a um segundo, mas de maneira independente de um terceiro” (Peirce, 1958, p. 221, CP 8.328, tradução nossa). A esse tipo de processo, daremos o nome de um processo de **transformação**. Em um processo de transformação, tomamos como base um conjunto de estruturas de dados já existentes (que podem ser de qualquer tipo: imagens, grafos, parâmetros de funções, etc.), e primeiramente utilizamos o gerador de números pseudoaleatórios para escolher uma estrutura que será transformada. Em seguida, aplicamos novamente o gerador de números pseudoaleatórios para escolher um conjunto de partes da estrutura original que serão modificadas. E por fim, utilizamos novamente o gerador de números pseudoaleatórios, agora para substituir a parte modificada, utilizando um número pseudoaleatório gerado pelo gerador de números pseudoaleatórios. Há várias maneiras diferentes de criar processos de transformação. Em todas elas, entretanto, sempre usamos o gerador de números pseudoaleatórios para funcionar como motor de primeiridade. Por exemplo, quando vamos selecionar qual estrutura de dados vamos escolher para ser modificada, podemos escolher essa estrutura de maneira completamente aleatória, ou utilizar algum viés que nos indique a efetividade que queremos obter com o processo criativo. Assim, podemos dar maior probabilidade de escolha para estruturas que sejam mais efetivas. Entretanto, sempre haverá alguma aleatoriedade envolvida. Da mesma forma, quando vamos escolher a parte da estrutura original que será modificada, podemos também utilizar (ou não) algum viés que nos permita escolher as partes que sejam mais promissoras para gerar estruturas mais efetivas. E por fim, também na substituição da parte da estrutura que será inserida, podemos ou não utilizar a parte original, que pode por exemplo, ser utilizada em uma operação com os números pseudoaleatórios gerados para compor a nova parte da estrutura. Vemos com isso, que há várias maneiras diferentes de gerar processos de transformação.

Apesar de podermos, em um primeiro momento nos darmos como satisfeitos com processos de **geração** e processos de **transformação**, podemos ainda nos valer do conceito de terceiridade, e com isso conceber um terceiro tipo de processo gerador de originalidade: um processo que

chamaremos de processo de **combinação**. Esse tipo de processo fará uso de uma terceiridade, que para Peirce é “tudo aquilo que é o que é, em função de colocar um segundo e um terceiro, em relação um com o outro” (Peirce, 1958, p. 221, CP 8.328, tradução nossa). Um processo de combinação guarda muitas semelhanças com um processo de transformação, mas ao invés de utilizar somente uma estrutura original como base para a geração de novas estruturas, faz uso de estruturas terceiras. Para um processo de combinação, não é necessário que usemos somente duas estruturas originais. Podemos fazer uso de um número qualquer de estruturas originais (até mesmo um número aleatório). Um processo de combinação é constituído das seguintes etapas: da mesma maneira que no processo de transformação, utiliza-se um conjunto de estruturas de dados já estabelecidas (por exemplo, um banco de imagens, um conjunto de parâmetros para funções, etc.), e na sequência utiliza-se um gerador de números pseudoaleatórios para decidir um subconjunto desse conjunto original, que deverá compor o conjunto de estruturas que serão usadas no processo de combinação. Essa seleção das estruturas originais pode (ou não) envolver algum tipo de métrica que defina a efetividade, ou qualidade da estrutura original, visando aqui não somente gerar originalidade, mas gerar criatividade (lembrando que para haver criatividade, é necessário que aquilo que é gerado seja não somente original, mas também efetivo). Em seguida, utiliza-se novamente o gerador de números pseudoaleatórios, para decidir quais partes das estruturas originais serão utilizadas na composição da nova estrutura que será gerada. Por fim, utiliza-se novamente o gerador de números pseudoaleatórios, para gerar as novas subestruturas que serão inseridas na nova estrutura. De novo, da mesma maneira que no caso dos processos de transformação, no processo de combinação, há várias maneiras diferentes que podem ser concebidas. Pode-se utilizar números pseudoaleatórios para decidir quanto de cada estrutura do banco de estruturas inicial será utilizado, bem como o grau de modificação dessas subestruturas, e pode-se utilizar algum tipo de métrica que avalie o grau de efetividade da modificação. Assim, diversos tipos de processos de combinação podem ser concebidos.

Desta maneira, podemos conceber, do ponto de vista computacional, três tipos de processos geradores de originalidade:

- Processos de **Geração** (primeiridade)
- Processos de **Transformação** (segundidade)
- Processos de **Combinação** (terceiridade)

É importante salientarmos que, apesar dos três processos serem processos de primeiridade, cada um deles o faz exibindo características distintas de primeiridade, segundidade e terceiridade.

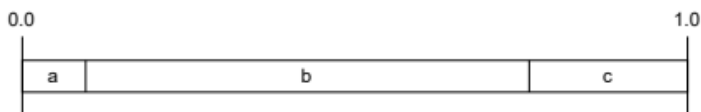
Um bom observador perceberá nesses processos de geração uma grande semelhança com operadores utilizados em algoritmos genéticos ou algoritmos evolutivos. O processo de transformação se assemelha muitíssimo com os processos de mutação, e os processos de combinação se assemelham muitíssimo com as operações de *crossover*. De fato, uma operação de mutação não deixa de ser um processo de transformação, e uma operação de *crossover* não deixa de ser um processo de combinação. Aqui em nossa análise estamos apenas generalizando os tipos de processos geradores de originalidade para abstrair esses operadores e compreender melhor por que esses operadores são efetivos no que se propõem a realizar, fazendo uma análise semiótica de seu funcionamento.

6 PROCESSOS DE DECISÃO POR EFETIVIDADE

Na seção anterior, já percebemos que, dentre as etapas de descrição dos processos geradores de originalidade, podemos fazer uso do motor de primeiridade proporcionado pelo algoritmo gerador de números pseudoaleatórios para não somente gerar estruturas originais, mas estruturas efetivas. Para isso, como sugerimos já na seção anterior, é necessária uma **métrica de efetividade**. Essa métrica deve ser capaz de avaliar a efetividade de uma dada estrutura, dando um valor a ela que seja proporcional a sua qualidade, para os objetivos que pretendemos atingir com ela. Essa métrica funcionará como um mediador no processo de decisão, ou seja, a mesma agirá como um elemento de terceiridade (mediação) na tomada de decisão. Mas como é possível realizar essa decisão mediada? Vamos aqui analisar o algoritmo mais comum utilizado nos algoritmos genéticos ou evolutivos para tomada de de-

cisão: o algoritmo da roleta³. Todo processo de decisão envolve uma seleção dentre um conjunto de alternativas. Vamos assumir aqui que cada alternativa corresponde a uma estrutura computacional, que pode (ou não) ter sido construída utilizando-se um processo gerador de originalidade.

Figura 1: Exemplo do Algoritmo da Roleta



Fonte: Elaborado pelo autor.

Observemos a Figura 1, que nos apresenta um exemplo do uso do algoritmo da roleta. Nesse exemplo, temos 3 estruturas: *a*, *b* e *c*, em que o algoritmo precisa decidir por uma delas. A métrica de efetividade foi utilizada para reservar uma região, dentro do intervalo 0 a 1, proporcional à métrica atribuída a cada uma das estruturas. Vemos, portanto, que a métrica de efetividade é menor para a estrutura *a*, bem maior para a estrutura *b* e tem um valor intermediário para a estrutura *c*. Dessa forma, definimos as regiões do intervalo 0 a 1 para cada uma das estruturas que precisará ser selecionada. Em seguida, utilizamos o gerador de números pseudoaleatórios para nos gerar um número aleatório entre 0 e 1 (uma grande gama de algoritmos de geração de números pseudoaleatórios gera valores entre 0 e 1), e dependendo de qual região esse número cair, é feita a decisão pela estrutura computacional correspondente. Vemos que, por possuir uma região maior, a probabilidade da seleção da estrutura *b* é maior do que das estruturas *c* e *a*. Apesar disso, existe sempre uma possibilidade de estruturas com menor probabilidade serem selecionadas, pois existe um fator aleatório no processo. Com isso, o algoritmo da roleta pode fazer escolhas

³ Algum leitor pode estranhar que a forma linear que o intervalo [0,1] é representada na figura 1, seria incompatível com o termo “roleta”. O algoritmo tem sua origem em roletas usadas em cassinos em jogos de azar. Nestes aparelhos, existe de fato um disco giratório, e sobre esse disco estão entalhados sulcos onde podem se acomodar uma pequena esfera que é lançada sobre o disco em movimento, sendo que a esfera acaba por se acomodar em um dos sulcos, após cessado seu movimento, realizando o sorteio do número correspondente ao número que identifica o sulco onde a esfera se acomodou. O algoritmo da roleta é inspirado em roletas reais, abstraídas em sua forma algorítmica e portanto acaba por levar seu nome por conta dessa inspiração.

que, ao mesmo tempo que envolvem certa aleatoriedade, proporcionam um viés de qualidade, onde a probabilidade de se escolher estruturas com maior valor na métrica de efetividade são maiores.

Aliás, é interessante observarmos a relação de processos de decisão por efetividade com inteligência. Apesar de existirem diferentes entendimentos do que seja inteligência e, de fato, essa ser uma questão em aberto na área de ciências cognitivas, se buscarmos a compreensão etimológica da palavra inteligência (Klein, 2003), veremos que esse termo advém da justaposição de “inter” com “legere”, ou seja, “pegar do meio de”, que em outras palavras indica a capacidade de fazer seleções, de fazer boas escolhas. O próprio Peirce identifica a terceiridade como a categoria da inteligência, talvez imaginando o processo de mediação onde uma escolha é mediada pelo valor atribuído a cada alternativa. Em nosso caso (processos criativos) estamos não somente fazendo uso de valores como mediadores, mas promovendo um hibridismo entre originalidade (primeiridade) e inteligência (terceiridade), pois a proporcionalidade da região da roleta para cada alternativa dá mais peso para as alternativas com maior efetividade (inteligência), mas também a aleatoriedade do processo de decisão envolve a importância da originalidade (primeiridade), envidando a descoberta de algo que seja novo e efetivo ao mesmo tempo.

7 POSSÍVEIS QUESTIONAMENTOS E DISCUSSÃO

Seguindo nossa linha de raciocínio até aqui, podemos concluir que uma criatividade computacional não é somente uma aparência, ou seja, as diferentes evidências na geração de imagens hiper-realistas que são criadas em ferramentas como o MidJourney⁴, ou o DALL-E⁵, e os textos criados com LLMs como o ChatGPT⁶, o Gemini⁷, o Llama⁸ não são somente um artifício de prestidigitação computacional, gerando ilusões de criatividade, mas que é sim possível falarmos de uma criatividade computacional,

⁴ <https://www.midjourney.com/>

⁵ <https://openai.com/index/dall-e-3/>

⁶ <https://chatgpt.com/>

⁷ <https://gemini.google.com/>

⁸ <https://www.llama.com/>

que programas de computador podem gerar comportamento criativo. Apesar disso, alguns possíveis questionamentos poderiam ser levantados. A fragilidade mais aparente em nossa argumentação é que de uma certa forma, toda originalidade deriva, de um modo ou de outro, dos algoritmos de geração de números pseudoaleatórios. Alguém poderia questionar: se os números são chamados de pseudoaleatórios (em outras palavras, falsamente aleatórios), não seriam de fato números aleatórios. Com isso, toda essa geração seria determinística e, portanto, não haveria originalidade genuína. Nessa linha de raciocínio, entretanto, nos esquecemos de um ponto crucial: a semente utilizada nos algoritmos para iniciar o processo de cálculo. Essa semente é genuinamente aleatória, e tem a ver com o instante em que o programa é iniciado. É realmente notável como um único fator genuinamente aleatório pode ser o elemento gerador de aleatoriedade para toda a criatividade computacional. Mas definitivamente é isso que acontece. Uma única semente aleatória transforma um gerador de números pseudoaleatórios em um gerador de números aleatórios, e com isso temos o motor de primeiridade capaz de gerar originalidade em processos computacionais.

É necessário ressaltar, entretanto, que a coisa não é tão simples assim. É necessário compor, de diferentes maneiras, primeiridade, segundidade e terceiridade, para compreender os processos de geração, transformação e combinação, que geram originalidade, e depois utilizar os processos de decisão com efetividade, para logarmos de fato estruturas criativas, ou seja, que não somente são originais, como possuem efetividade, possuem valor, possuem qualidade.

Um fator que ainda não abordamos, mas que é crucial aqui é a qualidade dos processos criativos computacionais. Quando analisamos a criatividade em seres humanos, vemos que existem diferentes níveis, diferentes intensidades de criatividade. Existe desde a criatividade inventiva de seres humanos médios (ou normais, utilizando aqui a definição estatística de normalidade), e um espectro que começa com a baixa criatividade de indivíduos menos dotados e a genialidade dos grandes gênios da humanidade, capazes de invenções transformadoras da humanidade, seja nas artes, seja na ciência. Será que poderíamos observar o mesmo na criatividade computacional?

Aqui vamos nos aprofundar um pouco mais na questão da qualidade de nosso gerador de números pseudoaleatórios. Apesar de até aqui não termos levantado ainda essa questão, ela é crítica. Nem todos os geradores de números pseudoaleatórios possuem a mesma qualidade. Como nos alertam Hellekalek (1998) e James e Moneta (2020), bons geradores de números aleatórios não são fáceis de encontrar. Todo algoritmo gerador de números aleatórios possui suas vantagens e deficiências. Hellekalek (1998) discute diferentes critérios para encontrar os melhores algoritmos, segundo diferentes perspectivas e entender suas deficiências. James e Moneta (2020) nos trazem uma revisão dos algoritmos mais populares nos dias de hoje, junto com as principais metodologias que são utilizadas para garantir sua eficiência em aplicações críticas, onde qualquer viés na geração desses números pode ser comprometedora, como por exemplo, em aplicações na área de criptografia. Podemos então inferir que o uso de diferentes algoritmos geradores de números aleatórios em nossos processos criativos computacionais pode nos levar também a diferentes níveis ou intensidades de criatividade. Com isso, podemos inferir que a qualidade de um processo capaz de criatividade computacional pode ser influenciada pela qualidade do algoritmo gerador de números pseudoaleatórios utilizada como base para seu funcionamento.

Da mesma forma, outro fator crítico para o desenvolvimento de processos de criatividade computacional é a métrica de efetividade utilizada. Uma métrica de efetividade é uma função numérica que avalia uma dada estrutura computacional e atribui um valor a ela. A presunção para uma boa métrica de efetividade é que exista uma proporcionalidade entre os números gerados por essa função e a qualidade da estrutura computacional sob escrutínio. Embora, em princípio, a ideia seja completamente compreensível, encontrar boas métricas de efetividade também não é fácil. Não-linearidades na adoção de uma ou outra métrica de efetividade podem comprometer seriamente a qualidade na avaliação das estruturas computacionais e implicar em uma distribuição deficiente de regiões no algoritmo da roleta, comprometendo todo o processo de geração criativa.

A grande dúvida, que poderia comprometer uma atribuição de capacidade criativa a máquinas, seria a questão do fator surpresa. Se assu-

mirmos a definição de criatividade de Runco e Jaeger (2012), poderíamos concluir que processos computacionais podem ser criativos. Se aderirmos à noção de Boden (1998, 2005), de que o fator surpresa é também, junto com novidade e valor, condição necessária para a criatividade, então talvez essa capacidade criativa possa ser questionável. De uma certa forma, o fator surpresa depende muito da criatividade de quem avalia. Se uma estrutura computacional gerada por um processo computacional nos causar surpresa, ou seja, se for muito diferente do que esperaríamos observar, iremos atribuir criatividade ao responsável por sua geração. Se apesar de original e efetivo, essa criação não for muito diferente de algo que nós mesmos poderíamos gerar, talvez questionemos se o processo é mesmo criativo. Entretanto, o uso do fator surpresa como condição necessária torna uma definição de criatividade algo muito subjetivo. Talvez não deixe de ser criativo, mas seja somente menos criativo do que nós mesmos seríamos. Uma definição de criatividade como essa acaba sendo por demais antropocêntrica e dá margem a muitas discussões. Daria margem até mesmo a que certos seres humanos fossem considerados não criativos. De fato, dependendo dos juízes escolhidos para atribuir criatividade, um número diferente de seres humanos poderia (ou não) ser considerados criativos – aqueles com capacidade criativa inferior a dos juízes.

Esse embate vem sendo travado na literatura. Alguns autores, como por exemplo Chakrabarty *et al.* (2023) defendem que a criatividade atribuída a LLMs é falsa, e que esses algoritmos não podem ser considerados criativos, mesmo que suas criações sejam originais e de valor. No estudo que realizaram, analisaram textos criados por LLMs, versus textos criados por escritores profissionais, onde um conjunto de especialistas era convidado a avaliar a criatividade dos textos, acabando por concluir que os textos gerados pelos LLMs não eram criativos (talvez pudéssemos dizer, menos criativos, ao invés de não criativos). Chakrabarty *et al.* (2023) chegam a sugerir que a colaboração entre humanos e LLMs poderia levar ao desenvolvimento de textos mais criativos, mas sempre dentro dessa perspectiva antropocêntrica, que reserva a criatividade como uma capacidade exclusiva a seres humanos. Já Franceschelli e Musolesi (2023) são mais abertos. Entendem que LLMs poderiam exibir criatividade combinacional (Boden,

1998), mas defendem que os atuais LLMs certamente não são capazes de exibir criatividade transformacional (aquela que demandaria abstrações ou mudanças de perspectiva no enquadramento das coisas, segundo Boden). Mas não excluem a possibilidade de que futuros LLMs, mais elaborados e refinados, possam também vir a exibir esse tipo de criatividade.

Vamos tentar entender a questão um pouco melhor. Sabemos que os atuais LLMs não levam em consideração a questão semântica em seu processo gerativo. Os textos produzidos por LLMs levam em consideração somente a vizinhança entre palavras, em um repertório gigantesco de textos utilizados como amostras, durante sua fase de aprendizagem, para estabelecer uma correlação entre as distâncias entre diferentes palavras em textos de referência, para gerar, utilizando um viés aleatório, a próxima palavra de uma sentença que está construindo. Isso é feito de maneira recursiva, levando em consideração não somente a probabilidade de uma palavra em relação à palavra anterior, mas a toda uma sequência anterior, e as probabilidades em que isso já pode ou não ter ocorrido no passado. Essa estratégia extremamente simples acaba por gerar textos surpreendentemente sofisticados, como se tivessem sido escritas por uma pessoa, com atribuição de significado. Na verdade, não uma única pessoa, mas todas as pessoas que redigiram os textos originais, utilizados como amostras durante o treinamento do algoritmo. Mas a atribuição de significado, de fato, se dá somente por conta do leitor, não do algoritmo. O percentual de aleatoriedade inserido no procedimento é crítica. Uma aleatoriedade muito pequena pode levar à clara identificação dos fragmentos dos textos originais que foram utilizados na fase de aprendizagem. Uma aleatoriedade muito grande, pode levar ao fenômeno conhecido por “alucinação”. Ou seja, os textos gerados tornam-se por demais “criativos” (adjetivo colocado aqui de modo propositadamente provocativo), chegando, em alguns momentos, até a perder o sentido, caso a aleatoriedade torne-se alta demais. Esse mesmo procedimento utilizado nos LLMs é também utilizado nos processos de geração de imagens hiper-realistas. Só que nesse caso, ao invés da proximidade entre palavras, em textos utilizados como amostras, é medida a proximidade entre pixels de imagens utilizadas como referência. Mas

a estratégia é similar, só que bidimensional, ao invés de unidimensional, como no caso de textos.

Poderíamos nos questionar se esses algoritmos possuem compreensão (ou entendimento) daquilo que geram (e aqui nossa posição é a de que de fato, não possuem esse entendimento, mesmo os sistemas mais modernos de inteligência artificial, na data de hoje – o que não significa que no futuro não o possam ter). Mas a pergunta aqui é outra, e não convém misturar entendimento com criatividade. Afinal, um indivíduo *savant*⁹ pode ser considerado extremamente criativo, mesmo que não compreenda exatamente o que está fazendo. Dessa forma, nosso entendimento é que os processos computacionais podem ser criativos, mesmo que o façam sem um entendimento comparável ao entendimento humano sobre aquilo que produzem.

Nosso entendimento é que processos computacionais podem sim ser criativos. Entretanto, essa criatividade possui um nível, uma intensidade, que pode ser bastante inferior à de seres humanos, principalmente quando consideramos especialistas humanos em certas habilidades. Com relação ao fator surpresa, constatamos que o mesmo possui forte dependência com o nível de aleatoriedade utilizado nos processos de geração, transformação e combinação de estruturas computacionais que são utilizados nos algoritmos de inteligência artificial. Quando essa aleatoriedade é pequena demais, o resultado acaba sendo menos criativo, mais parecido com os fragmentos de componentes (palavras, píxeis ou outro elemento atômico de dados) utilizados durante a fase de treinamento dos algoritmos. Quando essa aleatoriedade cresce, aumenta a chance das estruturas geradas se tornarem surpreendentes, até chegar ao ponto em que começam a “alucinar” e deixar de gerar estruturas realistas, condizentes com as que seres humanos seriam capazes de gerar. No caso de textos, esses começam a perder a coerência semântica. Em caso de imagens, elas deixam de ser realistas. Dessa forma, até mesmo o fator surpresa poderia ser considerado, dentro das possíveis definições de criatividade que poderíamos utilizar para evidenciar nosso entendimento. Entretanto,

⁹ O savantismo é considerado um distúrbio psíquico com o qual a pessoa possui uma grande habilidade intelectual aliada a um déficit de outros tipos de inteligência. Tais habilidades são sempre ligadas a uma memória extraordinária, porém com pouca compreensão do que está sendo descrito.

estudos futuros ainda serão necessários para uma melhor compreensão de como essa dose de aleatoriedade afeta as estruturas de dados geradas, e se haveria uma dose ideal de aleatoriedade onde estruturas surpreendentes, originais e com valor possam ser criadas, atendendo até mesmo a definição de criatividade de Boden. Da mesma forma, seria necessário compreender melhor por que, acima dessa dose ideal de aleatoriedade, os resultados começam a degenerar e as estruturas geradas deixam de ser compatíveis com aquelas que seriam geradas por seres humanos.

8 CONCLUSÃO

Podemos concluir, de toda essa discussão que: (a) apesar dos algoritmos geradores de números pseudoaleatórios serem determinísticos, o uso de uma semente verdadeiramente aleatória os transforma em geradores de números aleatórios; (b) esse gerador de números aleatórios, como um processo computacional, funciona como um motor de primeiridade, capaz de gerar originalidade; (c) essa originalidade, composta com elementos de segundidade e terceiridade (trazidos pelo critério de efetividade, que funciona como um mediador nos processos de decisão e seleção), nos permite derivar os processos de geração, transformação e combinação, que seriam os elementos principais na construção de processos de criatividade computacional; (d) a capacidade criativa pode possuir diferentes níveis de intensidade, que estão relacionadas, de forma ainda não completamente conhecida, com a quantidade e onde a aleatoriedade é inserida em estruturas obtidas a partir da realidade.

Dessa forma, podemos concluir que processos geradores de criatividade computacional são, não somente viáveis, mas explicáveis segundo a perspectiva da semiótica de Peirce. Entretanto, da mesma forma que a criatividade humana é variável, de indivíduo para indivíduo, os processos geradores de criatividade computacional podem ter diferentes níveis de qualidade, dependendo da qualidade dos geradores de números aleatórios, das métricas de efetividade utilizados e da proporcionalidade entre a aleatoriedade empregada e a preservação das estruturas originais utilizadas nos processos de transformação e combinação.

REFERÊNCIAS

- ARISTÓTELES. *The Categories*. The Project Gutenberg: Salt Lake City, 2021. Disponível em: <https://www.gutenberg.org/ebooks/2412>. Acesso em: 15 jul. 2025.
- AKYILDIZ, E. C. Generative text-to-image models in architectural design: a study on relationship of language, architectural quality and creativity. *Icontech international journal of surveys, engineering, technology*, Izmit, v. 7, n. 3, p. 12-26, Sept. 2023. Disponível em: <https://icontechjournal.com/index.php/ij/article/view/304/108>. Acesso em: 25 set. 2024.
- BODEN, M. A. Creativity and artificial intelligence. *Artificial intelligence*, v. 103, n. 1-2, p. 347-356, Aug. 1998.
- BODEN, M. A. What is creativity?. In: MITHEN, S. (ed.). *Creativity in human evolution and prehistory*. London: Routledge, 2005. p. 27-55.
- BORJI, A. Generated faces in the wild: quantitative comparison of stable Diffusion, Midjourney and DALL-E 2. *arXiv*, New York, n. 2210.00586v1, p. 1-9, Oct. 2022. Disponível em: <https://arxiv.org/pdf/2210.00586v1/>. Acesso em: 25 set. 2024.
- CHAKRABARTY, T.; LABAN, P.; AGARWAL, D.; MURESAN, S.; WU, C. Art or artifice? large language models and the false promise of creativity. *arXiv*, New York, n. 2309.14556v1, p. 1-42, Sept. 2023. Disponível em: <https://arxiv.org/pdf/2309.14556v1/>. Acesso em: 25 set. 2024.
- CHAKRABARTY, T.; PADMAKUMAR, V.; BRAHMAN, F.; MURESAN, S. Creativity support in the age of large language models: an empirical study involving emerging writers. *arXiv*, New York, n. 2309.12570v1, p. 1-33, Sept. 2023. Disponível em: <https://arxiv.org/pdf/2309.12570v1/>. Acesso em: 25 set. 2024.
- DENG, L.; LIN, D. K. J. Random number generation for the new century. *American Statistician*, [s.l.], v. 54, n. 2, p. 145-150, May 2000. Disponível em: <https://doi.org/10.2307/2686034/>. Acesso em: 25 set. 2024.
- DENNIS, J. L.; STELLA, A. Teaching creativity: the case for/against genetic algorithms as a model of human creativity. *The open education journal*, v. 4, n. 1, p. 36-40, Nov. 2011. Disponível em: <https://benthamopenarchives.com/abstract.php?ArticleCode=TOEDUJ-4-36/>. Acesso em: 25 set. 2024.
- VEGA, F. F. de; NAVARRO, L.; CRUZ, C.; CHÁVEZ, F.; ESPADA, L.; HERNÁNDEZ, P.; GALLEGU, T. Unplugging evolutionary algorithms: on the sources of novelty and creativity. In: IEEE CONGRESS ON EVOLUTIONARY COMPUTATION, July 2013, Cancun, *Proceedings* [...]. Cancun: Institute of electrical and electronics engineers, 2013. p. 2856-2863. Disponível em: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6557916/>. Acesso em: 25 set. 2024.

FRANCESCHELLI, G.; MUSOLESI, M. Creativity and machine learning: a survey. *arXiv*, New York, n. 2104.02726v1, p. 1-25, Apr. 2021. Disponível em: <https://arxiv.org/pdf/2104.02726v1>. Acesso em: 25 set. 2024.

FRANCESCHELLI, G.; MUSOLESI, M. On the creativity of large language models. *arXiv*, New York, n. 2304.00008v1, p. 1-17, Mar. 2023. Disponível em: <https://arxiv.org/pdf/2304.00008v1/>. Acesso em: 25 set. 2024.

HANNA, D. M. The use of artificial intelligence art generator “Midjourney” in artistic and advertising creativity. *Journal of design sciences and applied arts*, Cairo, v. 4, n. 2, p. 42-58, June 2023. Disponível em: <https://dx.doi.org/10.21608/jdsaa.2023.169144.1231/>. Acesso em: 25 set. 2024.

HELLEKALEK, P. Good random number generators are (not so) easy to find. *Mathematics and computers in simulation*, Bruxelles, v. 46, n. 5-6, p. 485-505, June 1998. Disponível em: [https://doi.org/10.1016/S0378-4754\(98\)00078-0/](https://doi.org/10.1016/S0378-4754(98)00078-0/). Acesso em: 25 set. 2024.

JAMES, F.; MONETA, L. Review of high-quality random number generators. *Computing and software for big science*, [s.l.], v. 4, n. 2, p. 1-12, Jan. 2020. Disponível em: <https://doi.org/10.1007/s41781-019-0034-3/>. Acesso em: 25 set. 2024.

JARUGA-ROZDOLSKA, A. Artificial intelligence as part of future practices in the architect's work: MidJourney generative tool as part of a process of creating an architectural form. *Architectus*, Wrocław, v. 3, n. 71, p. 95-104, Nov. 2022. Disponível em: <http://doi.org/10.37190/arc220310/>. Acesso em: 25 set. 2024.

KANT, I. *Critique of pure reason*. London: Minerva Heritage Press, 2024.

KLEIN, E. *A comprehensive etymological dictionary of the English language: dealing with the origin of words and their sense development thus illustrating the history of civilization and culture*. [Amsterdam]: Elsevier, 2003.

KNUTH, D. E. Random numbers. In: KNUTH, D. E. *The art of computer programming: seminumerical algorithms*. [USA]: Addison-Wesley Professional, 2014. v. 2, p. 1-193.

LANE, R. Peirce on reality and existence. In: De Waal, C. (ed.). *The Oxford handbook of Charles S. Peirce*. Oxford: Oxford University Press, 2024.

NAIK, R. B.; SINGH, U. A review on applications of chaotic maps in pseudo-random number generators and encryption. *Annals of data science*, [s.l.], n. 1, p. 25-50, Feb. 2024. Disponível em: <http://doi.org/10.1007/s40745-021-00364-7/>. Acesso em: 25 set. 2024.

PEARSON, A. The rise of CreAltives: using AI to enable and speed up the creative process. *Journal of AI, robotics & workplace automation*, London, v. 2, n. 2, p. 101-114, Jan. 2023.

PEIRCE, C. S. *Collected papers of Charles Sanders Peirce*. Editado por Charles Hartshorne e Paul Weiss. Cambridge: Harvard University Press, 1931. v. 1-2.

PEIRCE, C. S. *Collected papers of Charles Sanders Peirce*. Editado por Arthur W. Burks. Cambridge: Harvard University Press, 1958. v. 7-8.

RADHAKRISHNAN, A. M. Is MidJourney-AI a new anti-hero of architectural imagery and creativity?: an atypical era of Ai-based representation & its effect on creativity in the architectural design process. *Global Scientific Journals*, v. 11, n. 1, Jan. 2023. Disponível em: http://www.globalscientificjournal.com/researchpaper/Is_MidJourney_AI_a_new_Anti_hero_of_Architectural_Imagery_and_Creativity_.pdf/. Acesso em: 25 set. 2024.

RESCHER, N. *Process metaphysics: an introduction to process philosophy*. [New York]: Suny Press, 1996.

RUNCO, M. A.; JAEGER, G. J. The standard definition of creativity. *Creativity research journal*, Philadelphia, v. 24, n. 1, p. 92-96, Feb. 2012. Disponível em: <https://doi.org/10.1080/10400419.2012.650092/>. Acesso em: 25 set. 2024.

SILVEIRA, L. F. B. Acaso, existência e lei num universo em evolução. *Cognitio*, São Paulo, v. 1, n. 1, p. 117-137, jul./dez. 2000.

WERTZ, L.; KUHN, J. Adapting transformer language models for application in computational creativity: generating German theater plays with varied topics. In: INTERNATIONAL CONFERENCE ON COMPUTATIONAL CREATIVITY, 13., 2022, Bolzano. *Proceedings* [...]. Bolzano: Association for computational creativity, June/ July 2022. p. 17-25.

WHITEHEAD, A. N. *Process and reality*. [New York]: Simon and Schuster, 2010.

WIKIPEDIA CONTRIBUTORS. *Charles Sanders Peirce bibliography*. In: WIKIPEDIA: the free encyclopedia. Disponível em: https://en.wikipedia.org/w/index.php?title=Charles_Sanders_Peirce_bibliography&oldid=1187258778. Acesso em: 31 jan. 2024.

ZHANG, C. User-controlled knowledge fusion in large language models: balancing creativity and hallucination. *arXiv*, New York, n. 2307.16139, p. 1-7, Aug. 2023. Disponível em: <https://arxiv.org/pdf/2307.16139/>. Acesso em: 19 Jan. 2024.