Avaliação e Mitigação de Ataques Adversários em Sistema de Detecção de Intrusão IoT

Antonia Mayara de A. da Silva[1], **Paulo Antonio Leal Rego**[1], **Michel Sales Bonfim**[1]

[1] Universidade Federal do Ceará - (UFC)
Fortaleza – CE – Brasil

mayaraalmeida@alu.ufc.br, paulo@dc.ufc.br, michelsb@ufc.br

Abstract. *Security in IoT networks still faces limitations, especially when attacks target deep learning-based intrusion detection systems (IDS). In this work, four adversarial attack techniques were applied to CNN, LSTM and GRU models, and their accuracy was evaluated after malicious samples were inserted. In a second step, the models were retrained with these samples, learning to recognize the patterns of the attacks, which is an important step in mitigating this type of threat in IoT environments.*

Resumo. *A segurança em redes IoT ainda enfrenta limitações, especialmente quando os ataques têm como alvo sistemas de detecção de intrusão (IDS) baseados em aprendizado profundo. Neste trabalho, quatro técnicas de ataques adversários foram aplicadas a modelos CNN, LSTM e GRU, e sua acurácia foi avaliada após a inserção das amostras maliciosas. Em um segundo momento, os modelos foram retreinados com essas amostras, aprendendo a reconhecer os padrões dos ataques, que é um passo importante para a mitigação desse tipo de ameaça em ambientes IoT.*

1. Introdução

A Internet das Coisas (IoT) tornou-se uma das tecnologias mais utilizadas atualmente. Essa expansão, combinada com a heterogeneidade e funcionalidade dessas redes, representa um grande desafio para os fabricantes: a segurança. Muitos dispositivos possuem recursos computacionais limitados e são responsáveis por transmitir grandes volumes de dados, tornando-se alvos atrativos para ataques cibernéticos. Assim, garantir a proteção contra diferentes tipos de ataques, como *Distributed Denial of Service* (DDoS), *malware* e ataques *zero day*, é essencial para a segurança das comunicações em redes IoT [Dos Santos et al. 2022].

Técnicas de aprendizado de máquina (ML) e profundo (DL) têm se mostrado eficazes na detecção de intrusões em redes IoT [Hussain et al. 2020]. Esses algoritmos analisam o tráfego de rede de forma autônoma, classificando rapidamente os dados e permitindo ações preventivas [Cavalcante et al. 2024]. O desenvolvimento de IDSs deve considerar baixa latência, limitação de recursos e escalabilidade. Segundo [Saheed et al. 2022], o aumento de dispositivos traz novos tipos de ataques, exigindo a evolução contínua dos IDS.

Como apontado em [Qin et al. 2020], o uso isolado de algoritmos pode ser insuficiente, dada a heterogeneidade das redes. Nesse cenário, a integração entre DL e programabilidade do plano de dados surge como alternativa para aprimorar a segurança.

A linguagem P4 (*Programming Protocol-independent Packet Processors*) permite personalizar o encaminhamento de pacotes e aplicar regras adaptáveis com alta flexibilidade e rapidez, além de integrar-se a técnicas de aprendizado para detectar e mitigar ataques.

Entretanto, a aplicação de DL em IDS também trouxe novos desafios, como os ataques adversários, que introduzem perturbações nos dados de entrada, induzindo falhas de classificação. Conforme destacado por [da Silva et al. 2024], esses ataques podem causar sérios danos às redes, sendo fundamental detectá-los e mitigá-los rapidamente. Este trabalho avalia o impacto de quatro técnicas adversárias sobre modelos CNN, LSTM e GRU. A acurácia dos modelos foi significativamente reduzida após a inserção das amostras maliciosas. Em seguida, os modelos foram retreinados com essas amostras, aprendendo a reconhecer os padrões adversários, um passo importante para a segurança de ambientes IoT.

2. Trabalhos Relacionados

A Maioria das pesquisas que exploram ataques adversários na classificação dos algoritmos de aprendizado se concentra no desenvolvimento de técnicas de ataques. [Novaes et al. 2021] apresentou um sistema para detecção e mitigação de ataques DDoS adversários. Utilizando treinamento adversário com dados gerados por *Generative Adversal Network* (GAN), o sistema foi testado em cenários SDN e no conjunto CICDDoS 2019. As métricas de acurácia, precisão, *recall* e F1 *score* foram analisadas, e o método proposto superou os algoritmos CNN, LSTM e MLP, alcançando acurácia entre 94% e 96%.

No trabalho de [Anthi et al. 2021], foram geradas amostras adversárias a partir de JSMA e FGSM para um *dataset* de casa inteligente, abrangendo cinco tipos de ataques. Algoritmos como *Decision Tree* (DT), RF, SVM e *Bayesian Network* apresentaram alto desempenho inicial, com precisão em torno de 99%, mas sofreram quedas significativas após a exposição a amostras adversárias. O retreinamento com as amostras aumentou a precisão, com a métrica F1 *score* superando 90%. Por fim, [Reddy et al. 2024] propôs a detecção e mitigação de ataques adversários em *switches* P4. Algoritmos como Regressão Logística, *Naive Bayes* (NB), DT e RF foram avaliados nos conjuntos CICIDS2017 e USB-IDS, com DT sendo selecionado para mapeamento em regras P4 devido à sua compatibilidade com restrições do plano de dados. Após a inclusão de dados adversários, o desempenho caiu, mas o retreinamento com dados sintéticos gerou melhorias significativas na precisão e mitigação dos ataques.

A pesquisa em defesas contra ataques adversários ainda é limitada, especialmente no uso de DL em ambientes IoT. Trabalhos existentes analisam poucos tipos de ataques e, em sua maioria, não tratam da mitigação. Apenas [Novaes et al. 2021] e [Reddy et al. 2024] propõem defesas, sendo que o último utiliza P4, mas avalia apenas uma técnica adversária. Este trabalho se diferencia ao empregar quatro técnicas diferentes de geração de amostras adversárias, utilizar essas amostras para retreinamento dos modelos de DL e aplicar P4 na mitigação com geração de regras. Além disso, expande o escopo de [Anthi et al. 2021], que usa apenas FGSM e JSMA em ataques DDoS, ao incluir ataques variados, como PGD, CW e GAN, o que permite avaliar a eficiência dos modelos diante de cenários mais realistas e no contexto de IoT.

3. Ataques Adversários

Neste trabalho, foram selecionados quatro algoritmos para geração dos ataques adversários: *Fast Gradient Sign Method* (FGSM), *Projected Gradient Descent* (PGD), *Generative Adversal Network* GAN e *Carlini and Wagner* CW. Cada um desses métodos possui características distintas e abordagens diferentes para a geração de ataques adversários.

A implementação das quatro técnicas adversárias foi feita em cada modelo individualmente. Para isso, foi utilizada uma máquina com sistema operacional Ubuntu 22.04.5 LTS, arquitetura de 64 bits, processador Intel Core i9-12900F da 12ª geração com 16 núcleos, 125.65 GiB de RAM, além de uma GPU NVIDIA GeForce RTX 3080 Ti.

3.1. Técnica FGSM

O FGSM gera amostras adversárias a partir do gradiente da função de perda em relação à entrada original, aplicando uma pequena perturbação controlada por ϵ. Na equação (1), essa perturbação é direcionada pelo sinal do gradiente, alterando a entrada de forma a enganar o modelo. Para este trabalho, ϵ foi definido como 0.2 após testes que buscaram equilibrar uma mudança significativa nos dados com a preservação da similaridade entre as amostras originais (normalizadas entre 0 e 1) e as adversárias.

$$X_{\text{adv}} = X + \epsilon \cdot \text{sign}(\nabla_X \mathcal{L}(f(X), y)) \tag{1}$$

3.2. Técnica PGD

A Equação do PGD, apresentada em (2), é uma extensão do FGSM, com atualizações iterativas que aplicam pequenas perturbações controladas por α e limitadas por ϵ. A função Clip garante que a nova amostra adversária permaneça próxima da original. Na implementação, foram definidos $\epsilon = 0{,}05$, $\alpha = 0{,}005$ e 40 interações, com amostras normalizadas entre 0 e 1. Esses valores foram escolhidos com base em testes, buscando ataques realistas e sem alterar drasticamente os dados originais.

$$X_{\text{adv}} = \text{Clip}_{X,\epsilon}\left(X + \alpha \cdot \text{sign}\left(\nabla_X \mathcal{L}(f(X), y)\right)\right) \tag{2}$$

3.3. Técnica CW

A técnica CW adota uma abordagem baseada em otimização com restrição, buscando a menor perturbação possível para enganar o modelo. A Equação (3) mostra que o objetivo é minimizar a distância $|X' - X|_2^2$ entre a entrada original e a adversária, balanceando com a função de perda do modelo, ponderada por um coeficiente c. Para os experimentos, foram utilizados 100 ciclos de iteração, $c = 1e{-}1$ e taxa de aprendizado de $1e{-}4$, equilibrando eficácia do ataque e proximidade com os dados originais.

$$X_{\text{adv}} = \arg\min_{X'} \left(\|X' - X\|_2^2 + c \cdot \mathcal{L}(f(X'), y)\right) \tag{3}$$

3.4. Técnica GAN

O ataque baseado em GAN envolve dois componentes centrais: o gerador, que cria amostras semelhantes às reais, e o discriminador, que tenta distinguir entre amostras reais e

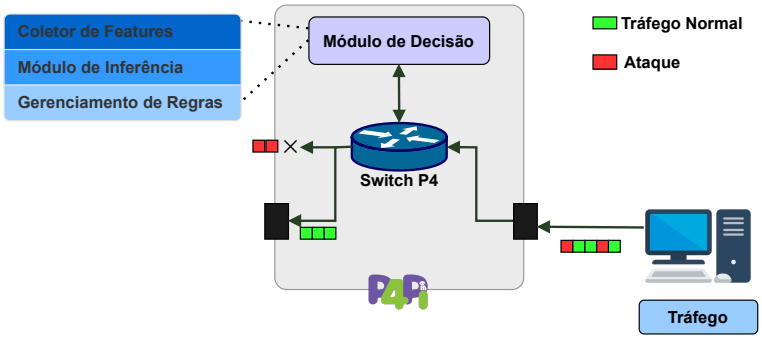
geradas. A técnica se baseia na minimização da perda adversária, conforme descrito na Equação (4), onde o gerador busca enganar o discriminador e este tenta maximizar sua capacidade de distinção. Para os experimentos, o espaço latente foi definido com dimensão 100. O gerador possui cinco camadas, e o discriminador, três. Ambas as redes utilizaram taxa de aprendizado de $1e{-}4$.

$$\min_G \max_D E_{x\sim p_{\text{real}}}[\log D(x)] + E_{z\sim p_{\text{noise}}}[\log(1 - D(G(z)))] \tag{4}$$

4. Implementação do IDS

A Figura 1 apresenta uma visão geral da arquitetura do IDS implementado. O tráfego, incluindo os ataques, é recebido por uma interface e repassado ao *switch* P4. Em seguida, os dados extraídos pelo *switch* são enviados ao módulo de decisão, no qual é composto pelo modelo de DL e o controlador responsável pela coleta das *features* e adição das regras. Caso o tráfego seja classificado como benigno, ele segue normalmente, mas é descartado se classificado como um ataque.

Figura 1. Arquitetura da solução



A arquitetura utilizada neste trabalho baseia-se em uma versão estendida da proposta apresentada em [da Silva et al. 2025], composta por módulos de coleta de *features*, classificação e gerenciamento de regras. O fluxo de pacotes e os componentes principais foram mantidos, com adaptações para incluir ataques adversários no processo de avaliação. O detalhamento do módulo de decisão, bem como a implementação do plano de dados com P4, pode ser consultado no trabalho anterior, onde esses componentes foram descritos.

4.1. Avaliação dos algoritmos

Os modelos de aprendizado profundo CNN, LSTM e GRU foram escolhidos para avaliação e, em seguida, implementação na Raspberry Pi 4, que oferece suporte à linguagem P4 por meio da plataforma P4Pi. Embora voltada inicialmente ao ensino, essa integração permitiu explorar a programabilidade do plano de dados em redes IoT.

O conjunto de dados utilizado foi o Edge-IIoTset[1], que contém mais de dez dispositivos IoT (como sensores de temperatura e frequência cardíaca) e 14 tipos de ataques

[1]https://ieee-dataport.org/documents/edge-iiotset-new-comprehensive-realistic-cyber-security-dataset-iot-and-iiot-applications

categorizados (como DoS, DDoS, malware, injeção e coleta de informações). A avaliação utilizou validação cruzada com cinco folds. O dataset continha 63,46% de tráfego normal e 36,54% de tráfego malicioso.

Os valores médios das métricas de treinamento são apresentadas na Tabela 1. Todos os algoritmos alcançaram cerca de 99% de acurácia, enquanto, na precisão, CNN superou GRU e LSTM, que marcaram 98,5% e 99,5%, respectivamente. Na métrica *recall*, LSTM apresentou leve vantagem sobre CNN (97%), e GRU atingiu 94%. No F1 *score*, CNN e LSTM mantiveram desempenho equilibrado, com o GRU alcançando 96,5%.

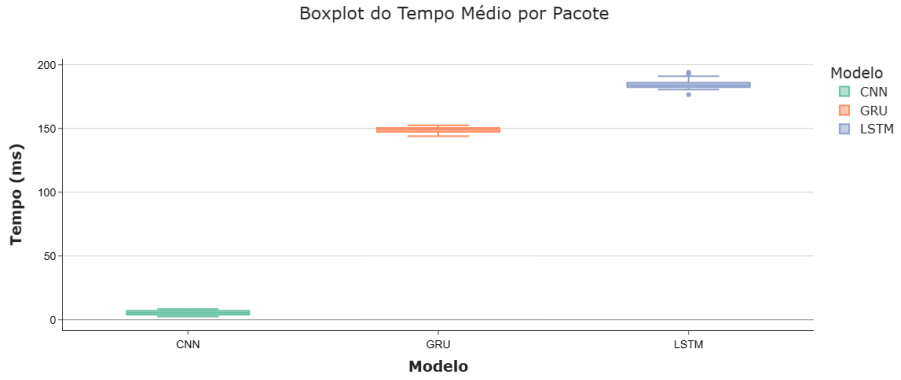
Tabela 1. Resultado do teste dos algoritmos

Algoritmo	**Acurácia (%)**	**Precisão (%)**	***Recall* (%)**	**F1 *score* (%)**
CNN	99,91	99,62	97,43	98,45
GRU	99,17	98,56	94,57	96,52
LSTM	99,82	99,51	97,63	98,38

De modo geral, os três algoritmos apresentaram desempenho semelhante nas etapas avaliadas. No entanto, o algoritmo CNN se destacou, obtendo os melhores resultados em três das quatro métricas analisadas, com exceção do *recall*, onde o LSTM apresentou desempenho superior. Por sua vez, o GRU registrou os menores valores em todas as métricas.

Com os algoritmos já na Raspberry Pi 4, foi analisado também o tempo de processamento dos *digests* até a inferência do modelo, com o objetivo de observar o impacto do módulo de inferência na rede. Durante a reprodução do tráfego, foi calculada a média do tempo necessário para classificar cada pacote. Com a utilização da CNN, o tempo médio por pacote foi de 3,66 milissegundos. Já com a GRU, o tempo médio foi de 149 milissegundos, enquanto na avaliação da LSTM, cada pacote levou, em média, 184 milissegundos para ser classificado. Na figura 2 é apresentado o tempo médio que cada algoritmo leva para a classificação.

Figura 2. Tempo para a inferência.



5. Avaliação dos Ataques Adversários e Retreinamento

Nesta seção são apresentados os resultados dos testes com os ataques adversários, além do retreinamento dos modelos com esses ataques.

5.1. Exposição aos ataques adversários

A Tabela 2 apresenta os resultados de cada algoritmo ao serem expostos aos diferentes ataques adversários. São apresentados os valores iniciais de acurácia dos modelos, bem como as acurácias após exposição aos diferentes ataques.

Tabela 2. Resultado do teste dos algoritmos

Acurácias	CNN	LSTM	GRU
Acurácia Inicial	99,91	99,82	99,17
Após ataque FGSM	61,57	76,18	72,79
Após ataque PGD	60,15	68,80	68,66
Após ataque CW	71,32	60,63	62,01
Após ataque GAN	51,10	57,12	53,42

Inicialmente, todos os modelos apresentavam acurácia próxima de 99%. No entanto, após a aplicação dos ataques adversários, observou-se uma queda significativa no desempenho. O ataque FGSM causou a maior queda no modelo CNN (61%), enquanto LSTM e GRU mantiveram resultados um pouco melhores (76% e 72%, respectivamente).

Com o ataque PGD, a acurácia continuou a cair, especialmente no CNN (60%). O LSTM e o GRU se saíram um pouco melhor, ambos com cerca de 68%. Já no ataque CW, o cenário se inverteu: o CNN foi o mais resiliente (71%), enquanto o LSTM teve o pior desempenho (60%), com o GRU um pouco mais acima (62%). O ataque mais agressivo foi o baseado em GAN, que apresentou as menores acurácias: 57% para o LSTM, 53% para o GRU e apenas 51% para o CNN.

5.2. Retreinamento e Mitigação

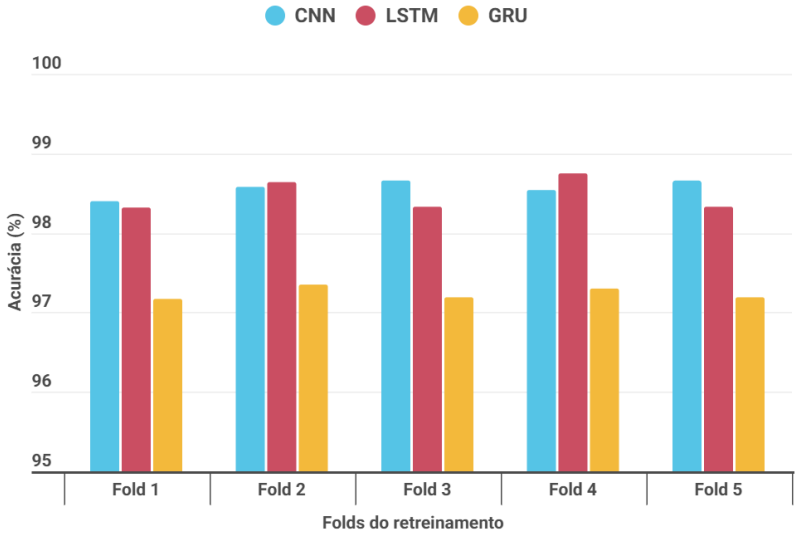
O retreinamento é uma etapa importante para garantir que o modelo seja capaz de identificar os ataques adversários e a depender do sistema, permitir a mitigação. Essa etapa com modelos CNN, LSTM e GRU foi realizada na mesma máquina da aplicação dos ataques adversários. As amostras foram divididas em ataque ou trafego normal, e os ataques adversários foram adicionados ao grupo de ataques para as etapas de treinamento e validação. A validação cruzada com 5 *folds* foi implementada para avaliação dos algoritmos e o *Batch size* foi como 128.

Os resultados obtidos no retreinamento são apresentados na Figura 3 e mostram que os algoritmos, apesar da redução considerável na acurácia ao serem expostos aos ataques, conseguem aprender rapidamente os padrões das amostras adversárias e melhorar seus desempenhos. O algoritmo CNN teve sua acurácia bastante afetada pelos ataques adversários, mas com a inclusão das amostras no conjunto de treinamento o algoritmo conseguiu um resultado médio de 98,57%.

Os algoritmos de rede recorrentes seguiram o mesmo padrão. Tanto o GRU como o LSTM conseguiram superar a baixa acurácia da avaliação dos ataques e obtiveram uma alta considerável na fase de retreinamento. Nesta etapa, o algoritmo GRU alcançou uma acurácia de 97,24% enquanto o LSTM obteve 98,44%.

A análise das métricas acurácia, precisão, *recall* e F1 *score* demonstrou que os algoritmos CNN e LSTM tiveram desempenho semelhante e superior ao GRU, que apresentou resultados inferiores devido à sua menor complexidade, limitando sua capacidade

Figura 3. Acurácia após o retreinamento.



de aprendizado. Nos testes de inferência , o CNN destacou-se por processar pacotes mais rapidamente, devido à sua arquitetura menos complexa.

Ao incluir ataques adversários nos testes e retreinamento, todos os algoritmos sofreram queda inicial de acurácia, mas o retreinamento permitiu recuperar resultados próximos aos iniciais. CNN e LSTM continuaram com desempenhos semelhantes, com ligeira vantagem para o CNN, enquanto o GRU permaneceu inferior aos outros. É importante destacar que os modelos são comprovadamente eficazes contra os quatro ataques avaliados. Como são ataques base e que abrangem diferentes técnicas adversárias, o algoritmo deve conseguir identificar também outros ataques adversários, já que é um modelo binário, e classifica os pacotes somente como normal ou malicioso. No entanto, para um resultado mais assertivo é necessário o retreinamento do algoritmo com a amostra adversária selecionada.

6. Conclusão

Os dispositivos IoT são alvos atraentes para ataques devido às limitações em sua fabricação e aplicação, portanto, implementar soluções de segurança não é tarefa fácil. O uso de soluções convencionais muitas vezes se torna ineficiente devido às exigências computacionais necessárias, à falta de adaptabilidade à variedade de dispositivos e aos padrões específicos de comunicação.

Entre os avaliados, o CNN mostrou-se mais adequado para este cenário, considerando que ele apresentou bons resultados de acurácia na primeira etapa, teve desempenho eficiente na Raspberry Pi com baixo consumo e tempo de inferência, e se mostrou capaz de se recuperar bem após o retreinamento frente aos ataques adversários. Já modelos como o LSTM e GRU, embora precisos, demandam maior tempo de processamento, o que dificulta sua adoção em redes IoT.

Os resultados da implementação das técnicas adversárias demonstram que deve existir uma preocupação maior com esse tipo de ataque, já que pode afetar drasticamente na assertividade dos modelos. As amostras adversárias foram geradas no formato CSV

e numpy, o que é ideal para o retreinamento. O código de cada técnica adversária e as amostras geradas estão disponíveis para a comunidade. No entanto, não foram geradas amostras no formato PCAP para a criação de um *dataset* totalmente adversário. A conversão para PCAP é uma possibilidade futura para permitir a análise direta do impacto dessas amostras adversárias no tráfego real da rede.

Referências

Anthi, E., Williams, L., Javed, A., and Burnap, P. (2021). Hardening machine learning denial of service (dos) defences against adversarial attacks in iot smart home networks. *Computers & Security*, 108:102352.

Cavalcante, J., Barros, T. G., and de Souza, J. N. (2024). Autonomous network intrusion detection for resource-constrained devices of the internet of things. In *Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg)*, pages 48–59. SBC.

da Silva, A. M. d. A., Bonfim, M. S., de Castro Callado, A., and Gonçalves, E. J. T. (2025). Detection and mitigation of attacks at the edge of iot networks using deep learning and p4. In *Simpósio Brasileiro de Sistemas de Informação (SBSI)*, pages 595–604. SBC.

da Silva, G. H. E., Junior, G. F., and Zarpelao, B. B. (2024). Impacto de ataques de evasão e eficácia da defesa baseada em treinamento adversário em detectores de malware. In *Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg)*, pages 829–835. SBC.

Dos Santos, B. V., Vergutz, A., Nogueira, M., and Macedo, R. T. (2022). Um método de ofuscação para proteger a privacidade no tráfego da rede iot. In *Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos (SBRC)*.

Hussain, F., Hussain, R., Hassan, S. A., and Hossain, E. (2020). Machine learning in iot security: Current solutions and future challenges. *IEEE Communications Surveys & Tutorials*, 22(3):1686–1721.

Novaes, M. P., Carvalho, L. F., Lloret, J., and Proença Jr, M. L. (2021). Adversarial deep learning approach detection and defense against ddos attacks in sdn environments. *Future Generation Computer Systems*, 125:156–167.

Qin, Q., Poularakis, K., and Tassiulas, L. (2020). A learning approach with programmable data plane towards iot security. In *2020 IEEE 40th International Conference on Distributed Computing Systems (ICDCS)*, pages 410–420. IEEE.

Reddy, S. S., Nishoak, K., Shreya, J., Reddy, Y. V., and Venkanna, U. (2024). A p4-based adversarial attack mitigation on machine learning models in data plane devices. *Journal of Network and Systems Management*, 32(1):5.

Saheed, Y. K., Abiodun, A. I., Misra, S., Holone, M. K., and Colomo-Palacios, R. (2022). A machine learning-based intrusion detection for detecting internet of things network attacks. *Alexandria Engineering Journal*, 61(12):9395–9409.