Created by:

MORPHUS
Part of Accenture

Segurança em Inteligência Artificial Generativa:

Desafios, Riscos e Medidas Preventivas

Laboratório de Segurança Cibernética

FEBRABAN FEDERAÇÃO BRASILEIRA DE BANCOS

SUMÁRIO

FEBRABAN FEDERAÇÃO BRASILEIRA DE BANCOS

1 INTRODUÇÃO

Este relatório tem como objetivo principal discutir os desafios de cibersegurança enfrentados pelas organizações ao adotarem a Inteligência Artificial Generativa — *Generative Artificial Intelligence* (GenAI) — em seus processos cotidianos. Buscamos fornecer informações valiosas que possam orientar as organizações na construção de estratégias confiáveis para o uso da Inteligência Artificial — *Artificial Intelligence* (AI) —, seja no desenvolvimento, implantação ou no monitoramento contínuo dessas soluções.

O diferencial deste trabalho é a adoção de uma abordagem de análise dos riscos com foco nas ameaças e vulnerabilidades exploradas por atores maliciosos, bem como nos padrões de ataques utilizados para explorar essas falhas de segurança. A partir desta perspectiva, é possível estabelecer um conjunto de contramedidas que podem ser utilizadas para mitigar os riscos advindos da utilização de ferramentas de GenAI.

A abordagem deste tema se tornou imprescindível, dado o crescimento exponencial no uso de ferramentas de GenAI, que ultrapassou a comunidade da computação e alcançou a sociedade como um todo. Um exemplo disso foi o lançamento do ChatGPT [1], pela empresa OpenAI, em novembro de 2022. De fácil utilização e sendo gratuito para uso, o ChatGPT alcançou 1M de usuários em apenas 5 dias. Realização impressionante, visto que redes sociais como Twitter, Facebook e Instagram levaram meses para alcançar essa marca.

Atualmente, estamos em um ponto de inflexão para as organizações e a sociedade, ponto este corroborado pelas seguintes estatísticas [2][3]:

- 98% dos executivos globais concordam que a AI desempenhará um papel importante nas estratégias das organizações nos próximos 3 a 5 anos;
- 40% de todas as horas de trabalho serão impactadas pelo uso de ferramentas de GenAI como é o caso do ChatGPT;
- 65% deste tempo será mais bem aproveitado por meio de inúmeras tarefas de automação, o que irá refletir em uma maior produtividade dos funcionários;
- 20% dos líderes disseram que a inteligência artificial e o aprendizado de máquina terão uma maior influência em suas estratégias de risco cibernético ao longo dos próximos dois anos;
- 76% das empresas esperam que a AI aumente o nível de risco de segurança significativamente ou moderadamente.

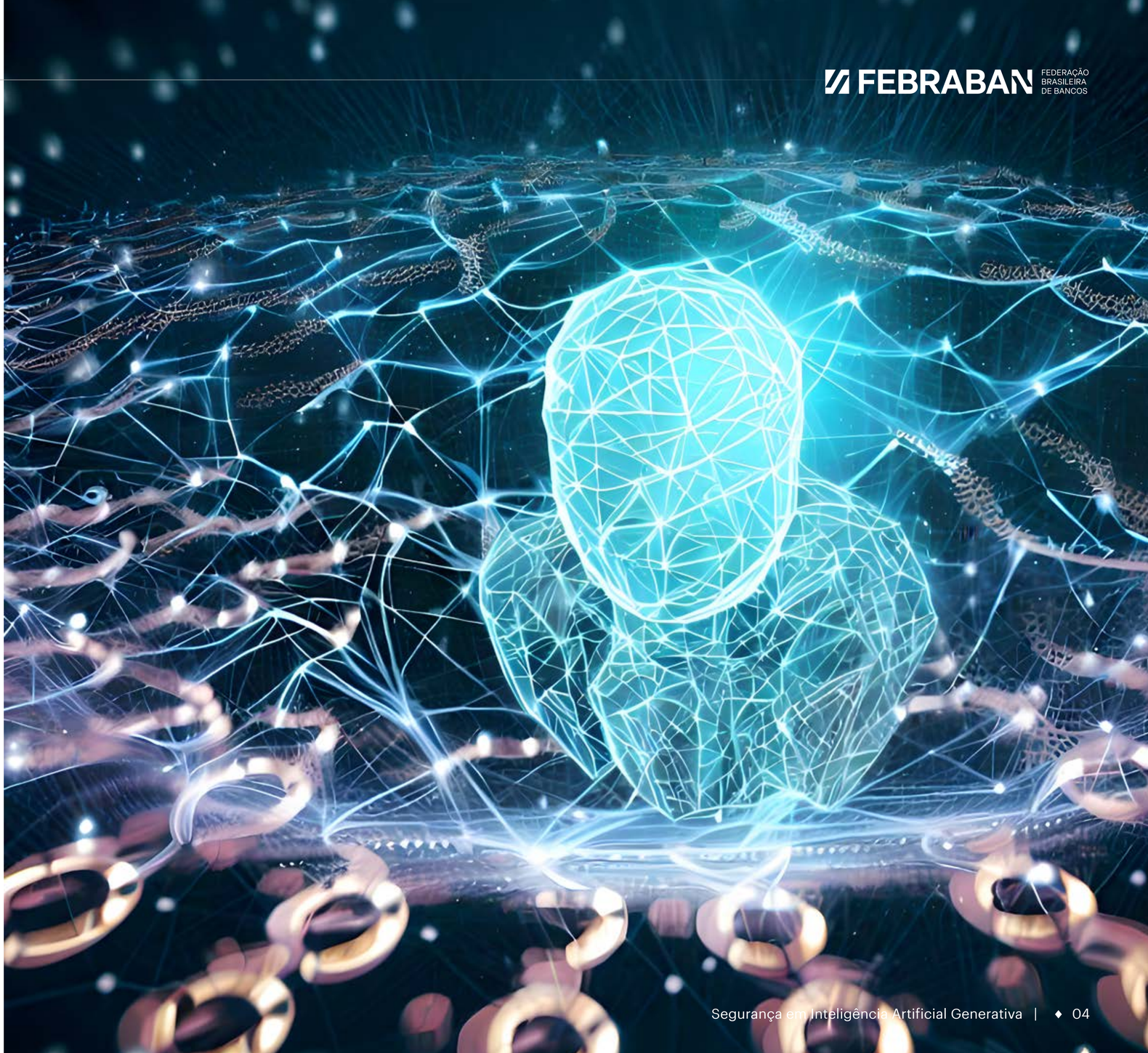
Em comparação com as tecnologias de AI tradicionais, as ferramentas que utilizam GenAI são muito mais fáceis de usar, graças a sua interface intuitiva e amigável que permite que os usuários editem os resultados em tempo real, muitas vezes, através de chats iterativos [4]. Além disso, a GenAI está sendo integrada em muitas aplicações de uso diário, como o pacote de ferramentas do Microsoft Office 365 e em plugins para as mais diversas funcionalidades nos navegadores de internet. Com isso, a GenAI está se tornando onipresente em nossas vidas, uma vez que está desempenhando papéis que vão desde a geração de documentos até a produção de peças artísticas.

Entretanto, apesar de todas as facilidades advindas do uso da GenAI, ferramentas que utilizam essa nova tecnologia podem apresentar grandes riscos a sociedade, entre eles: a geração automática de *malware*; criação de campanhas sofisticadas de *phishing* com o uso de texto, voz e vídeo (*deepfakes*) gerados por AI; ataques de desinformação com geração de conteúdos falsos; ataques contextuais de engenharia social; ampliação de preconceitos e discriminação existentes; etc.

Assim, fica claro que o uso dessas ferramentas é muito bem-vindo, visto que elas auxiliarão em inúmeras tarefas diárias, melhorando nossa produtividade. No entanto, dados os riscos de segurança, é necessário adotar estratégias que garantem que usuários mal-intencionados não possam explorar falhas e utilizar essas ferramentas

para objetivos criminosos. Por isso, é necessário aplicar medidas de controle de privacidade, políticas de acesso, auditoria e revisão dos resultados gerados pelas AI.

A estrutura do relatório segue a seguinte organização: o Capítulo 2 apresentará os conceitos fundamentais relacionados à inteligência artificial generativa, incluindo as aplicações dessa técnica, a definição de modelos base e as principais ferramentas presentes no mercado. No Capítulo 3, abordaremos os riscos à segurança dos sistemas de GenAI, explorando as potenciais ameaças, os tipos de vulnerabilidades e os padrões de ataques utilizados para explorá-los, bem como estratégias que podem ser adotadas como contramedidas para enfrentar as ameaças apresentadas. Por fim, no Capítulo 4, apresentamos as considerações finais que encerrarão o relatório.



2 INTELIGÊNCIA ARTIFICIAL GENERATIVA

A GenAI é o conjunto de algoritmos de AI projetados para criar dados e conteúdo de forma autônoma. Esses algoritmos têm a capacidade de gerar imagens, textos e áudio, que se assemelham aos dados de treinamento fornecidos a eles. Nas últimas décadas, avanços significativos foram alcançados em várias áreas, como visão computacional (conjunto de algoritmos especializados em compreender imagens digitais e vídeos), processamento de linguagem natural (algoritmos que possuem a capacidade de entender textos e a fala da mesma forma que os seres humanos) e áudio, impulsionando o desenvolvimento de modelos generativos sofisticados. A seguir, descreveremos o significado da AI Generativa, suas aplicações, o que são os modelos base e por fim, as principais ferramentas de GenAI.

2.1. - AI Discriminativa vs. Generativa

AI é um campo fascinante que busca replicar a capacidade humana de pensar e tomar decisões. Dentro desse campo, existem duas abordagens principais: a discriminativa e a generativa.
Para exemplicar, considere um cenário no qual existe um conjunto de imagens de gatos e cachorros. O modelo discriminativo aprende a mapear as características específicas de cada classe (e.g., gatos ou cachorros) e com isso, consegue identificar corretamente qual animal está presente em uma determinada imagem. AI discriminativa concentra-se nas diferenças observáveis entre os dois grupos e encontra uma fronteira de decisão que melhor separa as classes. Por outro lado, a GenAI tenta entender a estrutura dos dados, de modo a ser capaz de gerar novos exemplos que se assemelhem ao conjunto original. Em nosso exemplo, um modelo generativo tentaria aprender as características dos gatos e cachorros e, em seguida, seria capaz de gerar novas imagens de ambos os animais. Esse tipo de modelo é útil para tarefas como síntese de voz, criação de texto ou geração de imagens realistas.

Nos últimos anos, ferramentas como o ChatGPT [1], juntamente com o DALL-E [5], o Codex [6] e o Gopher [7], vêm ganhando significativa atenção da sociedade graças a capacidade de tornar o processo de criação de conteúdo mais eficiente e acessível. Essas ferramentas pertencem a uma categoria de algoritmos chamados de Conteúdo Gerado por Inteligência Artificial — *Artificial Intelligence Generated Content* (AIGC) —, que são técnicas avançadas de GenAI que permitem automatizar a criação de grandes quantidades de conteúdo em um curto período. O ChatGPT é um chatbot capaz de entender e responder de forma eficiente a entradas fornecidas pelos usuários através do uso da linguagem natural. Já o DALL-E é uma aplicação capaz de criar imagens únicas e de alta qualidade a partir de descrições textuais em poucos minutos.

Além dos benefícios trazidos pelo aumento do volume de dados e do poder computacional, pesquisadores também estão explorando formas de integrar novas tecnologias em conjunto com os algoritmos GenAI. Por exemplo, o ChatGPT utiliza Aprendizado por Reforço com *Feedback Humano* — *Reinforcement Learning with Human Feedback* (RLHF) — para determinar a resposta mais apropriada para uma determinada instrução, melhorando assim a confiabilidade e a precisão do modelo ao longo do tempo [19] [20] [21]. Enquanto isso, em visão computacional, a difusão estável [22], proposta pela Stability AI em 2022, também mostrou grande sucesso na geração de imagens. Ao contrário dos métodos anteriores, os modelos de difusão generativa podem ajudar a gerar imagens de alta resolução com diminuição do consumo de recursos.

Ao combinar esses avanços, os modelos de GenAI fizeram progressos significativos nas tarefas AIGC e vêm sendo adotados em vários setores, incluindo arte [23], publicidade [24] e educação [25]. Acreditamos que os algoritmos de AIGC continuarão a ser uma área de crescimento significativo e objeto de muitas pesquisas em inteligência artificial.

2.2. - Arquitetura Transformer

Tanto as aplicações mencionadas acima, assim como outras que utilizam GenAI, possuem como base um modelo linguístico pré-treinado conhecido como *Generative Pretrained Transformer* (GPT). Esse modelo linguístico é treinado em um conjunto enorme de dados de texto, o que os tornam adaptáveis a diferentes tarefas que envolvam o Processamento de Linguagem Natural — *Natural Language Processing* (NLP). Os *transformers* foram propostos pela primeira vez para resolver as limitações de modelos tradicionais de AI, como as Redes Neurais Recorrentes — *Recurrent Neural Networks* (RNNs) —, em lidar com sequências de entradas de tamanhos variáveis e de compreender o contexto do qual essas entradas fazem parte. Além disso, os *transformers* permitem que múltiplas sequências de entradas sejam avaliadas em paralelo, o que contribui para melhora do desempenho desses modelos [8].

A arquitetura do *transformer* é baseada principalmente em um mecanismo de autoatenção que permite ao modelo compreender as diferentes partes em uma sequência de entrada (i.e., palavras ou sentenças) e sua relevância dentro do todo - diferentemente dos modelos clássicos de RNN que analisavam as sequências de entrada de forma sequencial e não parte de um contexto muito maior. O *transformer* consiste em um codificador que recebe a sequência de entrada e gera representações ocultas (que são informações importantes sobre as palavras ou sentenças e o seu contexto); e de um decodificador que recebe as representações ocultas e gera a sequência de saída. Esse método permite que o modelo consiga lidar melhor com a dependência de longo prazo entre sequências de entrada.

Assim, podemos dizer que o processo de geração conteúdo dos modelos de GenAI pode ser dividido em duas etapas, que são: a extração de informações de intenção das sequências de entradas dadas pelos usuários; e a geração do conteúdo de acordo com as intenções extraídas. Esse processo não é novo, como demonstrado por estudos anteriores [9] [10], mas graças a maior quantidade de dados disponíveis para treinamento, foi possível desenvolver modelos de GenAI mais robustos e sofisticados. Para se ter uma ideia, o GPT-3 (terceira versão do modelo GPT) mantém a estrutura principal do GPT-2, contudo, o tamanho do conjunto de pré-treinamento passou de *WebText* [11] (38GB) para *CommonCrawl* [12] (570GB). Com isso, o GPT-3 possui uma melhor capacidade de generalização do que GPT-2 em várias tarefas, como, por exemplo, na extração de intenção humana.

2.3. - Modelos de Linguagem Mascarada vs Modelos de Linguagem Autorregressiva

Geralmente, esses modelos de linguagem pré-treinados baseados em *transformers* podem ser classificados em dois tipos, que são: os Modelos de Linguagem Mascarada – *Masked Language Model* (MLM) — e os Modelos de Linguagem Autorregressiva – *Autoregressive Language Model* (ALM) — [13]. Antes de definirmos as diferenças entre esses tipos de modelos, é importante falarmos sobre os *tokens*, um conceito fundamental para NLP. Os *tokens* correspondem a menor unidade de uma sentença que pode ser um caractere, uma palavra ou uma pontuação. A tokenização é um processo que visa simplificar a extração de informações de uma sequência de entradas, através da identificação dos *tokens* presentes nas sentenças. Esse processo permite que os algoritmos de AI possam combinar os diferentes *tokens* de acordo com regras, a fim de identificar o significado das sentenças.

Os MLMs são modelos de AI que omitem certos tokens das sentenças de entrada durante o processo de treinamento. Esse tipo de modelo de AI é responsável por prever essas palavras “mascaradas” com base em outras palavras da sentença. Os MLMs são ditos bidirecionais porque as palavras mascaradas podem ser previstas com base nas palavras que ocorrem a sua esquerda e ou direita. Exemplos famosos de ferramentas que utilizam esse modelo são o BERT [14], RoBERTa [15] e XL-Net [16]. Já o GPT-2, GPT-3 [17] e o OPT [18] são ALMs, pois utilizam técnicas autorregressivas para prever a próxima palavra em uma sentença de entrada com base nas palavras que vieram antes dela e no contexto geral das sentenças. Portanto, dizemos que esses modelos são direcionais, da esquerda para direita. Diferente dos modelos de linguagem mascarada, os modelos autorregressivos são mais adequados para tarefas generativas.

2.4. - Aplicações da AI Generativa

Está claro que a área de GenAI tem experimentado um avanço significativo nos últimos anos, com aplicações em diversos campos, como visão computacional, processamento de linguagem natural, áudio e multimodal. Cada um desses campos possui algoritmos e técnicas específicas capazes de alcançar resultados precisos e eficientes. A seguir, descreveremos diversas aplicações desses campos e como elas podem ser utilizadas pelos usuários em tarefas diárias.

No **campo da visão computacional**, as aplicações abrangem tarefas como: estimação de profundidade de imagens; classificação de imagens; segmentação de imagens, técnica utilizada para localizar objetos e limites (e.g., linhas, curvas, etc.); transformação de imagem para imagem, processo aplicado, por exemplo, na transferência de estilo, restauração e colorização; detecção de objetos; classificação de vídeos; geração de imagens incondicionais, procedimento que busca criar imagens originais que não sejam baseadas em imagens existentes; e classificação de imagens sem treinamento prévio. Cada uma dessas tarefas requer uma abordagem específica, que pode envolver a aplicação de algoritmos de Redes Neurais Convolucionais — *Convolutional Neural Network* (CNN) — (modelos muito utilizados no reconhecimento de imagens, por serem capazes de identificar padrões de forma eficiente), redes residuais, Redes Generativas Adversariais — *Generative Adversarial Networks* (GAN) — e outros métodos avançados.

Já no **processamento de linguagem natural**, as tarefas que podem ser realizadas são: aplicações conversacionais, como os *chatbots*; preenchimento de lacunas em textos; respostas a perguntas dos usuários; detecção de similaridade entre sentenças; sumarização de textos; classificação de textos; geração de textos; classificação de *tokens*; tradução de textos; e a classificação de textos sem treinamento prévio. As técnicas que podem ser utilizadas em NLP são as RNN, os *transformers* e os modelos de linguagem pré-treinados.

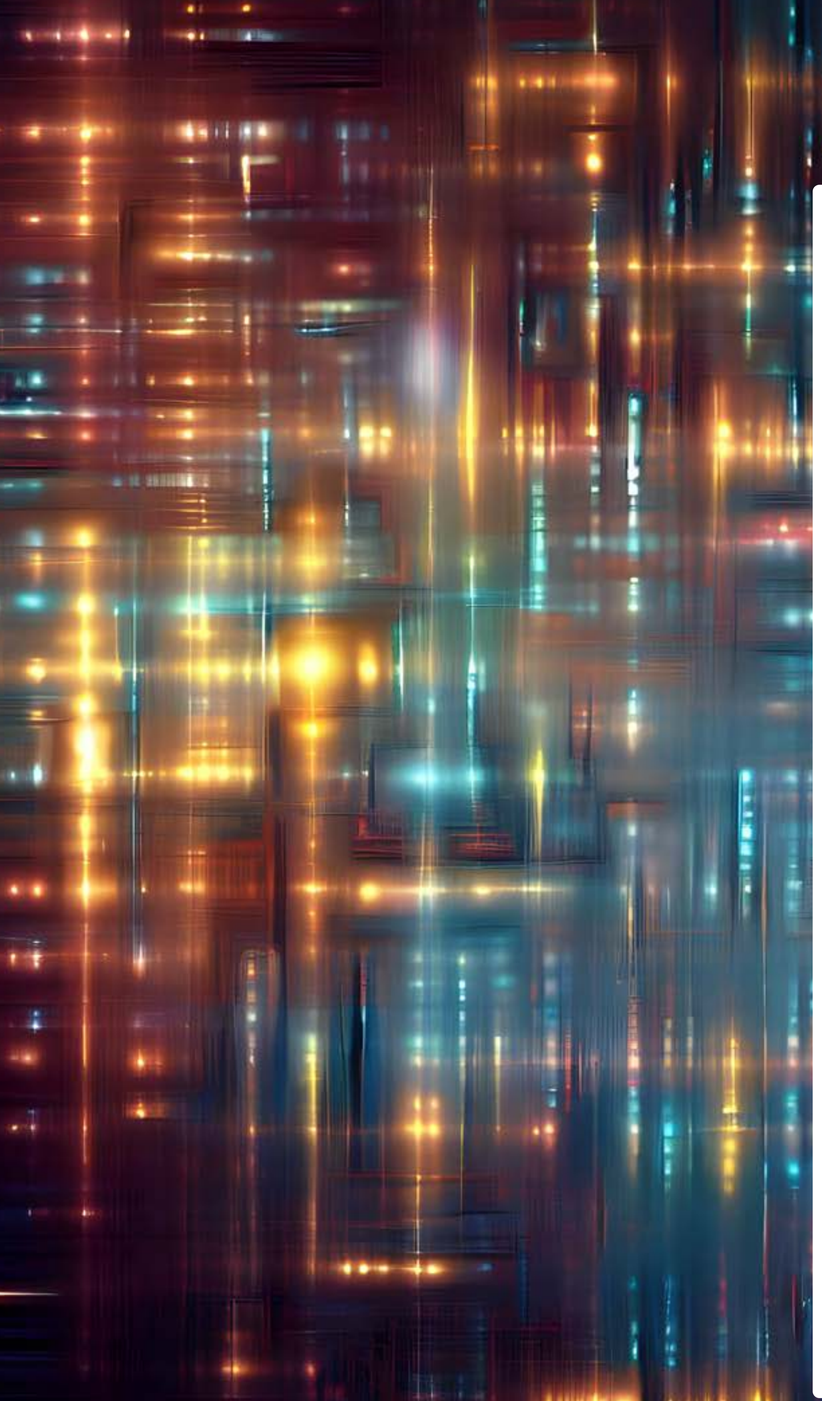
Na área de **áudio**, as tarefas que podem ser executadas são: classificação de áudio; conversão de áudio para áudio; reconhecimento automático de fala; transcrição de texto para fala; e técnicas tabulares para classificação e regressão. Algoritmos como as CNNs, RNNs e redes de atenção (técnica que procura imitar o processo cognitivo de aprendizado, conferindo uma importância maior a determinadas partes dos dados de entrada), são amplamente utilizados para lidar com dados de áudio.

Por último, no **domínio multimodal**, temos a combinação de diferentes modalidades de dados, como, por exemplo, de texto, imagem e vídeo, de modo a criar soluções que são capazes de processar e compreender informações complexas provenientes de múltiplas fontes. As tarefas que podem ser realizadas por aplicações multimodal são: respostas a perguntas acerca do conteúdo presente em documentos; extração de características de textos, áudio ou vídeo; conversão de imagem para texto, de texto para imagem e de texto para vídeo; e respostas a perguntas visuais. O domínio multimodal requer o uso de abordagens como CNNs, RNNs e de Redes Neurais Multimodais.

2.5. - Modelos Base

Os modelos base, também conhecidos como *foundation models*, são modelos de AI pré-treinados em grandes quantidades de dados de texto, imagem ou áudio. Esses modelos são utilizados no reconhecimento de padrões, uma vez que são capazes de aprender representações de alto nível presentes nesses dados. Após o treinamento, esses modelos podem ser aplicados a uma variedade de tarefas na área de visão computacional e no processamento de linguagem natural e de áudio. Esses modelos são criados usando técnicas avançadas, como as Redes Neurais Profundas — *Deep Neural Networks* (DNN) — e os *transformers*, que permitem a captura de relações semânticas entre as sequências de entrada (palavras e frases).

Em GenAI, os modelos base são utilizados para a criação de modelos generativos especializados. Isso é feito a partir do treinamento do modelo base em conjuntos de dados específicos, técnica chamada



de ajuste fino ou *finetune*. Por exemplo, considere um modelo base capaz de identificar muito bem padrões em imagens (e.g., contornos, sombras, objetos, etc). Podemos especializar esse modelo para que seja capaz de gerar imagens de gatos e cachorros. Para isso, precisamos ensiná-lo as características intrínsecas desses animais, o que é feito a partir do retreinamento com um conjunto de imagens específicas de gatos e cachorros. A seguir abordaremos diversos modelos base presentes na literatura.

2.5.1. - Visão Computacional

Na área de visão computacional, dois exemplos proeminentes de modelos generativos são as Redes Generativas Adversárias – *Generative Adversarial Networks* (GANs) e os Modelos de Difusão – *Diffusion Models* (DM). Os modelos GANs consistem em um sistema de dois componentes principais, que são: um gerador, que cria amostras sintéticas, e um discriminador, que avalia se as amostras são reais ou falsas. Isso significa que o gerador é treinado para enganar o discriminador, que por sua vez, a cada interação, está sendo treinado para dizer o quão realista é a saída do gerador. Essa abordagem é bastante interessante pois permite que os modelos GANs aprendam de forma não supervisionada, isto é, quando não se possui rótulos para as instâncias de treinamento.

Por outro lado, os modelos de DMs trabalham diretamente com a imagem alvo, refinando-a passo a passo para melhorar sua qualidade. Os DMs utilizam uma abordagem de amostragem condicional, em que a imagem é inicializada com ruído e é gradualmente refinada através de múltiplas iterações. Em cada iteração, um ruído é adicionado à imagem e a rede de difusão tenta reconstruir a imagem original a partir dessa nova amostra. Esse processo de difusão é repetido várias vezes, refinando a imagem a cada passo e gerando uma sequência de amostras que se aproximam cada vez mais da imagem alvo. Essa técnica permite controlar o processo de geração, ajustando a taxa de difusão e o nível de ruído adicionado a cada iteração. Isso possibilita a criação de imagens com diferentes estilos e variações, tornando a difusão uma ferramenta versátil para tarefas criativas e de geração de conteúdo visual.

Outro modelo notável é o DALL-E, que combina conceitos de GANs e *transformers* para gerar imagens baseadas em descrições de texto. O DALL-E pode criar representações visuais únicas e até mesmo produzir imagens surrealistas com base em descrições não convencionais dadas pelos usuários.

2.5.2. - Processamento da Linguagem Natural

Em NLP, os Modelos de Linguagem Grandes — *Large Language Models* (LLMs) — têm se destacado como uma área promissora de pesquisa. Esses modelos são construídos com base em arquiteturas

avançadas de AI e são treinados em extensas quantidades de texto para desenvolver a capacidade de compreender e gerar texto de forma coerente e fluente. A versatilidade desses modelos de LLMs, aliada à sua habilidade em produzir texto de alta qualidade, tem impulsionado sua adoção em diversas aplicações, desde assistentes virtuais e *chatbots* até a criação de conteúdo criativo e a solução de problemas complexos que envolvam o processamento de linguagem natural. Entre os LLMs notáveis estão o GPT, LaMDA, LLAMA, BART e T5, cada um com suas características distintas.

O modelo GPT é um dos modelos mais conhecidos e influentes no campo de processamento de linguagem. Ele foi desenvolvido pela OpenAI e tem como base a arquitetura *transformer*. O GPT é um modelo de linguagem auto-supervisionado, o que significa que ele é treinado em uma grande quantidade de dados não rotulados para capturar a estrutura, o contexto e as relações semânticas das palavras em um texto. Após o seu pré-treinamento, o GPT pode ser ajustado para desempenhar diversas tarefas em processamento de linguagem como, por exemplo, a geração de textos coerentes. Uma das principais vantagens do GPT e da arquitetura *transformer* é a capacidade de lidar com sequências longas de texto e capturar relações de longo prazo entre as sentenças. Atualmente, o GPT se encontra na sua quarta versão.

Já o BART (*Bidirectional and Auto-Regressive Transformer*) é um modelo que combina a abordagem auto-regressiva, usada pelo GPT, com uma arquitetura bidirecional. Isso significa que o BART pode não apenas gerar texto sequencialmente, mas também pode considerar o contexto tanto anterior quanto posterior à palavra atual, resultando em uma melhor compreensão do texto de entrada e uma geração mais precisa.

O LLaMA (*Large Language Model Meta AI*) é outro exemplo de LLM que se destaca por tomar uma sequência de palavras como entrada e prever a próxima palavra para gerar texto de forma recursiva. Para treinar este modelo, foram escolhidos textos de 20 línguas com o maior número de falantes, dando foco às línguas com alfabetos latino e cirílico. Essas capacidades tornam o LLAMA útil para tarefas de tradução, compressão de texto, geração de resumos e conversação.

O LaMDA, acrônimo para *Language Model for Dialogue Applications*, é um modelo de linguagem para aplicativos de diálogo. Sua base é um *transformer,* ou seja, um emaranhado de redes neurais artificiais profundas que geram uma saída desejada a partir de uma entrada textual. Essa rede neural treina a si própria com grandes quantidades de texto

Por fim, o T5 (Text-to-Text Transfer Transformer) é um modelo versátil que pode ser treinado em uma ampla variedade de tarefas de processamento de linguagem natural. Ele adota uma abordagem de transferência de texto para texto, o que significa que ele pode ser treinado em uma tarefa específica e depois aplicado a diferentes tarefas, como tradução de idiomas, sumarização de texto, resposta a perguntas, entre outras.

2.5.3. - Audio

No campo de áudio, os LLMs também têm desempenhado um papel significativo. Modelos como Wav2Vec2, HuBERT e o Wav2Vec2-XLS-R, são capazes de gerar transcrições e reconhecer a fala humana com alta precisão. Esse fato potencializou o desenvolvimento de uma variedade de aplicações como, por exemplo, os assistentes de voz pessoais e sistemas de tradução. A seguir descreveremos importantes modelos base de áudio.

Wav2Vec2 é um modelo, com duas fases de treinamento, utilizado no reconhecimento automático da fala. Na sua primeira fase, o Wav2Vec2 utiliza uma estratégia de aprendizado auto-supervisionado, onde um conjunto de dados não rotulados é utilizado no treinamento do modelo. Esse pré-treinamento permite o modelo entender melhor a fala, visto que esse processo é responsável por gerar mapeamentos que identificam o significado dos sons e como eles se relacionam. A segunda fase do treinamento utiliza uma estratégia supervisionada e é onde um ajuste fino é feito. Nesta fase, um conjunto de dados rotulados são usados

para treinar o modelo a prever palavras ou fonemas específicos. Os fonemas são a menor unidade possível de som em um determinado idioma, geralmente representado por uma ou duas letras. Dada essa abordagem de treinamento, o Wav2Vec2 pode alcançar bons resultados utilizando apenas uma pequena quantidade de dados rotulados. A principal diferença entre o Wav2Vec 2.0 e as aplicações de NLP, é que o Wav2Vec2 processa áudio em vez de texto.

Já o modelo HuBERT (*Hidden-Unit BERT*), se propõe a resolver três problemas enfrentados pelos métodos de aprendizado auto-supervisionado de representações de fala, que são: a presença de múltiplas unidades sonoras em cada enunciado de entrada; a inexistência de um léxico (dicionário) de unidades sonoras durante a fase de pré-treinamento; e o comprimento variável das unidades sonoras sem segmentação explícita. Para lidar com esses desafios, o HuBERT utiliza uma etapa de agrupamento *offline* para fornecer rótulos aos grupos de unidades sonoras. O modelo HuBERT aplica um método de previsão semelhante ao BERT, onde o modelo é treinado para prever apenas as regiões mascaradas (omitidas). Essa abordagem força o modelo a aprender as representações contextuais da fala, visto que o modelo deve considerar todo o contexto no qual as unidades sonoras aparecem para fazer a previsão. Os autores do HuBERT afirmam que o seu desempenho, em alguns cenários, se equipara ou é melhor que o observado pelo Wav2Vec2. Além do reconhecimento da fala, o HuBERT também pode ser utilizado para geração de falas.

Por fim, temos o Wav2Vec2-XLS-R, desenvolvido pelo Facebook AI, que é um modelo multilínguas pré-treinado para o processamento de fala em mais de 120 idiomas. O Wav2Vec2-XLS-R é treinado com um conjunto de falas não rotuladas contendo 436 mil horas de duração. Os conjuntos de dados utilizados são o VoxPopuli, o MLS, o CommonVoice, o BABEL e o VoxLingua107. O Wav2Vec2-XLS-R utiliza como base o modelo Wav2Vec2, mas o especializa, através de um processo de ajuste fino, para que possa ser utilizado no reconhecimento automático de falas e na tradução ou classificação delas.

2.6. - Ferramentas de AI Generativa

Os avanços nos modelos generativos têm impulsionado uma série de aplicações inovadoras em diferentes domínios. Essas aplicações que serão apresentadas são apenas alguns exemplos das inúmeras possibilidades que os modelos generativos oferecem. À medida que a pesquisa e o desenvolvimento continuam avançando nessa área, é provável que surjam ainda mais inovações e aplicações interessantes, impulsionando a criatividade e a interação entre humanos e máquinas.

2.6.1. - Geração de imagens

O **Canva** é uma plataforma online e gratuita que foi criada em 2013 com o objetivo de permitir que qualquer pessoa no mundo pudesse criar designs para publicação em diversos lugares. Essa ferramenta não requer o *download* ou a instalação de *software* no computador do usuário, tornando-a mais acessível para todos. Com uma interface amigável e mais de 50 mil templates disponíveis, o Canva ajuda a elaborar materiais de comunicação visual com qualidade profissional. Recentemente, o Canva adicionou uma assistente de AI capaz de fornecer recomendações aos usuários sobre gráficos e estilos conforme o tipo de projeto desenvolvido. Além disso, esse assistente permite escrever resumos de apresentações e listar ideias estratégicas que podem ser utilizadas nas redes sociais.

O **Stable Diffusion** é uma ferramenta, desenvolvida pela Stability AI, que utiliza inteligência artificial para gerar imagens foto realistas a partir de qualquer descrição textual dada pelos usuários. Essa ferramenta é capaz de produzir imagens em diversos estilos, alguns dos quais podem se assemelhar até mesmo a famosas obras de arte. Além da geração de imagens por texto, o Stable Diffusion foi adaptado para outras funções relacionadas como a geração de imagens a partir de rascunhos. O grande sucesso do Stable Diffusion se deu pelo fato do modelo ter sido lançado com licenciamento permissivo, isto é, todos os direitos sobre as imagens geradas são dos usuários, com a condição de que elas não sejam ilegais ou prejudiciais. Além disso, qualquer pessoa que possua um computador e uma placa de vídeo pode criar praticamente qualquer pintura digital imaginável.

Semelhante ao Stable Difusion, o **DALL-E** [5] [26], uma combinação de “WALL-E” e “Salvador Dalí”, é uma ferramenta para geração de imagens e ilustrações únicas a partir de uma entrada textual dada pelo usuário. O Dall-E é desenvolvido pela OpenAI e utiliza uma versão do modelo GPT-3 com 12 bilhões de parâmetros, o que lhe permite interpretar melhor as entradas dos usuários e com isso, gerar as imagens correspondentes. O DALL-E pode criar tanto imagens realistas de objetos existentes, quanto imagens de objetos que não existem na realidade.

Midjourney é um programa de inteligência artificial generativa criado e hospedado pelo laboratório de pesquisa Midjourney, Inc. Semelhante ao DALL-E e o Stable Diffusion da Stability AI, essa ferramenta é capaz de gerar imagens a partir de descrições de linguagem natural dadas por usuários. No entanto, ela funciona a partir de um bot em um canal de áudio de uma conhecida ferramenta de *Voice-over-IP* (VOIP) chamada de Discord. Assim, as imagens criadas são disponibilizadas para todos os membros do canal onde as artes foram criadas. Em comparação com o DALL-E, que renderiza imagens de forma mais realista, as imagens criadas pelo Midjourney são mais abstratas e criativas. Vale ressaltar que, embora os resultados do Midjourney sejam impressionantes em muitos casos, a plataforma ainda pode gerar imagens estranhas ou que não atendam às expectativas do usuário.

2.6.2. - Chat e NLP

A aplicação do LLM Processamento de Linguagem Natural tem sido amplamente explorada em diversos contextos, visando melhorar a interação e a compreensão de sistemas de conversação e assistentes virtuais. Duas aplicações notáveis são o ChatGPT e o Bard.

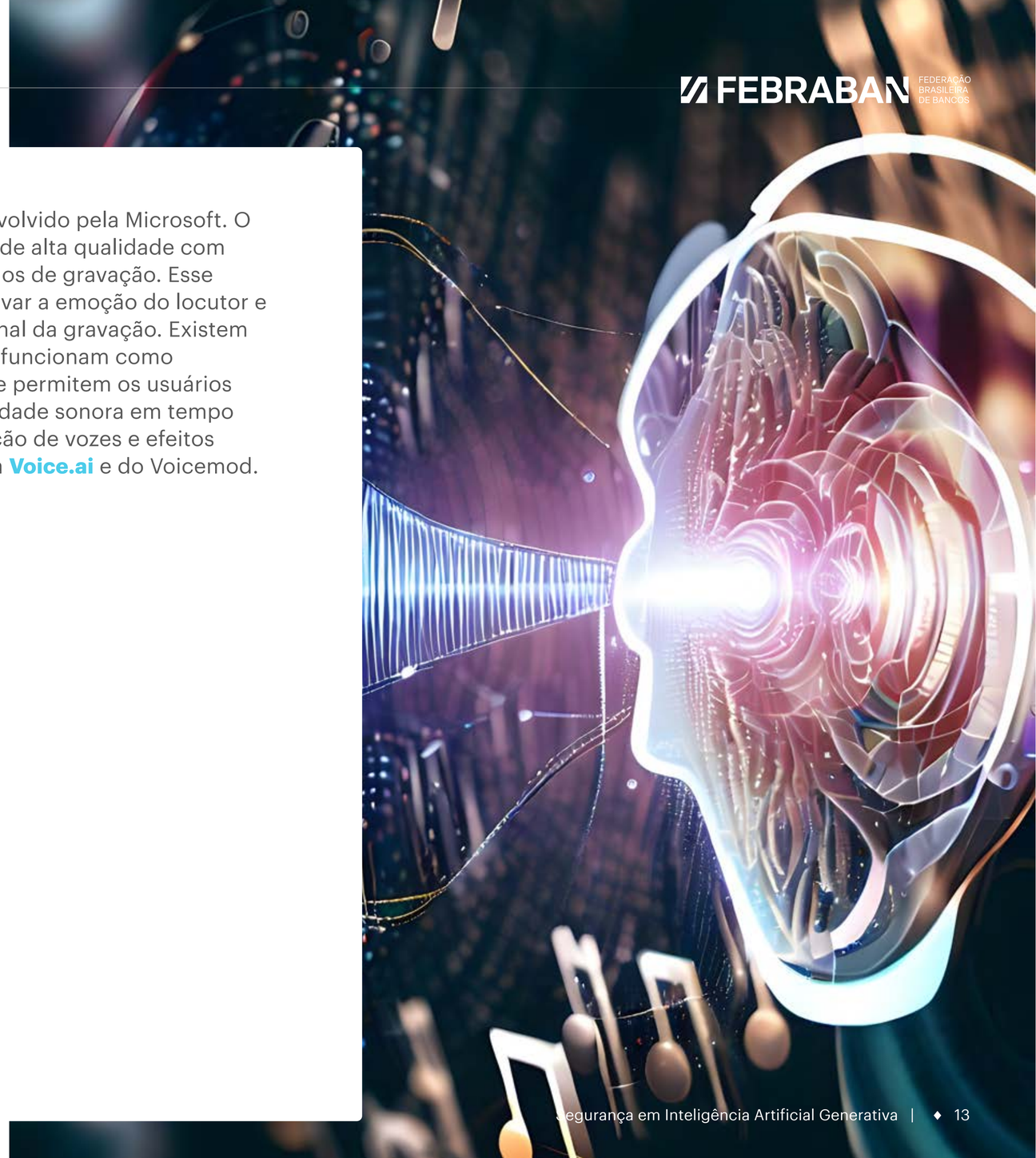
O **ChatGPT** é uma ferramenta de chat desenvolvida pelo laboratório de pesquisas em inteligência artificial OpenAI, que utiliza em sua arquitetura uma rede neural chamada de *transformer* e o aprendizado por reforço com *feedback* humano. Essa abordagem permite que a ferramenta aprimore suas respostas, garantindo assim, um sistema mais eficiente e personalizado. O **Bard** é uma ferramenta de *chat* desenvolvido pelo Google. Ele possui a capacidade de esclarecer dúvidas de forma organizada e original, além de acessar dados atualizados da internet para fornecer resultados mais aprofundados. O Bard foi criado com o objetivo

de tornar as pesquisas mais precisas. Assim, podemos dizer que o Bard se diferencia do ChatGPT por sua capacidade de buscar informações atualizadas diretamente na internet, enquanto o ChatGPT está limitado a conteúdos publicados até 2021.

2.6.3. - Audio

A **ReadSpeaker** é uma ferramenta, desenvolvida pela empresa de mesmo nome, que oferece soluções para conversão de texto em voz de forma *online* e *offline*. Com suporte para uma ampla variedade de idiomas, o ReadSpeaker oferece aplicações como: o webReader, que permite que os conteúdos digitais sejam lidos em voz alta com vozes mais realistas; o docReader, que torna os documentos mais acessíveis, permitindo que os usuários os ouçam em qualquer dispositivo, sem a necessidade de *plugins;* o TextAid que oferece apoio à literacia de pessoas com dificuldade em ler; e permite o leitura de textos em imagens por meio de técnicas de Reconhecimento Ótico de Caracteres — *Optical Character Recognition* (OCR.).

Além disso, existem ferramentas disponíveis no mercado que exploram a tecnologia de áudio generativo a partir de textos, o chamado *Text-To-Speech* (TTS). Um exemplo é o **TorToiSe**, que utiliza redes neurais para imitar vozes com base em exemplos de texto fornecidos. Outro modelo que também utiliza TTS é o VALL-E (*Neural Codec Language Models*), desenvolvido pela Microsoft. O VALL-E permite gerar fala de alta qualidade com base em apenas 3 segundos de gravação. Esse modelo é capaz de preservar a emoção do locutor e o ambiente acústico original da gravação. Existem também ferramentas que funcionam como modificadores de voz, que permitem os usuários personalizarem sua identidade sonora em tempo real, com uma vasta coleção de vozes e efeitos sonoros como é o caso da **Voice.ai** e do Voicemod.



③ RISCOS À SEGURANÇA

Como temos observado, a GenAI, uma forma avançada de AI capaz de criar conteúdo original, tem despertado admiração por suas aplicações criativas em diversas áreas. No entanto, é importante observar que a utilização de GenAI envolve aspectos que colocam em risco a segurança, conforme ilustrado na Figura 1. Ao centro, temos os riscos e nos vértices as ameaças, as vulnerabilidades e os ataques e explorações.

Primeiro, apresentaremos as potenciais ameaças decorrentes do uso massivo da GenAI, tais como deepfakes e clonagem de voz. Em seguida, discutiremos as principais vulnerabilidades que podem ser exploradas e acabar gerando impactos negativos às empresas.

Por fim, falaremos sobre a matrixz ATLAS e como sua aplicação pode ajudar a identificar e mitigar as ameças de GenAI.

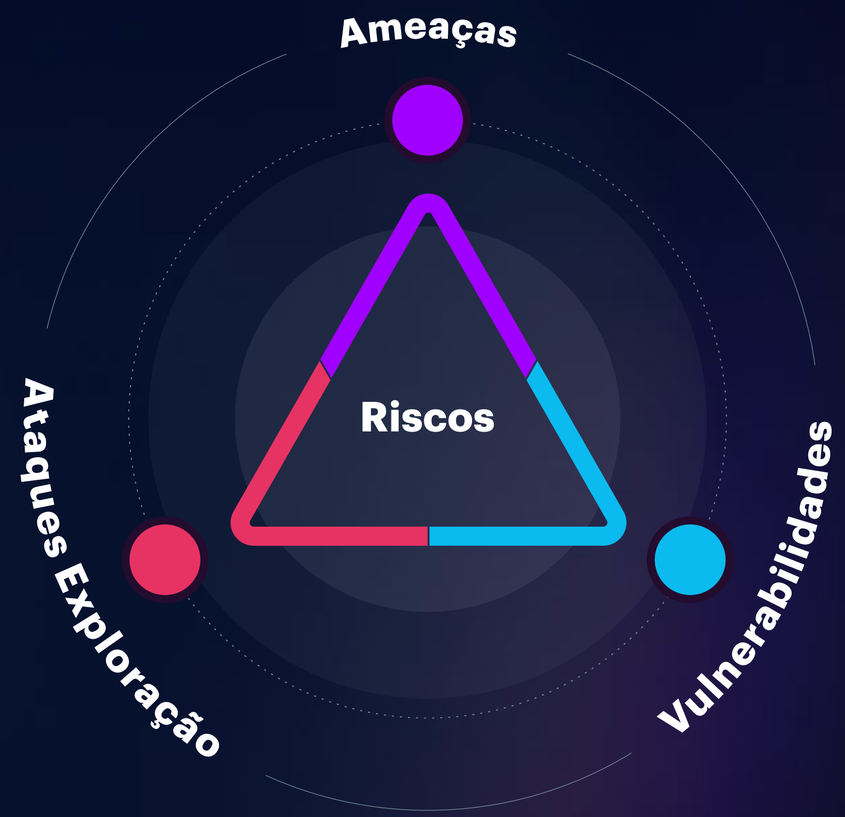


Figura 1: Triângulo de Risco. Origem: o autor.

3.1. - Potenciais Ameaças

A seguir, abordaremos as principais ameaças provenientes do uso de ferramentas que utilizem a inteligência artificial generativa.

3.1.1. - Deepfake

A criação de *deepfakes,* por meio do uso de GenAI, representa uma das ameaças mais preocupantes no cenário atual. Os *deepfakes* são conteúdos de mídia falsificados (áudio ou vídeo), que são projetados para parecerem autênticos. Os *deepfakes* podem ser utilizados para manipular pessoas e disseminar informações falsas. Essa tecnologia tem o potencial de causar sérios danos a reputação, privacidade e até mesmo à confiabilidade de pessoas, empresas e dos próprios veículos jornalísticos.

Como exemplo do poder destrutivo dessa ameaça, imagine a situação em que uma *deepfake* de um político influente é criada com o uso de GenAI. Nesse vídeo falso, o político é mostrado fazendo declarações falsas e controversas, levando ao descrédito de suas políticas e prejudicando sua imagem perante o público. Esse conteúdo manipulado é compartilhado em larga escala nas redes sociais e em outras plataformas de mídia, alcançando milhões de pessoas. Como resultado, a reputação do político é prejudicada e a confiança do público em suas ações e palavras é abalada.

Os *deepfakes* também podem ser utilizados para difamar celebridades, através da criação de vídeos falsos que as mostrem envolvidas em situações constrangedoras, ilegais ou moralmente questionáveis. Essas falsificações podem levar à destruição da imagem pública dessas personalidades, impactando suas carreiras e vida pessoal de maneiras devastadoras. A capacidade viral dos *deepfakes* pode causar um efeito em cascata, espalhando informações falsas e prejudicando a percepção pública sobre essas figuras.

Outra ramificação preocupante é a capacidade dos *deepfakes* de minar a confiança do público nos veículos jornalísticos. Com a disseminação de vídeos falsos que, muitas vezes, são indistinguíveis de conteúdos reais, a manipulação da informação se torna cada vez mais fácil. Isso pode levar a uma crescente desconfiança em relação aos relatos jornalísticos, colocando em risco a integridade e a objetividade da mídia. A disseminação em massa de *deepfakes* pode alimentar narrativas distorcidas e prejudicar a sociedade como um todo.

Para lidar com esse problema, é necessário um esforço conjunto de pesquisadores, governos e da indústria para desenvolver técnicas avançadas de detecção e criação de mecanismos eficazes de verificação de autenticidade, além do aprimoramento dos métodos de autenticação de identidade para mitigar tais ameaças.

3.1.2. - Clonagem de Voz

Assim como os *deepfakes*, a clonagem de voz atrai preocupações para a área de GenAI. Por meio dessa tecnologia, é possível criar gravações falsas que imitam de maneira surpreendentemente real a voz de uma pessoa específica. Essa capacidade de clonagem de voz tem o potencial de ser explorada em uma variedade de cenários criminosos, colocando em risco a autenticação de identidade e a segurança em sistemas de voz automatizados.

Imagine a seguinte situação: um indivíduo mal-intencionado obtém acesso a uma gravação da voz de uma pessoa conhecida, seja por meio de uma ligação telefônica ou até por vídeos publicados em suas redes sociais. Usando GenAI, esse indivíduo mal-intencionado é capaz de criar uma imitação perfeita da voz da vítima. Com essa gravação falsa em mãos, o golpista pode realizar chamadas telefônicas fraudulentas, entrando em contato com pessoas próximas à vítima, como amigos, familiares ou colegas de trabalho. Durante essas chamadas, o golpista pode se passar pela pessoa conhecida, imitando sua voz com precisão e convencendo as vítimas a fornecerem informações confidenciais, como senhas, números de contas bancárias ou detalhes de cartões de crédito. Esses dados podem ser utilizados para cometer diversos tipos de

fraudes financeiras, como transferências indevidas de dinheiro ou compras fraudulentas.
Neste sentido, observando o cenário brasileiro, percebemos que esta tecnologia pode potencializar golpes como, por exemplo, o do PIX via WhatsApp, onde o criminoso ao invés de tratar exclusivamente com a vítima via mensagens de texto de um número de celular tido como novo, inicia a troca de mensagens via voz tornando o golpe mais convincente.

Além disso, a clonagem de voz pode ser empregada para enganar sistemas de voz automatizados, como assistentes virtuais e autenticação por reconhecimento de voz em dispositivos eletrônicos. Por exemplo, um golpista pode utilizar a voz clonada para contornar os sistemas de autenticação biométrica, permitindo acesso não autorizado a dispositivos ou informações confidenciais armazenadas neles.

Esses exemplos ilustram o potencial impacto prejudicial da clonagem de vozes por meio da GenAI. A capacidade de criar gravações falsas realistas representa uma ameaça significativa à autenticação de identidade e à segurança em sistemas de voz automatizados. Medidas de segurança robustas, como autenticação multifatorial e análise detalhada de padrões de voz, são essenciais para mitigar os riscos decorrentes dessa tecnologia.

3.1.3. - Criação de vídeos e modelos 3D a partir de fotos

Com base em algumas fotografias, a GenAI pode criar vídeos e modelos 3D realistas de pessoas que nunca existiram ou que não deram consentimento para tal uso de sua imagem. Essa tecnologia tem o potencial de ser explorada para criar perfis falsos, espalhar desinformação ou até mesmo gerar material pornográfico não consensual, causando danos psicológicos e violando a privacidade das pessoas. Além disso, a capacidade de criar vídeos e modelos 3D realistas a partir de uma única foto, pode permitir a criação de vídeos falsos que levem a fraudes através da autenticação por reconhecimento facial. Este tipo de fraude pode impactar várias instituições financeiras, principalmente as *fintechs* que têm adotado cada vez mais o reconhecimento facial como forma de autenticação adicional para seus serviços.

A implementação de regulamentações e políticas de proteção de dados eficazes são essenciais para garantir que a criação e disseminação desse tipo de conteúdo sejam restritas e controladas. Para sistemas de autenticação e validação, a existência de camadas adicionais de segurança que verifiquem a presença de organismos vivos, como é o caso da circulação sanguínea, é uma solução candidata a prevenção de fraudes.

3.1.4. - Produção, assistência e execução de conteúdo criminoso

Com sua capacidade de gerar conteúdo realista, a GenAI tem se mostrado como uma ferramenta poderosa nas mãos de criminosos, que buscam explorar suas capacidades para cometer uma variedade de crimes. Dentre eles, destacam-se os crimes contra a honra, como calúnia, difamação e injúria, que podem ser amplificados pela disseminação em larga escala de informações falsas e prejudiciais.

Imagine a seguinte situação: um indivíduo mal-intencionado utiliza a GenAI para criar um artigo de notícia falso, com informações difamatórias sobre uma pessoa pública. Esse conteúdo, compartilhado através de redes sociais e outros meios de comunicação, causa danos à reputação e à vida pessoal e profissional dessa pessoa, levando a consequências devastadoras em sua vida.

Além disso, a GenAI também pode ser empregada na prática de estelionato, *SCAM* e *phishing*, que envolvem a criação de mensagens enganosas e persuasivas para induzir pessoas a revelar informações confidenciais ou realizar transações financeiras fraudulentas. Por exemplo, um e-mail convincente que parece ter sido enviado por um banco ou instituição financeira, solicitando que o destinatário forneça seus dados pessoais e bancários. Com a ajuda da GenAI, é possível criar um texto extremamente convincente, dificultando a identificação da fraude e aumentando as chances de sucesso do golpe.

Os riscos não param por aí. A GenAI também pode ser explorada por *hackers* em suas atividades ilícitas. Ao criar programas de *malware* personalizados, capazes de enganar sistemas de segurança e explorar vulnerabilidades, os *hackers* podem se infiltrar em redes, roubar informações valiosas, causar danos aos sistemas e até mesmo extorquir vítimas. Esses ataques, impulsionados pela geração automatizada de conteúdo malicioso, podem levar a perdas financeiras significativas e comprometimento da segurança de indivíduos e organizações.

A GenAI também pode ser utilizada como uma ferramenta facilitadora do terrorismo, permitindo a criação de conteúdo de propaganda radicalizado, recrutamento de membros para organizações extremistas e até mesmo na disseminação de instruções detalhadas para a realização de ataques. A natureza persuasiva e personalizada desse conteúdo pode tornar a influência e disseminação de ideologias extremistas ainda mais perigosas e impactantes.

Esses exemplos situacionais ilustram a extensão dos riscos de segurança associados à GenAI. As implicações nefastas dessas atividades criminosas reforçam a necessidade de medidas de segurança eficazes e regulamentações adequadas para mitigar os danos potenciais causados por essa tecnologia. Esforços devem ser direcionados para identificar e prevenir a criação e disseminação de informações falsas, bem como a utilização maliciosa dessa tecnologia para fins criminosos, como estelionato, *phishing* e terrorismo. Para isso, a cooperação entre agências de aplicação da lei, instituições acadêmicas e empresas de tecnologia é crucial para enfrentar esses desafios e desenvolver soluções efetivas. Além disso, é importante incluir seres humanos em pontos estratégicos para análise e supervisão. Isso significa ter especialistas humanos responsáveis pela revisão e validação do conteúdo gerado, a fim de garantir que os LLMs estejam em conformidade com as diretrizes estabelecidas.

3.1.5. - Internalização de sistemas de GenAI

A internalização de sistemas de GenAI, que ocorre quando as empresas começam a utilizar ferramentas de GenAI públicas nos processos internos, traz consigo diversas ameaças que merecem atenção como, por exemplo, a confiança cega nos conteúdos gerados e a geração de código vulnerável. No entanto, a ameaça mais conhecida é o compartilhamento de informações confidenciais, mesmo que de forma não intencional, com o modelo. Isso pode ocorrer quando o usuário

compartilha informações sensíveis como nome completo, CPF ou até mesmo segredos da empresa, acreditando que os modelos de LLMs são seguros.

Uma alternativa para o uso de ferramentas públicas de GenAI seria a criação de instâncias próprias pelas organizações. Como exemplo, podemos citar o LLAMA, desenvolvido pelo Meta, que é um modelo LLM de código aberto ajustado para ser utilizado em aplicativos de chat e pode substituir localmente o ChatGPT. No entanto, surge a dúvida se essas instâncias possuem os mesmos padrões de desenvolvimento e execução que a ferramenta original. Um dos problemas que podem ocorrer é tais instâncias permitirem a geração de respostas maliciosas, inadequadas ou criminosas, por não possuírem as mesmas políticas de uso e privacidade da OpenAI. Outro ponto importante é em relação a evolução dessas instâncias. É possível que elas sejam treinadas e aprimoradas internamente, no entanto, elas podem não acompanhar o mesmo nível de desenvolvimento da ferramenta original.

É necessário também a adoção de medidas de segurança adequadas para proteger os *prompts* internos do serviço. Essas medidas podem incluir criptografia dos dados, implementação de políticas de acesso restrito e auditorias regulares para identificar possíveis vulnerabilidades. Por outro lado, quando se trata da implementação de sistemas de GenAI para interação com clientes, como em um *chat*, existe a possibilidade de ataques de *jailbreak* (que é quando usuários mal-intencionados exploram vulnerabilidades no sistema para obter acesso não autorizado e modificar seu funcionamento).

Assim, ao utilizar ferramentas de GenAI, as empresas devem considerar questões como a proteção de dados sensíveis, a conformidade com regulamentações de privacidade, a integridade dos dados de treinamento e os possíveis vieses ou discriminações presentes nos modelos utilizados. Para mitigar esses riscos, é fundamental adotar uma abordagem cuidadosa e responsável, garantindo o uso de práticas adequadas de desenvolvimento seguro, governança de dados e de ética em todas as etapas do processo, que engloba as etapas de desenvolvimento, implantação e uso efetivo da ferramenta.

3.2. - Tipos de Vulnerabilidades em aplicações de AI

Além de suas capacidades impressionantes, os LLMs estão sendo integrados a outros aplicativos em um ritmo acelerado. Nos poucos meses após o lançamento do ChatGPT, testemunhamos o lançamento de vários *plugins* quase diariamente. No entanto, argumentamos que essa corrida de integração de AI não é acompanhada de proteções adequadas e avaliações de segurança. A seguir, listaremos e discutiremos sobre dez vulnerabilidades, elencadas pelo OWASP Top 10, que afetam as aplicações de AI que utilizam modelos base de LLMs [27]. Vale destacar que as vulnerabilidades descritas a seguir correspondem a versão 0.1 do OWASP Top 10 for Large Language Model Applications.

3.2.1. - *Prompt Injections*

O aprendizado baseado em comandos (*prompts*) é uma estratégia bastante utilizada para aprimorar o treinamento de modelos de AI sem a necessidade de realizar um novo treinamento. O ChatGPT, por exemplo, está sendo continuamente ajustado por meio dos comandos fornecidos pelos usuários. Assim, um *prompt* bem elaborado pode ajudar o modelo a gerar saídas mais precisas e relevantes, enquanto um *prompt* mal elaborado pode levar a saídas incoerentes ou irrelevantes.

Os ataques de *Prompt Injection* (PI) representam uma ameaça significativa à segurança das ferramentas de GenAI. Até o presente momento, os ataques conhecidos de PI estão limitados principalmente a indivíduos que atacam suas próprias instâncias LLM (i.e., um modelo público como ChatGPT instalado localmente). No entanto, o desenvolvimento de integrações dessas LLMs com outros aplicativos, pode torná-las suscetíveis à ingestão de dados não confiáveis que contenham comandos maliciosos, as chamadas Injeções

Indiretas de Comandos — *Indirect Prompt Injection* (IPI).

Esses novos vetores de ataque que utilizam técnicas de IPI, permitem que os adversários explorem remotamente (i.e., sem uma interface direta) aplicativos integrados aos LLMs, injetando estrategicamente prompts maliciosos nas entradas do sistema. Isso pode levar ao controle remoto do modelo, comprometimento persistente, roubo de dados e negação de serviço [28]. Além disso, esses ataques podem permitir que os usuários mal-intencionados disseminem informações falsas e manipulem os usuários. Por fim, mesmo em soluções "caixa-preta", que possuam mecanismos para mitigação de ataques [3], usuários mal-intencionados podem, por meio de ataques PI, contornar as restrições e obter acesso ao código fonte do modelo [24] [29].

Cenário #1: um atacante cria um *prompt* malicioso que faz com que a aplicação revele informações confidenciais, como credenciais de usuário ou detalhes do sistema interno. Isso só é possível porque o prompt malicioso é construído de modo a enganar o modelo, que pensa que a solicitação é legítima;

Cenário #2: um usuário mal-intencionado através de padrões de linguagem, *tokens* ou mecanismos de codificação específicos, consegue ignorar os filtros de conteúdo restrito. Com isso, o LLM acaba permitindo que o usuário execute ações que deveriam ser bloqueadas.

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:
Para se defender dessa vulnerabilidade, os modelos devem implementar estratégias de validação e sanitização dos prompts fornecidos pelos usuários. Além disso, é importe que os modelos de LLMs sejam regularmente ajustados para melhorar sua compreensão do contexto, a fim de identificar entradas maliciosas. Por fim, é importante que se monitore e registre as interações dos usuários com os LLMs, de modo a se identificar possíveis tentativas de injeção imediata.

3.2.2. - *Data Leakage*

Prevenir o vazamento de dados e garantir a privacidade dos usuários é uma prioridade para toda organização. Não é incomum que usuários mal intencionados, através de ataques adversariais, tentem manipular ou induzir sistemas a exporem informações confidenciais. No entanto, com o advento de ferramentas como o ChatGPT, o compartilhamento não intencional de informações sensíveis se tornou uma grande preocupação para as empresas.

O compartilhamento não intencional de informações confidenciais ocorre quando um usuário, inadvertidamente, compartilha dados pessoais ou confidenciais com o sistema de AI acreditando que ele é seguro [30]. O ChatGPT, por exemplo, utiliza uma técnica chamada de memória contextual para lembrar e fazer referências a entradas anteriores dos usuários, garantindo respostas mais relevantes e consistentes. Essa memória contextual é finita (a OpenAI nunca divulgou os limites exatos, mas pesquisadores acreditam que a ferramenta só possa processar 3.000 palavras por vez) e só se aplica à sua conversa atual. No entanto, os termos de uso da OpenAI afirmam que a empresa coleta os dados de entrada de ferramentas, como ChatGPT e Dall-E, para serem usadas para fins estritamente de pesquisa e desenvolvimento de produtos [31].

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um usuário inadvertidamente faz uma pergunta ao LLM, que por não possuir uma filtragem de saída adequada, pode revelar informações confidenciais;

Cenário #2: um atacante, através de prompts cuidadosamente elaborados, consegue extrair informações confidenciais que o LLM memorizou durante o seu treinamento.

Para se defender dessa vulnerabilidade, os modelos de LLM devem implementar rígidos mecanismos de reconhecimento de contexto para impedir que informações confidenciais sejam reveladas. Além disso, técnicas de privacidade e de anonimização de dados devem ser utilizadas durante o processo de treinamento dos modelos, de modo a reduzir o risco de overfitting (quando o modelo "memoriza" os dados de treinamento). Por fim, as respostas dadas pelos LLMs devem ser regularmente auditadas e analisadas, de modo a garantir que as informações confidenciais não sejam divulgadas inadvertidamente.

3.2.3. - *Inadequate Sandboxing*

Sandboxing é o processo de operar um ambiente seguro e isolado desacoplado da infraestrutura circundante. Destina-se a impedir que uma ameaça em potencial entre na rede. As *sandboxes* são ambientes seguros onde qualquer coisa que dê errado não pode prejudicar diretamente suas máquinas *host*. Sem uma *sandbox*, aplicativos ou *software* podem ter acesso irrestrito a todas as informações do usuário e suprimentos do sistema de rede.

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um atacante pode explorar o acesso de uma LLM a um banco de dados criando prompts que instruam a extrair e revelar informações confidenciais;

Cenário #2: o LLM possui permissão para executar ações no sistema e um atacante o manipula para executar comandos não autorizados.

Para se defender dessa vulnerabilidade, devem ser implementadas técnicas adequadas de *sandbox* para isolar os LLMs de outros sistemas e recursos críticos. Além disso, o acesso dos LLMs a recursos confidenciais e suas capacidades computacionais devem ser limitadas ao mínimo necessário para que desempenhe suas atividades. Por fim, o ambiente no qual se encontra a LLM, bem como seus controles de acesso, devem ser auditados e revisados regularmente de modo a garantir que o isolamento adequado seja mantido.

3.2.4. - *Unauthorized Code Execution*

A execução de código não autorizado ocorre quando os invasores criam comandos que acionam a execução de um código malicioso e não autorizado pelas ferramentas de GenAI, como os *chatbots*. Com isso, os invasores podem explorar vulnerabilidades e obter acesso parcial ou total sobre a máquina *host*. Esse tipo de vulnerabilidade apresenta alto risco, pois pode permitir o acesso não autorizado a dados sensíveis ou o comprometimento da confidencialidade, integridade e disponibilidade de todo o aplicativo.

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um atacante cria um prompt que instrui o LLM a executar um comando que inicia um shell reverso no sistema, concedendo ao invasor acesso não autorizado ao ambiente;

Cenário #2: o LLM tem permissão, não esperada, para interagir com uma API presente no mesmo ambiente. O atacante, sabendo disso, manipula o LLM para que ela envie requisições a API, que por sua vez, executa ações não autorizadas no sistema.

Para se defender dessa vulnerabilidade, devem ser implementados rígidos processos de validação e sanitização de entradas, para evitar que prompts

maliciosos ou inesperados sejam processados pelos LLMs. Além disso, o uso de *sandboxes*, para restringir o acesso a recursos e limitar a capacidade computacional das LLMs, é encorajado. Por fim, o ambiente e as interações dos LLMs devem ser auditadas e analisadas, a fim de verificar os controles de acesso e detectar a possível execução de códigos não autorizados.

3.2.5. - *Server-side Request Forgery*

Vulnerabilidades do tipo *Server-side Request Forgery* (SSRF) ocorrem quando um invasor explora uma LLM para executar solicitações não intencionais ou acessar recursos restritos, como serviços internos, APIs ou armazenamentos de dados. Apesar das informações confidenciais serem o alvo mais popular dos ataques SSRF, muitas das vezes, apenas alguns metadados são obtidos com o ataque. No entanto, esses dados não são menos críticos, pois permitem que os invasores aprendam sobre o sistema alvo. Através do uso do SSRF, os usuários mal-intencionados podem inclusive expor recursos internos a LLM.

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um atacante cria um *prompt* que instrui o LLM a fazer uma solicitação a um serviço interno, contornando os controles de acesso e obtendo acesso não autorizado a informações confidenciais;

Cenário #2: uma configuração incorreta nas configurações de segurança do aplicativo permite que o LLM interaja com uma API restrita e um atacante manipula o LLM para acessar ou modificar dados confidenciais.

Para se defender dessa vulnerabilidade, deve ser implementado a validação e sanitização para evitar que *prompts* maliciosos ou inesperados iniciem solicitações não autorizadas com o servidor. Encorajasse o uso de uma *sandbox* para restringir o acesso da LLM a recursos da rede, serviços e APIs.

Além disso, as configurações de segurança da rede e dos aplicativos devem ser periodicamente auditadas e revisadas para garantir que recursos internos não sejam expostos inadvertidamente ao LLM.

3.2.6. - *Overreliance on LLM-generated Content*

É notável, que os LLMs são capazes de produzir textos coerentes e que se assemelham à textos escritos por seres humanos. No entanto, atualmente, não é incomum que esses textos demonstrem falta de bom senso e contenham informação completamente falsas. Mas, com o avanço das tecnologias de AI, a tendência é que essas ferramentas fiquem mais precisas ao ponto de que será cada vez mais difícil distinguir entre um texto escrito por um ser humano e um gerado por máquina.

Este fato por si só irá criar tipos novos de desafios. Por exemplo, as ferramentas de LLMs podem ser utilizadas para plagiar textos escritos por seres humanos, fraudar documentos e disseminar desinformações. Além disso, com a ajuda dos LLMs, será ainda mais fácil criar uma quantidade infinita de texto para *troll farms* (um grupo institucionalizado de *trolls* da internet que busca interferir nas opiniões políticas e na tomada de decisões) e sites falsos, o que, por sua vez, levará a um declínio no nível de confiabilidade na internet [32]. Mesmo que os resultados dos LLMs muitas vezes pareçam muito convincentes, eles deixarão de ser confiáveis.

Outro aspecto a se considerar, é a capacidade das LLMs de gerar trechos de código ou até mesmo programas inteiros de forma automática [33]. Essa funcionalidade é extremamente útil, especialmente para tarefas de codificação repetitivas como, por exemplo, a criação de formulários, interação com banco de dados, geração de testes automatizados, etc [34]. No entanto, códigos gerados por AIs não seguem os conceitos de desenvolvimento seguro, dessa forma, podem conter graves vulnerabilidades de segurança. Confiar cegamente em códigos produzidos por LLMs significa, potencialmente, introduzir vulnerabilidades, mesmo que involuntariamente, em soluções de produção.

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um site de notícias utiliza uma ferramenta de LLM para gerar artigos sobre diversos tópicos. O LLM gera um artigo contendo informações falsas que são publicadas sem verificação. Os leitores confiam no artigo, levando à disseminação de desinformação;

Cenário #2: uma empresa conta com um LLM para gerar análises e relatórios financeiros. O LLM gera um relatório contendo dados financeiros incorretos, que a empresa utiliza para tomar decisões críticas de investimento. Isso resulta em perdas financeiras significativas devido à dependência de conteúdo impreciso gerado pelo LLM;

Cenário #3: um desenvolvedor utiliza uma LLM para desenvolver um formulário para login de usuários em um sistema de produção. Esse código, contém uma falha de segurança que permite o uso de comandos de *SQL Injection*. Um atacante, portanto, poderá obter acesso a dados confidenciais da ferramenta devido a falha.

Para se defender dessa vulnerabilidade, os usuários devem verificar o conteúdo gerado pelos LLMs e consultar fontes alternativas antes de tomar decisões ou aceitar as informações como fatos. Além disso, devem ser implementados processos de revisão para garantir que o conteúdo gerado seja preciso, apropriado e imparcial.

Em relação a geração de código automatizado, os desenvolvedores devem assumir que todo código gerado possui vulnerabilidades e por isso, deve ser revisado para se identificar problemas de segurança. Ademais, a documentação oficial das ferramentas e bibliotecas utilizadas no desenvolvimento das soluções devem ser sempre consultadas.

3.2.7. - *Inadequate AI Alignment*

À medida que a inteligência artificial evolui e as máquinas se tornam cada vez mais capazes [35] [36], a consideração de questões éticas e o alinhamento da AI torna-se cada vez mais importante. Existe o risco de que, se a inteligência artificial não for cuidadosamente projetada, isso possa ter consequências catastróficas para a humanidade [37] [38] [39]. Por exemplo, se a AI não for projetada para levar em consideração os valores humanos, ela poderá tomar decisões que sejam prejudiciais aos seres humanos [37].

Alternativamente, se a inteligência artificial não for projetada para ser transparente e compreensível para os humanos, ela pode tomar decisões opacas e difíceis de entender [40]. À medida que a inteligência das máquinas se torna rapidamente mais poderosa [36], as apostas associadas ao problema de alinhamento da AI só aumentam.

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um LLM treinado para otimizar o engajamento entre usuários pode inadvertidamente priorizar o uso de conteúdo controverso ou polarizador, resultando na disseminação de informações incorretas ou prejudiciais;

Cenário #2: um LLM projetado para auxiliar nas tarefas de administração do sistema está desalinhado, fazendo com que ele execute comandos nocivos ou priorize ações que degradam o desempenho ou a segurança do sistema.

Para se defender dessa vulnerabilidade, é necessário que os objetivos e o comportamento esperado dos LLMs sejam definidos com clareza durante o processo de design e desenvolvimento. Além disso, é necessário que as funções de recompensa e os dados de treinamento estejam alinhados com os resultados desejados e não encorajem comportamentos indesejados ou prejudiciais. Deve-se também implementar mecanismos de monitoramento e *feedback* para avaliar continuamente o desempenho e o alinhamento do LLM. Por fim, sempre que necessário, o modelo deve ser atualizado para melhorar o seu alinhamento.

3.2.8. - *Insufficient Access Controls*

Controle de acesso insuficiente é uma falha que ocorre quando um sistema não restringe o acesso aos seus recursos, permitindo que usuários não autorizados interajam com o LLM e potencialmente explorem uma vulnerabilidade. O controle de acesso envolve o uso de diversos mecanismos de proteção como: autenticação (comprovar a identidade de um ator), Autorização (garantir que um determinado ator possa acessar um recurso) e responsabilidade (rastrear as atividades que foram executadas). Quando algum desses mecanismos de proteção não é aplicado ou falha, os invasores podem comprometer a segurança do sistema obtendo privilégios, lendo informações confidenciais, executando comandos maliciosos, etc. Exemplos de vulnerabilidades de controle de acesso impróprio incluem:

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um atacante obtém acesso não autorizado a um LLM devido a mecanismos fracos de autenticação. Isso permite que ele explore vulnerabilidades ou manipule o sistema;

Cenário #2: um usuário com permissões limitadas é capaz de realizar ações além do escopo pretendido devido à implementação inadequada de controles de acesso, podendo causar danos ou comprometer o sistema.

Para se defender dessa vulnerabilidade, deve-se implementar mecanismos robustos de autenticação, como é o caso da autenticação de multifator, para garantir que apenas usuários autorizados possam acessar as LLMs. Além disso, recomenda-se o uso do Controle de Acesso Baseado em Função — *Role-based Access Control* (RBAC) – para definir e impor permissões de usuário com base em suas funções e responsabilidades. Por fim, é necessário que se audite e atualize regularmente os controles de acesso para manter a segurança e impedir o acesso não autorizado.

3.2.9. - *Improper Error Handling*

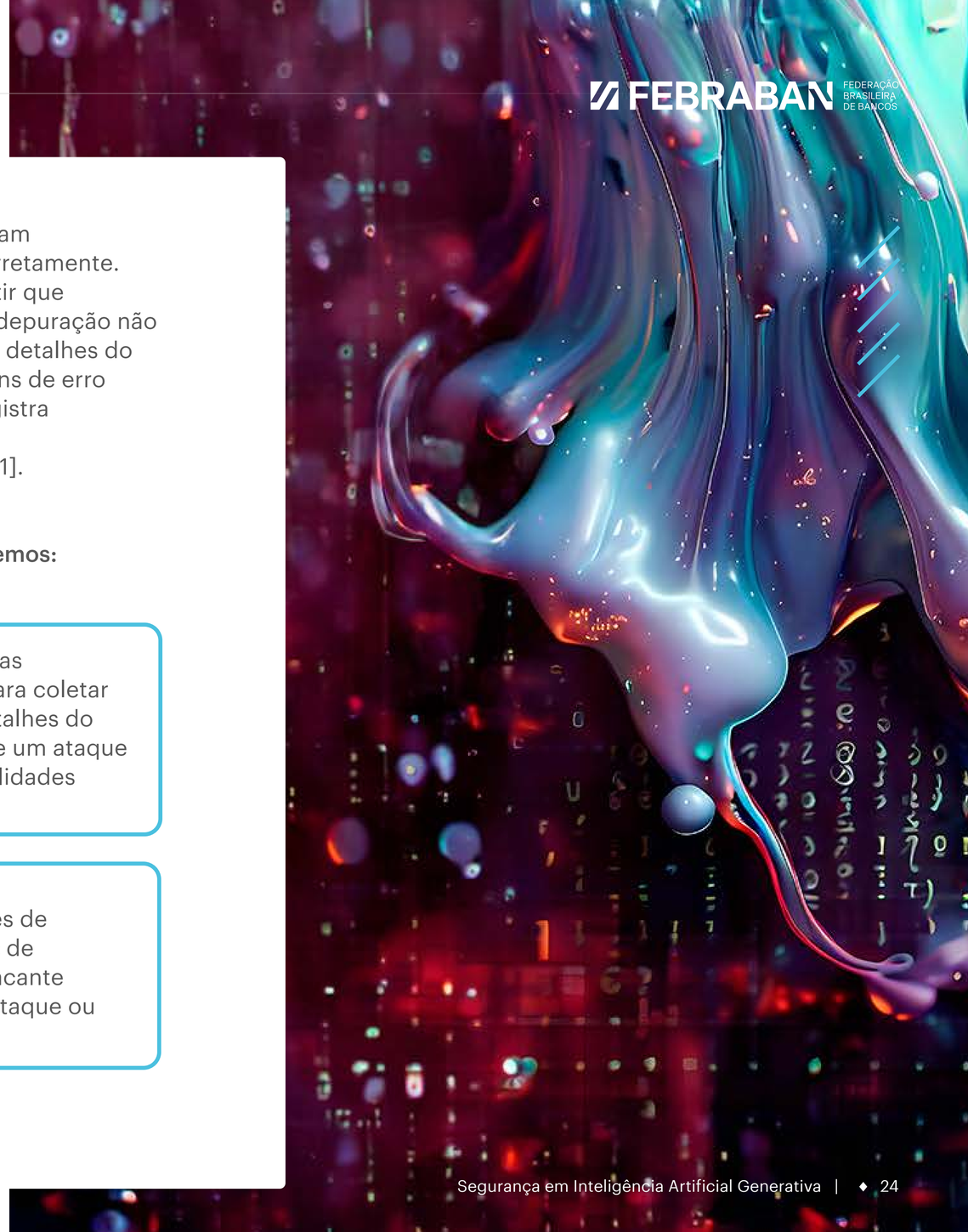
O tratamento inadequado de erros pode fornecer aos invasores informações valiosas sobre informações confidenciais, detalhes do sistema ou vetores de ataques em potencial para uma invasão. Estas informações podem ser utilizadas para planejar e lançar novos ataques. Controles de segurança inadequados podem tornar o sistema vulnerável a ataques. Além disso, o tratamento inadequado de erros também pode ter sérios efeitos colaterais, levando a vários problemas de segurança e proteção, como estado inconsistente, vazamento de informações, negação de serviços e outros [41].

Para atenuar essas vulnerabilidades, é importante implementar mecanismos adequados de tratamento de erros para garantir que os erros sejam detectados, registrados e tratados corretamente. Além disso, é muito importante garantir que mensagens de erro e informações de depuração não revelem informações confidenciais ou detalhes do sistema. Considere o uso de mensagens de erro genéricas para usuários, enquanto registra informações de erro detalhadas para desenvolvedores e administradores [41].

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um atacante explora as mensagens de erro de um LLM para coletar informações confidenciais ou detalhes do sistema, permitindo que ele lance um ataque direcionado ou explore vulnerabilidades conhecidas;

Cenário #2: um desenvolvedor acidentalmente deixa informações de depuração expostas no ambiente de produção, permitindo que um atacante identifique possíveis vetores de ataque ou vulnerabilidades no sistema.



Para se defender dessa vulnerabilidade, deve-se implementar mecanismos adequados de tratamento de erros, de modo a permitir que as falhas sejam detectadas, registradas e tratadas corretamente. Além disso, é necessário se certificar de que as mensagens de erro e as informações de depuração não revelem informações confidenciais ou detalhes do sistema. Por fim, deve-se revisar regularmente os logs de erros e tomar as ações necessárias para corrigir os problemas identificados, melhorando assim a estabilidade do sistema.

3.2.10. - *Training Data Poisoning*

Os ataques de envenenamento de dados ocorrem quando um invasor manipula os dados de treinamento para introduzir vulnerabilidades, *backdoors* ou vieses que podem comprometer a segurança, a eficácia ou o comportamento ético do modelo. Como exemplo, imagine um sistema que utiliza inteligência artificial para detectar anomalias na rede da organização. O invasor, sabendo da existência desse sistema, pode tentar introduzir dados falsos ao conjunto de dados de treinamento do modelo de AI, de modo que, eventualmente, a sua eficácia na detecção de atividades suspeitas diminua. Permitindo assim, que as atividades maliciosas do invasor se tornem indetectáveis pela ferramenta. Esse tipo de vulnerabilidade também é conhecida como distorção do modelo.

Os ataques de envenenamento de dados podem ser considerados um ataque de integridade porque a adulteração dos dados de treinamento afeta a capacidade do modelo de gerar previsões corretas. No entanto, é possível que tanto a confidencialidade, quanto a disponibilidade dos sistemas sejam afetados. Confidencialidade, decorrente do fato de que os invasores podem inferir informações potencialmente confidenciais sobre os dados de treinamento alimentando entradas para o modelo e disponibilidade, onde os invasores disfarçam suas entradas para enganar o modelo a fim de evitar a classificação correta [42].

O envenenamento de dados pode ser alcançado em um cenário de caixa preta contra ferramentas que dependem do *feedback* do usuário para atualizar seu aprendizado (como é o caso do ChatGPT) ou em um cenário de caixa branca em que o invasor obtém acesso ao modelo e seus dados de treinamento privados, possivelmente em algum lugar na cadeia de coleta de informações.

O principal problema com o envenenamento de dados é que não é fácil de corrigi-lo. Os modelos costumam ser retreinados com dados recém-coletados em determinados intervalos, dependendo do uso pretendido e da preferência do proprietário. Como o envenenamento geralmente ocorre ao longo do tempo e em alguns ciclos de treinamento, pode ser difícil dizer com precisão quando a previsão começou a mudar.

A reversão dos efeitos de envenenamento exigiria uma análise histórica demorada das entradas para a classe afetada, de modo a se identificar todas as amostras de dados ruins. Em seguida, essas amostras devem ser removidas e uma versão do modelo anterior ao início do ataque precisaria ser recuperada e retreinada. Ao lidar com grandes quantidades de dados e múltiplos vetores de ataques, não é incomum que os modelos nunca sejam corrigidos, pois esse processo de identificação e correção é bastante complexo.

Assim, dadas as dificuldades em corrigir modelos envenenados, os desenvolvedores de modelos precisam se concentrar em medidas que possam bloquear tentativas de ataque ou detectar entradas maliciosas antes que o próximo ciclo de treinamento aconteça. Para isso, os desenvolvedores podem empregar técnicas como a verificação de validade de entrada, testes de regressão, moderação manual das entradas de treinamento e uso de técnicas estatísticas para detecção de anomalias.

Por fim, é importante destacarmos que para realizar o envenenamento de dados, os invasores também precisam obter informações sobre como o modelo funciona, por isso é importante vazar o mínimo de informações possível e ter fortes controles de acesso para o modelo e os dados de treinamento. A esse respeito, as defesas de aprendizado de máquina estão vinculadas a práticas gerais de segurança, tais como restringir permissões, habilitar o controle de versão de arquivos e dados.

Como exemplo de cenários onde essa vulnerabilidade pode ser explorada, temos:

Cenário #1: um atacante se infiltra no *pipeline* de dados de treinamento e injeta dados mal-intencionados, fazendo com que o LLM produza respostas prejudiciais ou inadequadas;

Cenário #2: um usuário mal-intencionado compromete o processo de ajuste fino *(fine tunning),* introduzindo vulnerabilidades ou backdoors no LLM que podem ser exploradas posteriormente.

Para se defender dessa vulnerabilidade, é necessário garantir a integridade dos dados de treinamento obtendo-os através de fontes confiáveis e validando sua qualidade. Além disso, deve-se implementar técnicas robustas de sanitização e pré-processamento de dados para remover possíveis vulnerabilidades ou vieses nos dados de treinamento. Por fim, é necessário que se audite e revise regularmente os dados de treinamento do LLM, bom como, que se utilize mecanismo de monitoramento e alerta para detectar comportamentos incomuns que indiquem possível envenenamento do modelo.

3.3. - Ataques a sistemas de aprendizado de máquina

A ampla adoção de sistemas baseados em inteligência artificial em diversos setores da economia os torna alvos potenciais de ataques. É crucial que empreguemos medidas de segurança para proteger os usuários e essas ferramentas. Com o intuito de fornecer uma abordagem estruturada para entender e mapear os ataques relacionados ao aprendizado de máquina, o MITRE desenvolveu a matriz ATLAS (*ATLAS Machine Learning Threat Matrix*) [43]. Essa matriz, derivada do conhecido modelo ATT&CK do MITRE, oferece um conjunto de táticas, técnicas e sub-técnicas usadas por adversários para explorar vulnerabilidades em sistemas de aprendizado de máquina – *Machine Learning* (ML).

3.3.1. - Matriz Atlas

A matriz ATLAS organiza os ataques em torno de um conjunto de táticas, técnicas e sub-técnicas relacionadas a ML. As táticas representam o objetivo do adversário em realizar uma ação maliciosa, que pode ser, por exemplo, a persistência no sistema, a descoberta de informações, a movimentação lateral,

a execução de arquivos, a exfiltração de dados, etc. Por sua vez, as técnicas, que são agrupadas sob as táticas, representam como o adversário atinge seus objetivos táticos. Por exemplo, o usuário pode realizar um dump de senhas em busca de credenciais com privilégios que possam ser utilizadas posteriormente para movimentação lateral. Atualmente, o ATLAS encontra-se na versão 4.4.0 e conta com 1 matriz, 12 táticas, 40 técnicas, 27 sub-técnicas e 19 mitigações.

Além disso, o ATLAS ainda apresenta 17 casos de estudo. Por exemplo, o caso ML.CS0016, onde foi explorada uma vulnerabilidade no aplicativo MathGPT, que utiliza o modelo de linguagem GPT-3 para responder a perguntas matemáticas. Durante um processo de análise da ferramenta, identificou-se uma brecha de segurança que permite que usuários mal-intencionados manipulem os prompts fornecidos ao GPT-3, levando o modelo a comportamentos imprevistos. No incidente em questão, o ator obteve acesso não autorizado às variáveis de ambiente do sistema de hospedagem do MathGPT e à chave da API do GPT-3 usada pelo aplicativo. Além disso, o ator realizou um ataque de negação de serviço, prejudicando a disponibilidade da ferramenta.

Essa experiência demonstra a importância de considerarmos a segurança das aplicações de GenAI. Neste sentido, a matriz ATLAS, desenvolvida pelo MITRE, fornece uma estrutura valiosa para

Reconnaissance * 5 techniques	Search for Victim's Publicly Available Research Materials	Search for Publicly Available Adversarial Vulnerability Analysis	Search Victim-Owned Websites	Search Application Repositories	Active Scanning *		
Resource Development * 7 techniques	Acquire Public ML Artifacts	Obtain Capabilities *	Develop Adversarial ML Attack Capabilities	Acquire Infrastructure	Publish Poisoned Datasets	Poison Training Data	Establish Accounts *
Initial Access * 4 techniques	ML Supply Chain Compromise	Valid Accounts *	Evade ML Model	Exploit Public-Facing Application *			
ML Model Access 4 techniques	ML Model Inference API Access	ML-Enabled Product or Service	Physical Environment Access	Full ML Model Access			
Execution * 2 techniques	User Execution *	Command and Scripting Interpreter *					
Persistence * 2 techniques	Poison Training Data	Backdoor ML Model					
Defense Evasion * 1 techniques	Evade ML Model						
Discovery * 3 techniques	Discover ML Model Ontology	Discover ML Model Family	Discover ML Artifacts				
Collection * 3 techniques	ML Artifact Collection	Data from Information Repositories *	Data from Local System *				
ML Attack Staging 4 techniques	Create Proxy ML Model	Backdoor ML Model	Verify Attack	Craft Adversarial Data			
Exfiltration * 2 techniques	Exfiltration via ML Inference API	Exfiltration via Cyber Means					
Impact * 7 techniques	Evade ML Model	Denial of ML Service	Spamming ML System with Chaff Data	Erode ML Model Integrity	Cost Harvesting	ML Intellectual Property Theft	System Misuse for External Effect

* Indica uma adaptação de ATT&CK

Figura 2: Matriz ATLAS do MITRE. Origem: https://atlas.mitre.org.

entender e mapear ataques específicos a sistemas de ML. O caso apresentado acima exemplifica a aplicação da matriz ATLAS, identificando a tática de "Acesso Inicial" e a técnica de "Injeção de *Prompt*" usada pelo ator mal-intencionado. Assim, fica claro como a matriz ATLAS pode ser útil no combate a esses ataques, pois fornece uma estrutura simples para entender as táticas e técnicas utilizadas pelos adversários, permitindo que profissionais de segurança desenvolvam contramedidas e mitigações apropriadas. Além disso, a matriz ATLAS promove o compartilhamento de conhecimento e a colaboração entre a comunidade de segurança, auxiliando no avanço contínuo da defesa contra ameaças relacionadas ao ML.

3.3.2. - Mitigação de Ataques

A adoção da matriz ATLAS para o mapeamento e estudo dos ataques a sistemas de ML, além de possibilitar uma visão das técnicas e táticas que foram observadas durante o ataque, conta com uma relação de 19 medidas para mitigação que podem auxiliar no fortalecimento da segurança. A Tabela 1 mostra as medidas de mitigação que podem ser tomadas pelos usuários contra ataques realizados a sistemas de ML.

Tabela 1: Mitigações de Ataques Contra Sistemas de ML

Atlas ID	Nome	Descrição
AML.M0000	Liberação Limitada de Informações Públicas	Limite a divulgação pública de informações técnicas sobre o conjunto de aprendizado de máquina usado nos produtos ou serviços de uma organização
AML.M0001	Limitar a Liberação de Artefatos do Modelo	Limite a divulgação pública de detalhes técnicos do projeto, incluindo dados, algoritmos, arquiteturas de modelo e pontos de verificação do modelo que são usados em produção ou que são representativos daqueles usados em produção
AML.M0002	Ofuscação Passiva de Saída de ML	Reduza a fidelidade das saídas do modelo fornecidas ao usuário final pode reduzir a capacidade de adversários extrair informações sobre o modelo e otimizar ataques
AML.M0003	Fortalecimento do Modelo	Utilize técnicas para tornar os modelos de aprendizado de máquina robustos a entradas adversárias, como treinamento adversarial ou destilação de rede
AML.M0004	Restringir o Número de Consultas ao Modelo de Aprendizado de Máquina	Limite o número total e a taxa de consultas que um usuário pode realizar ao modelo
AML.M0005	Controlar o Acesso a Modelos de Aprendizado de Máquina e Dados em Repouso	Estabeleça controles de acesso nos registros internos dos modelos e limite o acesso interno aos modelos de produção. Limite também o acesso aos dados de treinamento apenas a usuários aprovados
AML.M0006	Utilizar Métodos de Conjunto	Utilize um conjunto de modelos para inferência para aumentar a robustez a entradas adversárias. Alguns ataques podem evitar efetivamente um modelo ou família de modelos, mas serem ineficazes contra outros
AML.M0007	Sanitizar Dados de Treinamento	Detecte e remova dados de treinamento adulterados. Os dados de treinamento devem ser sanitizados antes do treinamento do modelo e periodicamente para um modelo de aprendizado ativo. Implemente um filtro para limitar os dados de treinamento recebidos. Estabeleça uma política de conteúdo que remova conteúdos indesejados, como certos idiomas explícitos ou ofensivos
AML.M0008	Validar Modelo de Aprendizado de Máquina	Valide se os modelos de aprendizado de máquina funcionam como pretendido, testando possíveis gatilhos de backdoor ou com viés adversarial
AML.M0009	Utilizar Sensores Multimodais	Incorpore múltiplos sensores para integrar perspectivas e modalidades variadas a fim de evitar um único ponto de falha suscetível a ataques físicos

Tabela 1 (Continuação): Mitigações de Ataques Contra Sistemas de ML

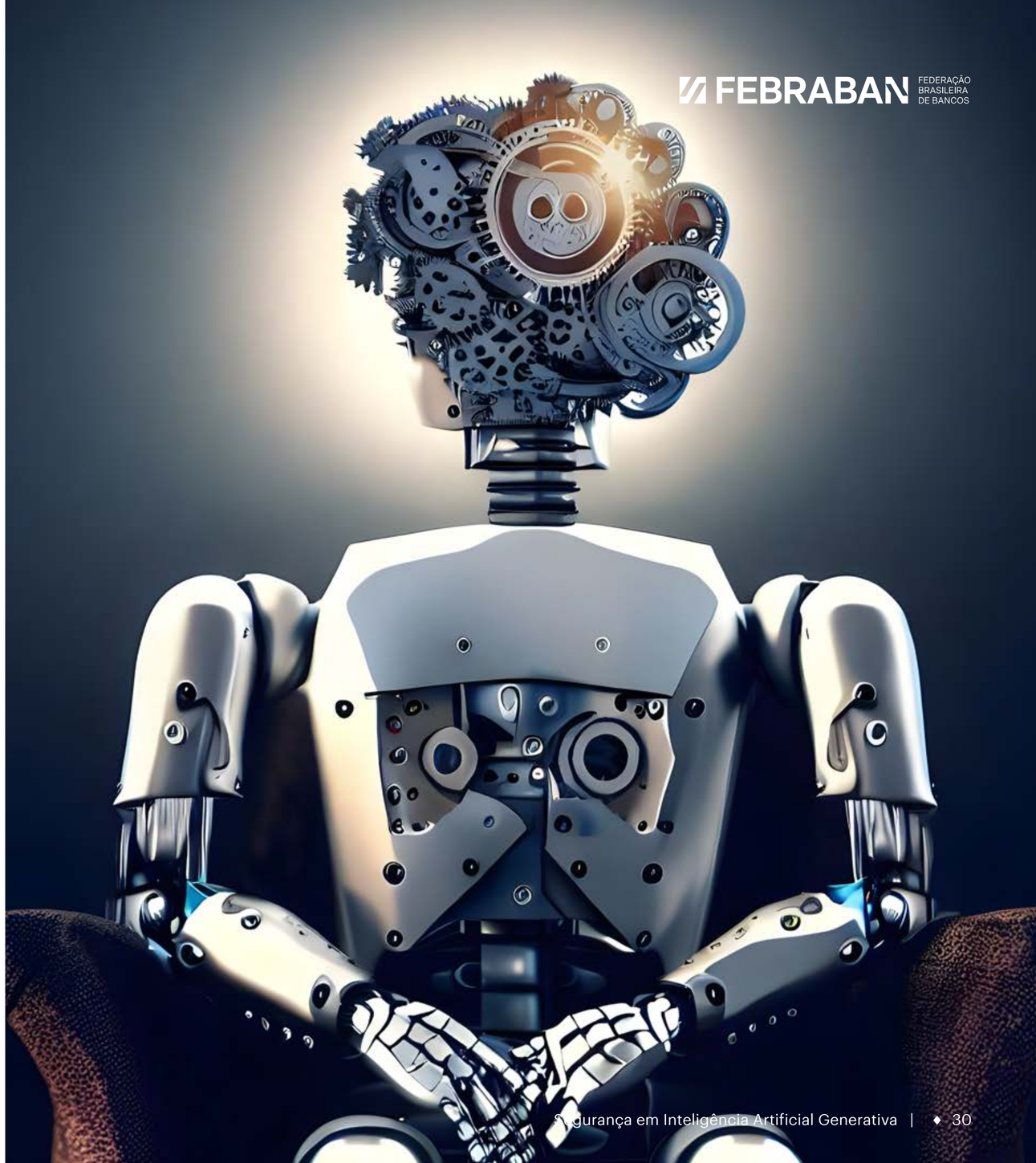
Atlas ID	Nome	Descrição
AML.M0010	Restauração de Entrada	Pré-processar todos os dados de inferência para anular ou reverter possíveis perturbações adversárias
AML.M0011	Restringir Carregamento de Bibliotecas	Evite o abuso de mecanismos de carregamento de bibliotecas no sistema operacional e software para carregar código não confiável, configurando mecanismos apropriados de carregamento de bibliotecas e investigando possíveis softwares vulneráveis. Formatos de arquivo, como arquivos pickle, comumente usados para armazenar modelos de aprendizado de máquina, podem conter vulnerabilidades que permitem o carregamento de bibliotecas maliciosas
AML.M0012	Encriptar Informações Sensíveis	Encripte dados sensíveis, como modelos de aprendizado de máquina, para proteger contra tentativas de acesso por parte de adversários
AML.M0013	Assinatura de Código	Imponha a integridade binária e de aplicativos com a verificação de assinatura digital para impedir a execução de código não confiável. Adversários podem incorporar código malicioso em software ou modelos de aprendizado de máquina. A aplicação da assinatura de código pode prevenir a comprometimento da cadeia de suprimentos de aprendizado de máquina e evitar a execução de código malicioso
AML.M0014	Verificar Artefatos de Aprendizado de Máquina	Verifique o checksum criptográfico de todos os artefatos de aprendizado de máquina para garantir que o arquivo não tenha sido modificado por um adversário
AML.M0015	Detecção de Entrada Adversária	Detecte e bloqueie entradas adversárias ou consultas atípicas que se desviem do comportamento benigno conhecido, exibam padrões de comportamento observados em ataques anteriores ou que provenham de IPs potencialmente maliciosos. Incorpore algoritmos de detecção adversária ao sistema de aprendizado de máquina
AML.M0016	Escaneamento de Vulnerabilidades	A varredura de vulnerabilidades é usada para encontrar vulnerabilidades potencialmente exploráveis em software para remediá-las. Formatos de arquivo, como arquivos pickle, comumente usados para armazenar modelos de aprendizado de máquina, podem conter vulnerabilidades que permitem a execução de código arbitrário
AML.M0017	Métodos de Distribuição do Modelo	Implantar modelos de aprendizado de máquina em dispositivos de borda pode aumentar a superfície de ataque do sistema. Considere servir modelos na nuvem para reduzir o nível de acesso que o adversário tem ao modelo
AML.M0018	Treinamento do Usuário	Eduque os desenvolvedores de modelos de aprendizado de máquina sobre práticas seguras de codificação e vulnerabilidades de aprendizado de máquina

4 CONSIDERAÇÕES FINAIS

Em conclusão, a GenAI pode apresentar riscos significativos quando consideramos ameaças como deepfakes, clonagem de vozes, criação de vídeos e modelos 3D realistas, além da produção de conteúdo criminoso e até mesmo o uso interno desta tecnologia por organizações mundo a fora. Além disso, vulnerabilidades também representam brechas de segurança que podem ser exploradas por criminosos, quando não tratadas ou negligenciadas durante a concepção de novos produtos e serviços.

Para mitigar estes riscos, é essencial investir em pesquisas avançadas, desenvolver técnicas de detecção e verificação de autenticidade, implementar regulamentações de proteção de dados e fortalecer a cooperação entre diferentes organizações. Somente com um esforço conjunto será possível enfrentar esses desafios e garantir a segurança e confiabilidade da GenAI. Além disso, a conscientização e a adoção de práticas de segurança adequadas são fundamentais para garantir a proteção dos sistemas e dos dados envolvidos.

Por último, recomendamos o uso da matriz ATLAS do MITRE, que é uma ferramenta valiosa para entender e mapear os ataques direcionados ao aprendizado de máquina, assim como um conjunto de mitigações que podem ser seguidas podem fortalecer os aspectos de segurança em sistemas que utilizam aprendizado de máquina. Sua organização estruturada em categorias bem definidas fornecem uma visão abrangente dos métodos utilizados pelos adversários, permitindo uma melhor compreensão das vulnerabilidades e o desenvolvimento de estratégias de defesa eficazes.



5 REFERÊNCIAS

[1] OpenAI, “ChatGPT: Optimizing language models for dialogue,” OpenAI, 30 Novembro 2022. [Online]. Available: https://openai.casa/blog/chatgpt/. [Acesso em 05 Julho 2023].

[2] Accenture, “TRUST in the GPT ERA: Governance, Assurance and Security for Generative Artificial Intelligence,” Accenture, 2023. [Online].

[3] Accenture, “Governance and Security of Generative AI,” Accenture, Maio 2023. [Online].

[4] Team8, “A CISOs Guide: Generative AI and ChatGPT Enterprise Risks,” Abril 2023. [Online]. Available: https://team8.vc/rethink/cyber/a-cisos-guide-generative-ai-and-chatgpt-enterprise-risks/.

[5] Ramesh, Aditya and Pavlov, Mikhail and Goh, Gabriel and Gray, Scott and Voss, Chelsea and Radford, Alec and Chen, Mark and Sutskever, Ilya, “Zero-shot text-to-image generation,” International Conference on Machine Learning, pp. 8821—8831, 2021.

[6] Chen, Mark and Tworek, Jerry and Jun, Heewoo and Yuan, Qiming and Pinto, Henrique Ponde de Oliveira and Kaplan, Jared and Edwards, Harri and Burda, Yuri and Joseph, Nicholas and Brockman, Greg and others, “Evaluating large language models trained on code,” arXiv preprint arXiv:2107.03374, 2021.

[7] Rae, Jack W and Borgeaud, Sebastian and Cai, Trevor and Millican, Katie and Hoffmann, Jordan and Song, Francis and Aslanides, John and Henderson, Sarah and Ring, Roman and Young, Susannah and others, “Scaling language models: Methods, analysis \& insights from training gopher,” arXiv preprint arXiv:2112.11446, 2021.

[8] Elhage, Nelson and Nanda, Neel and Olsson, Catherine and Henighan, Tom and Joseph, Nicholas and Mann, Ben and Askell, Amanda and Bai, Yuntao and Chen, Anna and Conerly, Tom and others, “A mathematical framework for transformer circuits,” Transformer Circuits Thread, vol. 1, 2021.

[9] M. Stefanini, M. Cornia, L. Baraldi, S. Cascianelli, G. Fiameni, and R. Cucchiara, “From Show to Tell: A Survey on Deep Learning-based Image Captioning,” arXiv:2107.06912, Novembro 2021.

[10] P. P. Liang, A. Zadeh, and L.P. Morency, “Foundations and Recent Trends in Multimodal Machine Learning: Principles, Challenges, and Open Questions,” arXiv:2209.03430 , Setembro 2022.

[11] Gokaslan, Aaron and Cohen, Vanya and Pavlick, Ellie and Tellex, Stefanie, “Openwebtext corpus,” 2019.

[12] Tom B. Brown and Benjamin Mann and Nick Ryder and Melanie Subbiah and Jared Kaplan and Prafulla Dhariwal and Arvind Neelakantan and Pranav Shyam and Girish Sastry and Amanda Askell and Sandhini Agarwal and Ariel Herbert-Voss and Gretchen Krueger and Tom H, “Language Models are Few-Shot Learners,” arXiv:2005.14165, 2020.

[13] Qiu, Xipeng and Sun, Tianxiang and Xu, Yige and Shao, Yunfan and Dai, Ning and Huang, Xuanjing, “Pre-trained models for natural language processing: A survey,” Science China Technological Sciences, vol. 63, pp. 1872—1897, 2020.

[14] Devlin, Jacob and Chang, Ming-Wei and Lee, Kenton and Toutanova, Kristina, “Bert: Pre-training of deep bidirectional transformers for language understanding,” arXiv preprint arXiv:1810.04805, 2018.

[15] Liu, Yinhan and Ott, Myle and Goyal, Naman and Du, Jingfei and Joshi, Mandar and Chen, Danqi and Levy, Omer and Lewis, Mike and Zettlemoyer, Luke and Stoyanov, Veselin, “Roberta: A robustly optimized bert pretraining approach,” arXiv preprint arXiv:1907.11692, 2019.

[16] Yang, Zhilin and Dai, Zihang and Yang, Yiming and Carbonell, Jaime and Salakhutdinov, Russ R and Le, Quoc V, “Xlnet: Generalized autoregressive pretraining for language understanding,” Advances in neural information processing systems, vol. 32, 2019.

[17] Brown, Tom and Mann, Benjamin and Ryder, Nick and Subbiah, Melanie and Kaplan, Jared D and Dhariwal, Prafulla and Neelakantan, Arvind and Shyam, Pranav and Sastry, Girish and Askell, Amanda and others, "Language models are few-shot learners," Advances in neural information processing systems, vol. 33, pp. 1877—1901, 2020.

[18] Zhang, Susan and Roller, Stephen and Goyal, Naman and Artetxe, Mikel and Chen, Moya and Chen, Shuohui and Dewan, Christopher and Diab, Mona and Li, Xian and Lin, Xi Victoria and others, "Opt: Open pre-trained transformer language models," arXiv preprint arXiv:2205.01068, 2022.

[19] Ouyang, Long and Wu, Jeffrey and Jiang, Xu and Almeida, Diogo and Wainwright, Carroll and Mishkin, Pamela and Zhang, Chong and Agarwal, Sandhini and Slama, Katarina and Ray, Alex and others, "Training language models to follow instructions with human feedback," Advances in Neural Information Processing Systems, vol. 35, pp. 27730—27744, 2022.

[20] Christiano, Paul F and Leike, Jan and Brown, Tom and Martic, Miljan and Legg, Shane and Amodei, Dario, "Deep reinforcement learning from human preferences," Advances in neural information processing systems, vol. 30, 2017.

[21] Stiennon, Nisan and Ouyang, Long and Wu, Jeffrey and Ziegler, Daniel and Lowe, Ryan and Voss, Chelsea and Radford, Alec and Amodei, Dario and Christiano, Paul F, "Learning to summarize with human feedback," Advances in Neural Information Processing Systems, vol. 33, pp. 3008—3021, 2020.

[22] Rombach, Robin and Blattmann, Andreas and Lorenz, Dominik and Esser, Patrick and Ommer, Bjorn, "High-resolution image synthesis with latent diffusion models," Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 10684—10695, 2022.

[23] Anantrasirichai, Nantheera and Bull, David, "Artificial intelligence in the creative industries: a review," Artificial intelligence review, pp. 1—68, 2022.

[24] Kietzmann, Jan and Paschen, Jeannette and Treen, Emily, "Artificial intelligence in advertising: How marketers can leverage artificial intelligence along the consumer journey," Journal of Advertising Research, vol. 58, pp. 263—267.

[25] Kandlhofer, Martin and Steinbauer, Gerald and Hirschmugl-Gaisch, Sabine and Huber, Petra, "Artificial intelligence and computer science in education: From kindergarten to university," IEEE frontiers in education conference (FIE), pp. 1—9, 2016.

[26] Ramesh, Aditya and Dhariwal, Prafulla and Nichol, Alex and Chu, Casey and Chen, Mark, "Hierarchical text-conditional image generation with clip latents," arXiv preprint arXiv:2204.06125, 2022.

[27] OWASP, "OWASP Top 10 for Large Language Model Applications," OWASP, 24 Maio 2023. [Online]. Available: https://owasp.org/www-project-top-10-for-large-language-model-applications/. [Acesso em 10 Julho 2023].

[28] Greshake, K and Abdelnabi, S and Mishra, S and Endres, C and Holz, T and Fritz, M, "Not what you've signed up for: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection," arXiv:2302.12173, Maio 2023.

[29] Grace, Katja and Salvatier, John and Dafoe, Allan and Zhang, Baobao and Evans, Owain, "When will AI exceed human performance? Evidence from AI experts," Journal of Artificial Intelligence Research, vol. 62, pp. 729—754, 2018.

[30] Sweeney, Latanya, "k-anonymity: A model for protecting privacy," International journal of uncertainty, fuzziness and knowledge-based systems, vol. 10, pp. 557—570, 2002.

[31] OpenAI, "OpenAI - Privacy policy," OpenAI, 23 Junho 2023. [Online]. Available: https://openai.com/policies/privacy-policy. [Acesso em 2023 Julho 2023].

[32] Strasser, Anna, “On pitfalls (and advantages) of sophisticated large language models,” arXiv preprint arXiv:2303.17511, 2023.

[33] Trend Micro, “Security Vulnerabilities of ChatGPT - Generated Code,” Trend Micro, 17 Maio 2023. [Online]. Available: https://www.trendmicro.com/en_us/devops/23/e/chatgpt-security-vulnerabilities.html. [Acesso em 10 Julho 2023].

[34] Hammond Pearce and Baleegh Ahmad and Benjamin Tan and Brendan Dolan-Gavitt and Ramesh Karri, “Asleep at the Keyboard? Assessing the Security of GitHub Copilot's Code Contributions,” arXiv 2108.09293, 2021.

[35] Grace, Katja and Salvatier, John and Dafoe, Allan and Zhang, Baobao and Evans, Owain, “When will AI exceed human performance? Evidence from AI experts,” Journal of Artificial Intelligence Research, vol. 62, pp. 729—754, 2018.

[36] Sevilla, Jaime and Heim, Lennart and Ho, Anson and Besiroglu, Tamay and Hobbhahn, Marius and Villalobos, Pablo, “Compute trends across three eras of machine learning,” International Joint Conference on Neural Networks (IJCNN), pp. 1—8, 2022.

[37] Bostrom, Nick, Superintelligence, Dunod, 2017.

[38] Bostrom, Nick, “Existential risk prevention as global priority,” Global Policy, vol. 04, nº 01, pp. 15—31, 2013.

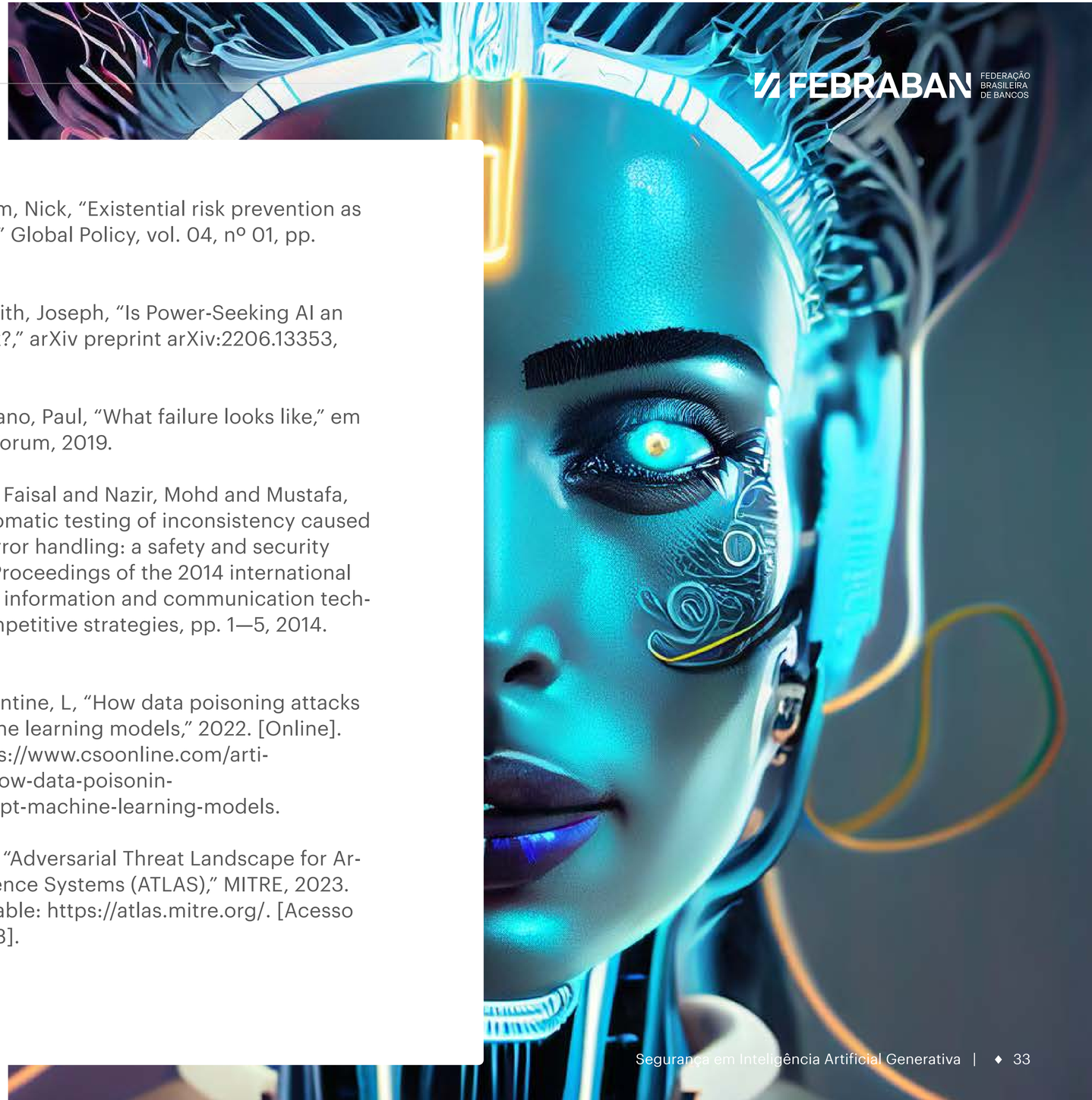
[39] Carlsmith, Joseph, “Is Power-Seeking AI an Existential Risk?,” arXiv preprint arXiv:2206.13353, 2022.

[40] Christiano, Paul, “What failure looks like,” em AI Alignment Forum, 2019.

[41] Anwer, Faisal and Nazir, Mohd and Mustafa, Khurram, “Automatic testing of inconsistency caused by improper error handling: a safety and security perspective,” Proceedings of the 2014 international conference on information and communication technology for competitive strategies, pp. 1—5, 2014.

[42] Constantine, L, “How data poisoning attacks corrupt machine learning models,” 2022. [Online]. Available: https://www.csoonline.com/article/3613932/how-data-poisoningattacks-corrupt-machine-learning-models.

[43] MITRE, “Adversarial Threat Landscape for Artificial-Intelligence Systems (ATLAS),” MITRE, 2023. [Online]. Available: https://atlas.mitre.org/. [Acesso em Junho 2023].



AUTORES

RENATO MARINHO

Chief Research Officer
at Morphus Labs

RAIMIR HOLANDA

Scientific Research
Leader at Morphus Labs

ANTÔNIO HORTA

Principal Cyber Research
Scientist at Morphus Labs

RODRIGO PARENTE

Project Research Leader
at Morphus Labs

MATEUS SANTOS

Threat Research Leader
at Morphus Labs

www.**morphus**.com.br