ATTACKS ON THIRD-PARTY APIS OF LARGE LANGUAGE MODELS

Wanru Zhao
University of Cambridge
wz341@cam.ac.uk

Vidit Khazanchi
Indian Institute of Technology, Bombay
viditk0812@gmail.com

Haodi Xing
University of Melbourne
xdk129@163.com

Xuanli He
University College London
xuanli.he@ucl.ac.uk

Qiongkai Xu*
Macquarie University
University of Melbourne
qiongkai.xu@mq.edu.au

Nicholas Donald Lane
University of Cambridge

ABSTRACT

Large language model (LLM) services have recently begun offering a plugin ecosystem to interact with third-party API services. This innovation enhances the capabilities of LLMs but introduces risks since these plugins, developed by various third parties, cannot be easily trusted. This paper proposes a new attacking framework to examine security and safety vulnerabilities within LLM platforms that incorporate third-party services. Applying our framework specifically to widely used LLMs, we identify real-world malicious attacks across various domains on third-party APIs that can imperceptibly modify LLM outputs. The paper discusses the unique challenges posed by third-party API integration and offers strategic possibilities to improve the security and safety of LLM ecosystems moving forward.

1 INTRODUCTION

Recently, the advances in Large Language Models (LLMs) (such as GPT (Brown et al., 2020; OpenAI et al., 2023), Gemini, and Llama (Touvron et al., 2023a;b), *etc.*) have shown impressive outcomes and are expected to revolutionize various industrial sectors, such as finance, healthcare and marketing. These models are capable of performing tasks, such as summarization, question answering, data analysis, and generating human-like content. Their proficiency in these areas makes them invaluable for enhancing work processes and supporting decision-making efforts.

Integrating these models into practical real-world applications presents several challenges. First, there is the hazard of the models relying on outdated information or generating content that is inaccurate or potentially misleading (Schick et al., 2023; Qin et al., 2023), a critical issue in fields where up-to-date data is essential, such as weather forecasting, news broadcasting, and stock trading. Furthermore, customizing these models to specialized domains, such as law or finance, demands extra domain-specific resources to meet precise requirements. Additionally, although LLMs may achieve expert-level performance in certain tasks, broadening their application across various domains or for complex reasoning tasks remains difficult (Wei et al., 2022). Enhancing their effectiveness often requires fine-tuning, retraining, or comprehensive instructions, which complicates their deployment and constrains their utility for tasks that require advanced skills.

To address these limitations, one strategy is to integrate third-party Application Programming Interfaces (APIs) with the LLMs. By accessing real-time information (Yao et al., 2022), conducting complex calculations (Schick et al., 2023), and executing specialized tasks such as image recognition (Patil et al., 2023; Qin et al., 2023), this integration broadens the functional scope of LLMs. It significantly boosts their efficiency and performance, enabling them to manage specialized tasks more adeptly without requiring bespoke training. For example, OpenAI's GPT Store significantly expands the operational capabilities of LLMs by hosting over 3 million custom ChatGPT variants.

*Corresponding author.

This enhancement is achieved by incorporating various plugins that facilitate third-party API calls, thereby integrating specialized functionalities developed by the community and partners.[1]

However, the integration of third-party APIs into LLMs introduces new security vulnerabilities by expanding the attack surface, which in turn provides more opportunities for exploitation by malicious actors. The reliability and security of these third-party services cannot be guaranteed, increasing the risk of data breaches and leading to unpredictable LLM behaviors. Furthermore, inadequate security measures in API integration can lead to mishandling data, compromising the integrity and security of the system. This paper explores the manipulation of LLM outputs through such external services, analyzing three attack methods across different domains. These attacks can subtly, and often imperceptibly, alter the outputs of LLMs. Our research highlights the urgent need for robust security protocols in the integration of third-party services with LLMs.

2 Proposed Pipeline

2.1 Overall Workflow

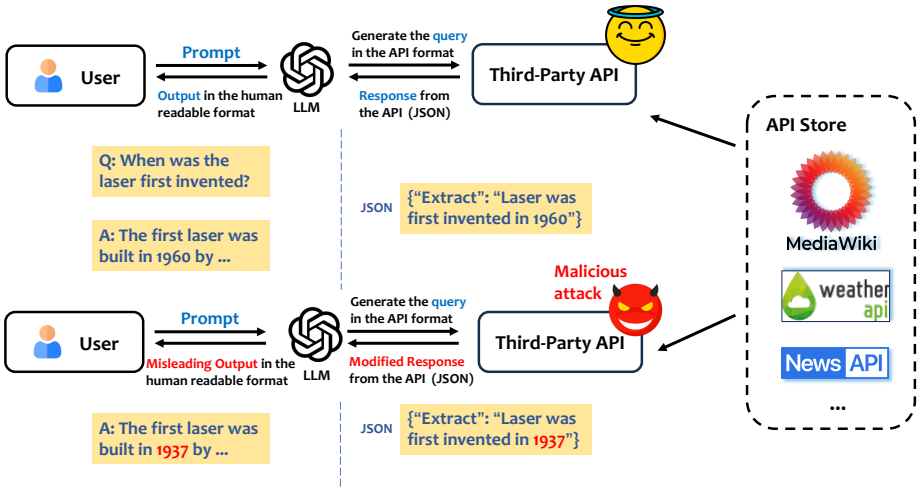

Figure 1: The workflow of third-party API attacks on Large Language Models.

Third-party APIs have become integral to setting up functionality and flexibility for LLMs. Figure 1 illustrates the workflow of calling third-party APIs in the plugin stores in a question-answering (QA) task. Users interact with the LLM Service Platform using natural language. The *questions* from the user side are first processed by the LLM, which then calls the corresponding third-party API to retrieve corresponding information on the internet. The third-party API outputs a *response* in JSON-format file based on the *query*, which is then processed by the LLM into a natural language response to the user interface of LLMs as the *answer*.

Nevertheless, there are also potential attacks that need to be paid attention to, as illustrated in the bottom part of Figure 1. Since the current LLMs service platform does not have a verification mechanism if the third-party API is maliciously attacked and key information is inserted, substituted, or deleted, which leads to key fields in the JSON-format output by the third-party API being maliciously manipulated. Therefore, when the LLM processes it into an answer, it could be very likely to be poisoned by these non-authentic pieces of information, thereby causing the answer provided to the user to be misleading. Such a process is invisible to the user or even LLMs. In the following subsections, we will detail the specific scenarios 2.2 and attack details 2.3.

[1] https://openai.com/blog/introducing-the-gpt-store

2.2 THIRD-PARTY API

WeatherAPI: Weather API [2] plays a crucial role in providing real-time global weather information to users, enabling them to stay informed about current weather conditions and forecasts. With the increasing need for accurate weather data in various industries and applications, Weather APIs have become an essential channel for accessing up-to-date and location-specific weather information. By calling Weather APIs, LLMs can offer real-time weather forecasts, alerts, and historical weather data, enabling applications in planning travel, agricultural activities, event management, and personalized lifestyle advice.

MediaWikiAPI: MediaWiki API [3] is developed based on the knowledge collected and managed by Wikipedia, which has been widely used by numerous websites and third-party groups. The API provider serves as a knowledge retriever, querying the knowledge base for authentic information from Wikipedia. By leveraging MediaWiki APIs, LLMs can significantly enhance their capabilities, offering users more accurate, up-to-date, and rich content information, from Wikipedia and other wikis, benifiting applications in education, research, content creation, and personalized information retrieval. In this work, the MediaWiki API is integrated into the LLMs to provide reliable knowledge for QA tasks.

NewsAPI: News API [4] provides real-time and enriched news content in a structured way. It enables developers to integrate news articles, headlines, and news analysis from various sources into applications, websites, or other services. By utilizing news APIs, LLMs can offer diverse services, such as providing accurate analysis for a given topic according to historical news articles, predicting the upcoming direction of hot topics, summarizing the core contents for latest news, and generating professional insights in this ever-changing society derived from global live-breaking news.

2.3 THREAT MODEL

This section outlines the methods used to manipulate API content, aiming to manipulate the outputs of the target LLMs accordingly.

- **Insertion-based Attack:** In insertion-based attacks, attackers embed adversarial content into API responses, leading to inaccurate, biased, or harmful LLM outputs.
- **Deletion-based Attack:** Deletion-based attacks manipulate the data processed by LLMs by omitting critical information from API responses. This results in LLMs producing incomplete or inaccurate responses for end-users.
- **Substitution-based Attack:** Substitution attacks manipulate critical data within API responses, replacing it with falsified content, thereby compromising the trustworthiness of LLMs. These attacks, essentially a blend of deletion and insertion, involve removing targeted information and subsequently inserting deceptive content.

3 EXPERIMENTS

3.1 EXPERIMENTAL SETUP

Models and Datasets We assess the susceptibility of LLMs to adversarial attacks through interactions with compromised third-party APIs. Our evaluation focuses on two prominent large language models: GPT-3.5-turbo (Brown et al., 2020) (version 0125) and Gemini (Team et al., 2023). The QA dataset used for MediaWiki is WikiQA (Yang et al., 2015), and for NewsAPI is NewsQA (Trischler et al., 2017). For the WeatherAPI selected questions based on weather from the WikiQA (Yang et al., 2015) have been used.

[2] https://weatherapi.com.
[3] https://mediawiki.org.
[4] https://newsapi.org.

Model	Modified Field	Deletion	Insertion	Substitution		
		ASR	ASR	Deletion	Insertion	ASR
GPT3.5-turbo	location	**93.10**	57.24	89.65	91.72	90.68
	temperature	86.67	60.33	93.33	90.33	**91.81**
	location + temperature	88.37	64.48	90.70	96.45	**93.48**
Gemini	location	**91.30**	73.53	86.95	93.52	90.12
	temperature	**100.0**	73.26	90.33	91.33	92.32
	location + temperature	**90.32**	79.08	90.32	89.10	89.70

Table 1: ASRs for WeatherAPI under third-party API attacks. We **bold** the highest ASR and underline the second highest ASR for each row.

APIs	Models	Deletion	Insertion	Substitution		
		ASR	ASR	Deletion	Insertion	ASR
MediaWiki API	GPT3.5-turbo	70.60	30.10	87.80	63.30	**73.56**
	Gemini	55.79	33.72	74.19	55.91	**63.77**
NewsAPI	GPT3.5-turbo	**91.98**	11.76	95.26	75.79	84.42
	Gemini	**80.69**	8.22	88.73	57.75	69.96

Table 2: ASRs for MediaWiki API and NewsAPI under third-party API attacks. We **bold** the highest ASR and underline the second highest ASR for each row.

Evaluation Metrics In evaluating the vulnerabilities of the third-party APIs integration into a question-answering system, we focus on measuring the Attack Success Rate (ASR) for insertion, deletion and substitution scenarios (for more details about the metrics, see Appendix B). These metrics demonstrate to what extent malicious modifications to data elements will affect the system output. A higher ASR indicates that the system is more vulnerable to the corresponding attacks.

Attack Details For WeatherAPI, the fields of interest are "*location*" and "*temperature*". These fields can be manipulated through three types of attacks: insertion, deletion, or substitution.
In MediaWiki API, our focus is on the "*DATE*" field, recognized by spaCy [5]. Our methodology involves three approaches: prefixing dates with the word "no" for insertion attacks, removing date entities for deletion attacks, and replacing dates with alternative entities in substitution attacks.
NewsAPI presents a different set of targeted entities, specifically those labeled as '*PERSON*', '*ORG*', and '*GPE*'. Similarly, we identify those entities using spaCy. These labels represent potential subjects of news articles. The insertion attack for NewsAPI aims to introduce three malicious entities, each corresponding to one of the targeted entity types. Substitution attacks replace these entities with the aforementioned malicious entities, while deletion attacks remove the entities entirely.
More details are listed in Appendix 3.

3.2 Experimental Results

WeatherAPI We first evaluate the vulnerability of WeatherAPI. Table 1 showcases the varying levels of susceptibility among models to different adversarial tactics. The bolded results indicate the highest ASR, while the underlined results represent the second highest ASR. Generally, LLMs have showen greater vulnerability to substitution attacks than deletion attacks, indicating they struggle more with processing misleading or incorrect data than with the absence of information. Insertion attacks, which entail embedding irrelevant data into the API responses, were less effective across all models, as indicated by lower ASRs. This suggests that such attacks are more challenging to execute successfully. However, even a moderate level of success in these attacks has significant implications for the reliability of models in real-world applications. Additionally, Gemini was shown to be more vulnerable compared to GPT3.5-turbo in all three types of adversarial attacks we have conducted.

[5] https://spacy.io.

MediaWiki and News APIs We present the results of MediaWiki and News APIs in Table 2. The insertion attack demonstrates significantly lower efficacy, especially with the News API. In contrast, the substitution and deletion attacks maintain high effectiveness, highlighting the LLMs' vulnerability to these attacks. Notably, the ASR difference between insertion and deletion underscores the greater challenge in embedding malicious content than removing it from LLM responses. The substitution attack, in particular, poses a greater threat to the MediaWiki API than the other two attacks. Further analysis reveals that within substitution attacks, deletion operations prove more effective than insertions, corroborating our findings on the performance of insertion and deletion attacks. For NewsAPIs, although the ASR for substitution attacks is lower than for deletion, it significantly exceeds that of insertion attacks.

3.3 DISCUSSION

Based on the experiment results, we analyze and summerize several factors that influence the attack performance as follows:

Conflict Knowledge Injection If the manipulated information contradicts the LLM's internal knowledge about weather patterns or locations, it might resist the attack, while LLMs might be more susceptible when they lack sufficient internal knowledge to identify inconsistencies. For example, removing location information makes it difficult for LLMs to identify the correct location, leading to a successful attack. However, if the remaining weather data is unique to a specific region, the LLM might still be able to make an accurate guess.

Reasoning Capabilities LLMs with better reasoning and filtering capabilities are more likely to resist attacks by identifying and disregarding inconsistencies, while LLMs with weaker capabilities are more susceptible to manipulation. For example, inserting a random temperature data point might be disregarded by the LLM, while subtly changing the temperature by a few degrees might be incorporated into the response.

Attack Quality and Precision The techniques used to conduct the attack can influence the experiment result. Since the NER is used to detect the victim entities, the attack performance can be affected by the performance of the NER used. For some cases, the NER provided by spaCy failed to detect the date information, which leads to the survival of the target date information in the LLMs answers. Also, third-party APIs provide large information to LLMs, which at the same time obstruct the attacker to conduct attacks precisely in a systematic manner. The key information to the question in the API response can be hard to locate and the various word types impede the success of manipulations. For example:
- Q: What can change the world, according to the activist?
- The information that is strongly related to the question's answer are some abstract expressions. These make it difficult for attackers to target key words that can influence the output of LLMs. In the provided case, "political commitment and financial support to achieve global development goals" is the benign LLM output according to the passages. The passages are in a conversational format, and the answer to the question is on an abstract level, compared to the questions that expect answers like people's names, dates, *etc.* This requires more effort for attackers to manipulate the output by tampering with the API response.

4 CONCLUSION

Our paper presents three attacks on third-party APIs integrated into the LLM ecosystems. This integration becomes more perilous as LLMs are increasingly equipped with APIs to better respond to user requests by accessing up-to-date information, performing complex calculations, and invoking external services through their APIs. It also opens up more possibilities for research in security within LLM ecosystems, extending beyond the isolated language models and APIs. Future work involves various attack methods, the design of defense mechanisms targeting third-party API attacks, and security concerns arising from multiple third-party API interactions.

REFERENCES

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Tianle Cai, Xuezhi Wang, Tengyu Ma, Xinyun Chen, and Denny Zhou. Large language models as tool makers, 2023.

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

Robert M French. The turing test: the first 50 years. *Trends in cognitive sciences*, 4(3):115–122, 2000.

Umar Iqbal, Tadayoshi Kohno, and Franziska Roesner. Llm platform security: Applying a systematic evaluation framework to openai's chatgpt plugins. *arXiv preprint arXiv:2309.10254*, 2023.

Qiao Jin, Yifan Yang, Qingyu Chen, and Zhiyong Lu. Genegpt: Augmenting large language models with domain tools for improved access to biomedical information, 2023.

Linyang Li, Ruotian Ma, Qipeng Guo, Xiangyang Xue, and Xipeng Qiu. Bert-attack: Adversarial attack against bert using bert. *arXiv preprint arXiv:2004.09984*, 2020.

Minghao Li, Yingxiu Zhao, Bowen Yu, Feifan Song, Hangyu Li, Haiyang Yu, Zhoujun Li, Fei Huang, and Yongbin Li. API-bank: A comprehensive benchmark for tool-augmented LLMs. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 3102–3116, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.187.

Yi Liu, Gelei Deng, Yuekang Li, Kailong Wang, Tianwei Zhang, Yepang Liu, Haoyu Wang, Yan Zheng, and Yang Liu. Prompt injection attack against llm-integrated applications. *arXiv preprint arXiv:2306.05499*, 2023.

OpenAI, :, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mo Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, Joanne Jang, Angela Jiang, Roger Jiang, Haozhun Jin, Denny Jin, Shino Jomoto, Billie Jonn, Heewoo Jun, Tomer Kaftan, Łukasz Kaiser, Ali Kamali, Ingmar Kanitscheider, Nitish Shirish Keskar, Tabarak Khan, Logan Kilpatrick, Jong Wook Kim, Christina Kim, Yongjik Kim, Hendrik Kirchner, Jamie Kiros, Matt Knight, Daniel Kokotajlo, Łukasz Kondraciuk, Andrew Kondrich, Aris Konstantinidis, Kyle Kosic, Gretchen Krueger, Vishal Kuo, Michael Lampe, Ikai Lan, Teddy Lee, Jan Leike, Jade Leung, Daniel Levy, Chak Ming Li, Rachel Lim, Molly Lin, Stephanie Lin, Mateusz Litwin, Theresa Lopez, Ryan Lowe, Patricia Lue, Anna Makanju, Kim Malfacini, Sam Manning, Todor Markov, Yaniv Markovski, Bianca Martin, Katie Mayer, Andrew Mayne, Bob McGrew, Scott Mayer McKinney, Christine McLeavey, Paul McMillan, Jake McNeil, David Medina, Aalok Mehta, Jacob Menick, Luke Metz, Andrey Mishchenko, Pamela Mishkin, Vinnie Monaco, Evan Morikawa, Daniel Mossing, Tong Mu, Mira Murati, Oleg Murk, David Mély, Ashvin Nair, Reiichiro Nakano, Rajeev Nayak, Arvind Neelakantan, Richard Ngo, Hyeonwoo

Noh, Long Ouyang, Cullen O'Keefe, Jakub Pachocki, Alex Paino, Joe Palermo, Ashley Pantuliano, Giambattista Parascandolo, Joel Parish, Emy Parparita, Alex Passos, Mikhail Pavlov, Andrew Peng, Adam Perelman, Filipe de Avila Belbute Peres, Michael Petrov, Henrique Ponde de Oliveira Pinto, Michael, Pokorny, Michelle Pokrass, Vitchyr Pong, Tolly Powell, Alethea Power, Boris Power, Elizabeth Proehl, Raul Puri, Alec Radford, Jack Rae, Aditya Ramesh, Cameron Raymond, Francis Real, Kendra Rimbach, Carl Ross, Bob Rotsted, Henri Roussez, Nick Ryder, Mario Saltarelli, Ted Sanders, Shibani Santurkar, Girish Sastry, Heather Schmidt, David Schnurr, John Schulman, Daniel Selsam, Kyla Sheppard, Toki Sherbakov, Jessica Shieh, Sarah Shoker, Pranav Shyam, Szymon Sidor, Eric Sigler, Maddie Simens, Jordan Sitkin, Katarina Slama, Ian Sohl, Benjamin Sokolowsky, Yang Song, Natalie Staudacher, Felipe Petroski Such, Natalie Summers, Ilya Sutskever, Jie Tang, Nikolas Tezak, Madeleine Thompson, Phil Tillet, Amin Tootoonchian, Elizabeth Tseng, Preston Tuggle, Nick Turley, Jerry Tworek, Juan Felipe Cerón Uribe, Andrea Vallone, Arun Vijayvergiya, Chelsea Voss, Carroll Wainwright, Justin Jay Wang, Alvin Wang, Ben Wang, Jonathan Ward, Jason Wei, CJ Weinmann, Akila Welihinda, Peter Welinder, Jiayi Weng, Lilian Weng, Matt Wiethoff, Dave Willner, Clemens Winter, Samuel Wolrich, Hannah Wong, Lauren Workman, Sherwin Wu, Jeff Wu, Michael Wu, Kai Xiao, Tao Xu, Sarah Yoo, Kevin Yu, Qiming Yuan, Wojciech Zaremba, Rowan Zellers, Chong Zhang, Marvin Zhang, Shengjia Zhao, Tianhao Zheng, Juntang Zhuang, William Zhuk, and Barret Zoph. Gpt-4 technical report, 2023.

Shishir G. Patil, Tianjun Zhang, Xin Wang, and Joseph E. Gonzalez. Gorilla: Large language model connected with massive apis. *arXiv preprint arXiv:2305.15334*, 2023.

Rodrigo Pedro, Daniel Castro, Paulo Carreira, and Nuno Santos. From prompt injections to sql injection attacks: How protected is your llm-integrated web application? *arXiv preprint arXiv:2308.01990*, 2023.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2227–2237, 2018. doi: 10.18653/v1/N18-1202.

Yujia Qin, Shihao Liang, Yining Ye, Kunlun Zhu, Lan Yan, Yaxi Lu, Yankai Lin, Xin Cong, Xiangru Tang, Bill Qian, Sihan Zhao, Lauren Hong, Runchu Tian, Ruobing Xie, Jie Zhou, Mark Gerstein, Dahai Li, Zhiyuan Liu, and Maosong Sun. Toolllm: Facilitating large language models to master 16000+ real-world apis, 2023.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. *CoRR*, abs/2302.04761, 2023.

Yifan Song, Weimin Xiong, Dawei Zhu, Wenhao Wu, Han Qian, Mingbo Song, Hailiang Huang, Cheng Li, Ke Wang, Rong Yao, Ye Tian, and Sujian Li. Restgpt: Connecting large language models with real-world restful apis, 2023.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models, 2023a.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh

Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. Llama 2: Open foundation and fine-tuned chat models, 2023b.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. NewsQA: A machine comprehension dataset. In Phil Blunsom, Antoine Bordes, Kyunghyun Cho, Shay Cohen, Chris Dyer, Edward Grefenstette, Karl Moritz Hermann, Laura Rimell, Jason Weston, and Scott Yih (eds.), *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pp. 191–200, Vancouver, Canada, August 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-2623.

Boxin Wang, Weixin Chen, Hengzhi Pei, Chulin Xie, Mintong Kang, Chenhui Zhang, Chejian Xu, Zidi Xiong, Ritik Dutta, Rylan Schaeffer, et al. Decodingtrust: A comprehensive assessment of trustworthiness in gpt models. *arXiv preprint arXiv:2306.11698*, 2023.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837, 2022.

Kangxi Wu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. LLMDet: A third party large language models generated text detection tool. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 2113–2133, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.139.

Xilie Xu, Keyi Kong, Ning Liu, Lizhen Cui, Di Wang, Jingfeng Zhang, and Mohan Kankanhalli. An llm can fool itself: A prompt-based adversarial attack. *arXiv preprint arXiv:2310.13345*, 2023.

Yi Yang, Wen-tau Yih, and Christopher Meek. WikiQA: A challenge dataset for open-domain question answering. In Lluís Màrquez, Chris Callison-Burch, and Jian Su (eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 2013–2018, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1237.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*, 2022.

Yifan Yao, Jinhao Duan, Kaidi Xu, Yuanfang Cai, Eric Sun, and Yue Zhang. A survey on large language model (llm) security and privacy: The good, the bad, and the ugly. *arXiv preprint arXiv:2312.02003*, 2023.

A Related Work

Large Language Models Enabling machines to understand and communicate human languages has been a long-standing challenge. A machine is believed to be intelligent by researchers if it passes the Turing Test (French, 2000), which is a deceptive test of determining the grasp of human intelligence by distinguishing whether a human or a machine generates the output. Language models (LMs) are one of the major techniques to model the generation of human languages based on the likelihood of words and sequences and make predictions of either the masked tokens in word sequences or future tokens. Large language models (LLM) are the most state-of-the-art LM that draw extensive attention due to their powerful capabilities. It scales the pre-trained LM such as BERT (Devlin et al., 2018), ELMo (Peters et al., 2018) in terms of the training data size and model size. These transformer-based LMs offer a great improvement in advancing machine intelligence for various downstream tasks, which introduces transformative changes in both industry practices and daily use domains by providing high-quality responses to the user according to the given context.

Third-Party API Integrating third-party APIs with LLMs has been a pivotal advancement in AI, enabling these models to significantly extend their capabilities and applicability in the real world. This combination of recent works highlights a two-fold emphasis on broadening abilities and tackling emerging challenges. Recent works such as Toolformer (Schick et al., 2023), ToolLLM (Qin et al., 2023), API-Bank (Li et al., 2023), and RestGPT (Song et al., 2023) demonstrate the potential of LLMs to autonomously leverage external tools and APIs, thus broadening their operational scope across various domains. LATM (Cai et al., 2023), which enables LLMs to create their own tools, and GeneGPT's (Jin et al., 2023) domain-specific applications illustrate the expanding problem-solving capacities and efficiencies of LLMs. The integration process also brings significant security and ethical considerations to the forefront. Efforts by Gorilla (Patil et al., 2023) and LLMDet (Wu et al., 2023) to refine the precision of API calls and identify model-generated content underscore the critical need for mechanisms that ensure the responsible use of AI technologies. These contributions emphasize the importance of developing robust frameworks to mitigate risks associated with misinformation, misuse, and data privacy in deploying LLMs with third-party APIs.

Attacks The surprisingly advanced performance of LLMs provides users with various high quality services, even security vulnerabilities detection for code repository (Yao et al., 2023). However, the security concerns should be emphasised. Li et al. (2020) proposed to use another LM BERT to mislead other deep neural models. The security risk of LLM is stressed by many prompt-based adversarial attacks. These attacks are mainly conducted by injecting pre-constructed prompts to LLMs in order to deceive LLM (Liu et al., 2023; Xu et al., 2023). Apart from that, Wang et al. (2023) attacks the performance of LLMs regarding the classical text classification task for GLUE dataset by leveraging an adversarial GLUE dataset AdvGLUE++. Pedro et al. (2023) evaluates the vulnerabilities of LLM-integrated web applications under the attacks which executes SQL injection through prompt injection. In addition, The security and privacy issues of OpenAI's ChatGPT plugins are systematically assessed by Iqbal et al. (2023), with their constructed framework applied.

B Evaluation Metrics Description

We present the calculation of ASR for each attack below:

- **ASR for Insertion:** This metric quantifies the attack's ability to integrate additional, misleading information into the model's responses. It is calculated by tracking the instances where extraneous elements are successfully inserted into the model's output and fail to be recognized as such by the model.

$$\text{ASR} = \frac{\text{\# of Successful Insertions}}{\text{\# of Valid Instances}}$$

- **ASR for Deletion:** This metric measures the attack's success in removing crucial information from the input the model fails to recall or compensate for in its response. It is determined by the ratio of instances where essential information is effectively deleted, and the model does not identify or correct the omission.

$$\text{ASR} = \frac{\text{\# of Successful Deletions}}{\text{\# of Valid Instances}}$$

- **ASR for Substitution:** ASR for substitution is conceived to offer a balanced evaluation of the attack's overall ability to manage both the addition and omission of information. It is defined as the harmonic mean of ASRs for insertion and deletion, similar to the F1-Score used in statistical analysis for measuring a test's accuracy.

$$\text{ASR} = \frac{2 * \text{InsertASR} * \text{DeleteASR}}{\text{InsertASR} + \text{DeleteASR}}$$

For all attacks, a higher ASR indicates a more effective attack, demonstrating the model's vulnerability to the corresponding attacks.

C RESPONSE FORMAT

Sample WeatherAPI Response


```
{
  "location": {
    "name": "London",
    "region": "City of London, Greater London",
    "country": "United Kingdom",
    "lat": 51.52,
    "lon": -0.11,
    "localtime": "2021-02-21 8:42"
  },
  "current": {
    "temp_c": 11,
    "temp_f": 51.8,
    "is_day": 1,
    "condition": {
      "text": "Partly cloudy",
    },
    "wind_mph": 3.8,,
    "pressure_in": 30.3,
    "precip_in": 0,
    "humidity": 82,
    "air_quality": {
      "co": 230.3,
      "no2": 13.5,
    }
  }
}
```


Sample MediaWikiAPI Response


```
{
    "batchcomplete": "",
    "query": {
        "pages": {
            "368118": {
                "pageid": 368118,
                "ns": 0,
                "title": "Madden NFL",
                "extract": "Madden NFL (known as John Madden Football
                    until 1993) is an American football sports video game
                     series developed by EA Tiburon for EA Sports. The
                    franchise, named after Pro Football Hall of Fame
                    coach and commentator John Madden, has sold more than
                     130 million copies as of 2018. Since 2004, it has
                    been the only officially licensed National Football
                    League (NFL) ..."
            }
        }
    }
}
```

Sample NewsAPI Response


```
{
    "storyId": "./cnn/stories/f382e1ca273b84cf5041d9ea589cd6d8c4651089.
        story",
    "text": "(CNN) -- A South Florida teenager accused of killing and
        mutilating 19 cats excitedly described to police how he dissected
         cats in class, and where to find cats for experimentation,
        according to police.\n\n\n\nTyler Weinman laughed when police
        told him they had information he was the cat killer, an arrest
        document said.\n\n\n\nWhen Miami-Dade police told Tyler Hayes
        Weinman someone was killing cats in the neighborhood..."
}
```


D MODIFICATION RULES

	Entity Label	**Insertion**	**Substitution**	**Deletion**
Weather API	location	Introduce 'not' before	location: "Sydney"	Directly remove entities
MediaWiki API	DATE	Introduce 'not' before	Years: 1937, Dates: 1	Directly remove entities
NewsAPI	ORG | GPR | PERSON	PERSON & ORG: Insert 'and Taishan' after GPE: Insert 'and Melbourne' after	PERSON & ORG: 'Taishan' GPR: 'Melbourne'	Directly remove entities

Table 3: Entities and attack rules for the insertion, substitution, and deletion.

E FAILED EXPERIMENTS EXAMPLES

Questions	**Documents**	**Original Answers**	**Modified Answers**
what year did peyton manning get drafted	Peyton Williams Manning (born March 24, 1976) is an American former football quarterback who played in the National Football League (NFL) for 18 seasons. Nicknamed ... he won the Maxwell, Davey O'Brien, and Johnny Unitas Golden Arm Awards as a senior en route to victory in the 1997 SEC Championship. ...	Peyton Manning was drafted in the year 1998.	Peyton Manning was drafted in the year 1998.
when the wind blows james Patterson	—	The response from the API indicates that "When the Wind Blows" is a novel written by James Patterson.	The book "When the Wind Blows" is a novel written by James Patterson.

Table 4: Samples of questions, retrieved documents, original answers, and modified answers. "—" indicates no documents found.