Large Language Models para detecção de conluios em licitações

Jorge N. Pavão[1], **Diego Brandão**[1], **Kele Belloze**[1]

[1]Programa de Pós-Graduação em Ciência da Computação (PPCIC)
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (Cefet/RJ)
Rio de Janeiro – RJ – Brasil

{kele.belloze}@cefet-rj.br

Abstract. *Identifying collusion in public procurement remains a persistent challenge, with various machine learning-based approaches being explored in the literature. This study investigates the application of Large Language Models (LLMs) in detecting collusion in procurement processes. We conducted experiments using prompt engineering and fine-tuning techniques, comparing the performance of the models with that of traditional algorithms such as Random Forest, Logistic Regression, and Support Vector Machine. The results showed that, although prompt engineering did not yield satisfactory outcomes, fine-tuning enabled the ChatGPT 4o Mini to outperform traditional algorithms on the analyzed datasets.*

Resumo. *A identificação de conluios em licitações públicas é um desafio persistente, com diversas abordagens baseadas em aprendizado de máquina sendo exploradas na literatura. Este trabalho investiga a aplicação de Large Language Models (LLMs) na detecção de indícios de conluio em licitações. Foram realizados testes utilizando técnicas de engenharia de prompt e fine-tuning, comparando o desempenho dos modelos com os de algoritmos tradicionais como Random Forest, Regressão Logística e Support Vector Machine. Os resultados demonstraram que, embora a engenharia de prompt não tenha alcançado resultados satisfatórios, o fine-tuning permitiu que o ChatGPT 4o Mini superasse os algoritmos tradicionais nos conjuntos de dados analisados.*

1. Introdução

O conluio em licitações ocorre quando empresas que deveriam competir entre si colaboram secretamente para manipular os resultados e elevar os preços ofertados (OCDE, 2021). Essa prática pode aumentar em até 25% o custo final de contratos públicos, gerando prejuízos significativos ao erário. A detecção automática de indícios de conluio tem sido objeto de diversas abordagens da ciência de dados e do aprendizado de máquina. No entanto, ainda persistem desafios importantes, como a escassez de dados rotulados para treinamento supervisionado e o desempenho limitado de algoritmos tradicionais, como *Random Forest*, Regressão Logística e *Support Vector Machines*, em determinados conjuntos de dados.

Nesse contexto, *Large Language Models* (LLMs) têm o potencial de mitigar a necessidade de grandes volumes de dados rotulados, além de capturar padrões complexos que podem escapar aos métodos convencionais. Apesar do sucesso recente dos LLMs

em tarefas de linguagem natural, sua aplicação no domínio da detecção de conluio — especialmente com dados predominantemente numéricos — ainda é incipiente e carece de validação empírica.

Este trabalho investiga o uso de LLMs, como o ChatGPT-4o Mini e o modelo O4 Mini da OpenAI, na identificação de indícios de conluio em processos licitatórios. Para esse propósito, duas estratégias de aplicação são avaliadas: engenharia de *prompts* e *fine-tuning*, cujos resultados são comparados com os reportados por Rodríguez et al. (2022) em um *benchmark* de modelos tradicionais. Além desta introdução, este artigo apresenta as seções de referencial teórico, trabalhos relacionados, metodologia, avaliação experimental e considerações finais.

2. Referencial Teórico

Large Language Models são modelos de aprendizado profundo treinados com grandes volumes de dados textuais para gerar e interpretar linguagem natural com elevado grau de sofisticação. Esses modelos utilizam, na maioria das vezes, arquiteturas baseadas em *Transformers* (Berryman and Ziegler, 2024), que permitem entender como diferentes partes de um texto se relacionam entre si, mesmo que estejam distantes. Isso ajuda o modelo a compreender melhor o significado completo das frases.

Duas das principais formas de adaptação dos LLMs a tarefas específicas são a engenharia de *prompt* e o *fine-tuning* do modelo. A engenharia de *prompt* consiste na criação e otimização de *prompts* que orientam o comportamento do modelo sem alterar seus parâmetros internos. Ela envolve a criação de instruções claras, concisas e eficazes para que o modelo produza a resposta desejada (Benzinho et al., 2024). Uma técnica comum é o *Few-Shot Prompting* que consiste em apresentar alguns exemplos no *prompt* para explicar ao modelo o que se deseja. Esses exemplos podem ser usados tanto para ensinar ao LLM o padrão de resposta desejado quanto para auxiliá-lo a interpretar as questões (Berryman and Ziegler, 2024).

O *fine-tuning* é uma abordagem que contempla um treinamento adicional de um LLM, com um conjunto de dados customizado, para adaptá-lo a tarefas ou domínios específicos (Anisuzzaman et al., 2025). Essa técnica tem se mostrado mais eficaz em tarefas que exigem alta precisão, especialmente em contextos onde os dados apresentam padrões difíceis de serem capturados apenas por meio de *prompts*.

3. Trabalhos relacionados

A identificação de conluio em licitações é um tema amplamente estudado, com diversas abordagens propostas ao longo dos anos. No entanto, até onde vai o conhecimento dos autores e com base na revisão da literatura realizada, não foram encontrados trabalhos que explorem o uso de LLMs especificamente para essa finalidade.

As abordagens mais comuns envolvem algoritmos tradicionais de aprendizado supervisionado e não supervisionado, redes neurais, análise de grafos, identificação de padrões frequentes e construção de variáveis estatísticas. Lima (2021) utilizou algoritmos como KNN, *Random Forest* e *Naïve Bayes* para identificar licitações suspeitas de conluio com base em dados textuais do Diário Oficial da União. De forma semelhante, Souza (2023) aplicou modelos supervisionados a diferentes fontes de dados, com destaque para

o bom desempenho de métodos do tipo *ensemble*, como *Extra Trees*, *Random Forest* e *AdaBoost*.

O uso de algoritmos tradicionais combinados com a criação de variáveis estatísticas (*screens*) é a abordagem adotada na maior parte dos trabalhos produzidos nos últimos anos (Scoralick et al., 2024; Wallimann et al., 2023; Huber et al., 2022; Silveira et al., 2022; Imhof and Wallimann, 2021; Huber and Imhof, 2019).

Além disso, alguns estudos combinaram algoritmos tradicionais com redes neurais. Lima (2021) testou arquiteturas como redes neurais profundas e bidirecionais, enquanto Souza (2023) empregou *Multi-Layer Perceptron*. Técnicas não supervisionadas também foram exploradas, como o uso de DBSCAN para identificar licitantes com endereços coincidentes (Velasco et al., 2021), e *Gaussian Mixture Models* para detectar agrupamentos anômalos sem o uso de dados rotulados (Silveira et al., 2023).

Este trabalho contribui para o tema ao avaliar e comparar o desempenho de LLMs — com técnicas de *prompt engineering* e *fine-tuning* — com os resultados obtidos por Rodríguez et al. (2022), aqui referido como artigo de referência. Esse artigo foi selecionado por sua ampla citação na área, pela aplicação de algoritmos reconhecidos de aprendizado de máquina (como SGD, *Random Forest*, *AdaBoost*, *Gradient Boosting*, dentre outros) e pelo compromisso com a reprodutibilidade ao disponibilizar publicamente seus dados e códigos. A comparação com essa base sólida permite avaliar de forma mais rigorosa o potencial dos LLMs na detecção de conluios em diferentes contextos nacionais.

4. Metodologia

Os LLMs avaliados neste estudo foram o ChatGPT-4o Mini e o O4 Mini, ambos da OpenAI. O ChatGPT-4o Mini é uma versão compacta do modelo GPT-4o, projetada para oferecer alto desempenho com baixo custo computacional, sendo ideal para tarefas cotidianas de automação, agentes inteligentes e interações rápidas em linguagem natural[1]. Essa versão foi escolhida em vez do ChatGPT-4o completo devido ao custo reduzido e ao desempenho competitivo em diversas tarefas. O modelo O4 Mini é voltado para tarefas mais complexas, que exigem maior capacidade de raciocínio. Ele adota a estratégia de *chain-of-thought*, que consiste em dividir o problema em etapas menores antes de gerar uma resposta final[2].

A Figura 1 apresenta uma visão geral da metodologia adotada. Dois experimentos foram conduzidos: o primeiro baseado em técnicas de engenharia de *prompt*, e o segundo utilizando *fine-tuning*. No primeiro experimento, ambos os modelos da OpenAI foram testados com a técnica de *Few-shot Prompting*, na qual exemplos anotados foram incorporados ao *prompt* para orientar o modelo na identificação de conluio. Foram avaliadas diferentes quantidades de exemplos e variações no formato de entrada, com o objetivo de verificar se seria possível obter desempenho competitivo sem necessidade de ajuste dos parâmetros internos dos modelos.

O segundo experimento envolveu a aplicação de *fine-tuning*, realizada apenas no ChatGPT-4o Mini, já que a OpenAI ainda não disponibiliza essa funcionalidade para o modelo O4 Mini. Dados rotulados foram utilizados para ajustar os parâmetros do modelo

[1]Disponível em: https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/
[2]Disponível em: https://openai.com/index/introducing-o3-and-o4-mini/

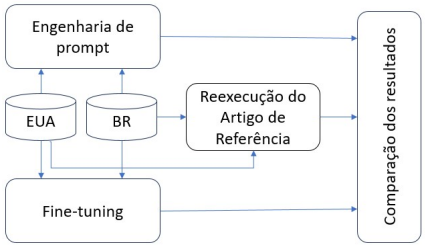

Figura 1. Etapas para a avaliação de LLMs em dados de licitações.

à tarefa de detecção de conluio, seguindo uma divisão estratificada dos dados em treino, validação e teste, respeitando a distribuição original das classes. O objetivo foi avaliar se a especialização do modelo por meio de ajuste fino proporcionaria desempenho superior aos métodos tradicionais de aprendizado de máquina.

Foram utilizados dois conjuntos de dados extraídos do artigo de referência: Brasil, com 683 registros, e Estados Unidos, com 7.004 registros. Ambos são compostos exclusivamente por atributos numéricos associados aos lances ofertados nas licitações. Esses conjuntos foram escolhidos por representarem os extremos de desempenho no estudo de Rodríguez et al. (2022) (melhor e pior resultado, respectivamente). Eles refletem um cenário de desbalanceamento, com predominância da classe negativa (ausência de conluio).

A métrica selecionada para comparação do desempenho foi a F1 Score, pois, no problema em análise, tão importante quanto o modelo identificar o máximo possível de ocorrências de conluio é o fato dele gerar o mínimo possível de falsos positivos. No artigo de referência, embora o código-fonte disponibilizado indique que a métrica F1 Score foi calculada, os resultados não foram explicitamente apresentados. Por este motivo, foi necessário reexecutar o código fonte.

5. Avaliação Experimental

Essa seção apresenta a avaliação dos LLMs utilizando as abordagens de engenharia de *prompt* e *fine-tuning*.

Experimento 1: Engenharia de *prompt*

Few-Shot Prompting foi a abordagem escolhida para avaliar se, com a engenharia de *prompt*, os LLMs seriam capazes de produzir resultados satisfatórios na detecção de conluio. Inicialmente, os dados foram divididos em treino e teste, sendo 80% para treino e 20% para teste. Dos dados de treino, foram selecionados dez registros classificados como conluio para serem apresentados ao LLM como exemplos. Os dados de teste foram todos enviados ao modelo para classificação.

Nas primeiras execuções com dados do Brasil, os modelos retornavam quantidades incorretas de respostas — por exemplo, 138 saídas para 137 entradas. Reduzindo o número de registros por *prompt*, as respostas passaram a ter a quantidade correta. Assim, os dados de teste foram divididos em *prompts* com no máximo 20 registros para classificação. Como os resultados com 10 exemplos de conluio foram insatisfatórios, aumentou-se para 100 exemplos, buscando melhorar o desempenho do modelo.

Em todos os experimentos, foram realizadas cinco rodadas de execução com dife-

rentes dados de treino e teste, e o resultado considerado foi a média dos valores obtidos. Para garantir que a diferença no desempenho dos LLMs avaliados não estaria sendo influenciada pelos dados apresentados como exemplo, cada rodada utilizou a mesma semente no momento da divisão dos dados de treino e teste e na seleção aleatória dos registros que seriam enviados como exemplo para os modelos. Ou seja, a rodada 1 da execução com o ChatGPT 4o Mini utilizou a mesma semente da rodada 1 com o modelo O4 Mini e o mesmo ocorreu nas outras rodadas. A Tabela 1 resume os valores de F1 Score obtidos nos testes com engenharia de *prompt*.

Tabela 1. Comparação do F1 score com uso de engenharia de *prompt*.

Dataset / Modelo	Artigo de referência	10 exemplos		100 exemplos	
		ChatGPT 4o Mini	O4-Mini	ChatGPT 4o Mini	O4-Mini
Brasil	0,794	0,337	0,704	0,378	0,656
Estados Unidos	0,301	0,283	0,302	0,284	0,292

Nota-se que ambos os modelos não apresentaram desempenho satisfatório. No conjunto de dados do Brasil, o ChatGPT 4o Mini obteve resultados inferiores, e, embora o modelo O4 mini tenha alcançado o valor de F1 de 0,704, ainda foi inferior ao melhor desempenho reportado no artigo de referência, obtido com o algoritmo *Gradient Boosting*. No conjunto de dados dos Estados Unidos, o resultado, apesar de similar ao obtido no artigo de referência, foi de apenas 0,302, o que é considerado insatisfatório.

Nos testes citados anteriormente, tanto os exemplos quanto os registros para classificação foram enviados no formato CSV, ou seja, separados por vírgula. Com o intuito de avaliar se outro formato dos dados poderia promover melhoria no desempenho dos modelos, o teste com 10 exemplos do conjunto de dados do Brasil foi repetido com as informações sendo enviadas no formato JSON. Identificou-se que o novo formato não resultou em F1 Scores mais elevados, indicando que, no caso em análise, o melhor formato é o CSV, pois produz resultados semelhantes, levemente superiores, com um menor custo por gerar *prompts* com uma quantidade muito menor de *tokens*.

Experimento 2: *Fine-tuning*

Como mencionado anteriormente, os experimentos com *fine-tuning* foram realizados exclusivamente com o modelo ChatGPT-4o Mini, gerando dois modelos distintos: um treinado com dados do Brasil e outro com dados dos Estados Unidos. O processo de ajuste fino foi conduzido por meio da funcionalidade oficial da OpenAI.

Para garantir uma comparação justa com os algoritmos tradicionais, utilizou-se todo o conjunto de treinamento disponível no *fine-tuning*, diferentemente dos testes com engenharia de *prompt*, que utilizaram apenas subconjuntos dos dados. A divisão seguiu dois estágios estratificados: inicialmente, separou-se 80% dos dados para treino e 20% para teste. Em seguida, os dados de treino foram novamente divididos (80% para o arquivo de treino e 20% para validação), sendo essas duas partes utilizadas no processo de ajuste fino. Os dados de teste foram reservados exclusivamente para a avaliação final.

De modo a trazer um nível maior de segurança de que os resultados obtidos refletem a real capacidade do modelo e não estão sendo influenciados por uma divisão específica nos dados, adotou-se um procedimento nos moldes da validação cruzada. Foram treinados três modelos para cada conjunto de dados, utilizando diferentes sementes na

divisão dos dados, e considerou-se como resultado final a média do F1 Score alcançado.

As inferências com os modelos ajustados foram feitas via API da OpenAI. Considerando a variabilidade típica das respostas dos LLMs, cada teste foi repetido cinco vezes, como também ocorreu nos experimentos com *prompting*. No entanto, diferentemente da etapa anterior, os *prompts* continham apenas um registro por vez, já que os modelos gerados não conseguiam processar múltiplas entradas simultaneamente.

Os resultados obtidos com o *fine-tuning* superaram amplamente tanto os alcançados por engenharia de *prompt* quanto os do artigo de referência. A Figura 2 apresenta os valores de F1 Score, comparando-os com os melhores resultados obtidos pelos algoritmos tradicionais no artigo de referência.

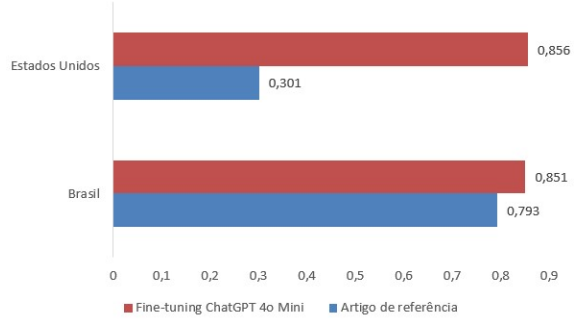

Figura 2. Comparação do melhor F1 Score usando *fine-tuning*.

Observa-se uma melhoria significativa em ambos os conjuntos de dados. Para os Estados Unidos, onde o desempenho do artigo de referência era insuficiente para aplicação prática, o *fine-tuning* elevou o F1 Score para 0,856. No conjunto do Brasil, o valor alcançou 0,851, indicando um nível de confiabilidade elevado para uso na detecção de conluios.

6. Considerações Finais

A partir desses resultados é possível concluir que, com o devido *fine-tuning*, os LLMs são capazes de lidar bem com dados numéricos e podem ser usados como ferramenta para auxiliar na identificação de conluios em licitações. Pode-se inferir também que essa capacidade é alcançada não somente com os modelos mais caros, uma vez que esses resultados foram obtidos com o ChatGPT 4o Mini, versão compacta e mais econômica do ChatGPT 4o. Essa característica é especialmente importante em situações em que o ajuste fino do modelo demande uma grande quantidade de registros.

Uma limitação deste trabalho foi o elevado custo de utilização de alguns LLMs, o que inviabilizou o uso de modelos como o ChatGPT 4o e o O3. Se, por um lado, essa limitação levou à concentração dos testes nas versões *mini* desses modelos - permitindo concluir que versões compactas e de menor custo podem apresentar bons resultados -, por outro, impediu a avaliação do potencial pleno dos LLMs.

Como continuação desse trabalho, pretende-se explorar novos conjuntos de dados e avaliar o desempenho de outros modelos conhecidos, como o Claude e o DeepSeek. O objetivo é verificar se a superioridade dos LLMs em relação aos algoritmos tradicionais de aprendizado de máquina se mantém ou se essa é uma característica apenas dos modelos da OpenAI.

Referências

Anisuzzaman, D., Malins, J. G., Friedman, P. A., and Attia, Z. I. (2025). Fine-tuning large language models for specialized use cases. *Mayo Clinic Proceedings: Digital Health*, 3(1):100184.

Benzinho, J. et al. (2024). Llm based chatbot for farm-to-fork blockchain traceability platform. *Applied Sciences*, 14(19).

Berryman, J. and Ziegler, A. (2024). *Prompt Engineering for LLMs: The Art and Science of Building Large Language Model–Based Applications*. O'Reilly Media, 1st edition.

Huber, M. and Imhof, D. (2019). Machine learning with screens for detecting bid-rigging cartels. *International Journal of Industrial Organization*, 65:277–301.

Huber, M., Imhof, D., and Ishii, R. (2022). Transnational machine learning with screens for flagging bid-rigging cartels. *Journal of the Royal Statistical Society. Series A: Statistics in Society*, 185:1074–1114.

Imhof, D. and Wallimann, H. (2021). Detecting bid-rigging coalitions in different countries and auction formats. *International Review of Law and Economics*, 68.

Lima, M. C. (2021). Deep vacuity: Detecção e classificação automática de padrões com risco de conluio em dados públicos de licitações de obras. Master's thesis, Universidade de Brasília.

OCDE (2021). Combate a cartéis em licitações no brasil: Uma revisão das compras públicas federais. Organização para a Cooperação e Desenvolvimento Econômico (OCDE).

Rodríguez, M. G. et al. (2022). Collusion detection in public procurement auctions with machine learning algorithms. *Automation in Construction*, 133.

Scoralick, L., Brandão, D., and Belloze, K. (2024). Aprimoramento de modelos para detecção de conluios em licitações públicas brasileiras com variáveis estatísticas e modelos explicáveis. In *Anais do XXXIX Simpósio Brasileiro de Bancos de Dados*, pages 680–686, Porto Alegre, RS, Brasil. SBC.

Silveira, D. et al. (2022). Won't get fooled again: A supervised machine learning approach for screening gasoline cartels. *Energy Economics*, 105.

Silveira, D. et al. (2023). Who are you? cartel detection using unlabeled data. *International Journal of Industrial Organization*, 88.

Souza, R. V. F. D. (2023). Identificação automática de conluio em pregões do comprasnet com aprendizado de máquina. Master's thesis, Universidade de Brasília.

Velasco, R. et al. (2021). A decision support system for fraud detection in public procurement. *International Transactions in Operational Research*, 28:27–47.

Wallimann, H., Imhof, D., and Huber, M. (2023). A machine learning approach for flagging incomplete bid-rigging cartels. *Computational Economics*, 62:1669–1720.