International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

SCRAPPING DATA FROM GOOGLE MAPS USING PYTHON

Mrs.RANJITHAKALA KAKOLLU[1], EDE DURGA RAJYA VEERAMMA[2], CHITTIBOINA MOHIT[3], KOLLI LAHARSHITHA REDDY[4], JANNU VINAY VARDHAN[5]

[1]Associate Professor, Dept. of CSE, V.K.R, V.N.B & A.G.K COLLEGE OF ENGINEERING,GUDIVADA

[2345]UG Students, Dept. of CSE,V.K.R, V.N.B & A.G.K COLLEGE OF ENGINEERING, GUDIVADA

ABSTRACT

Web scraping has become an essential technique for collecting large-scale data from online platforms for research, analytics, and business intelligence. This project focuses on scraping publicly available business information from **Google Maps** using Python. Google Maps contains extensive data such as business names, addresses, contact details, ratings, reviews, and geographic coordinates, which can be useful for market analysis, competitor research, location intelligence, and trend analysis.The proposed system utilizes Python libraries such as **`requests`, `BeautifulSoup`,** and browser automation tools like `Selenium` to extract structured information from dynamically loaded web pages. Since Google Maps uses JavaScript to render content, automated browser control is implemented to simulate user interaction and retrieve dynamic data. The collected data is parsed, cleaned, and stored in structured formats such as CSV or Excel files for further analysis. The system emphasizes automation, scalability, and data organization. It also considers ethical and legal aspects, including compliance with website terms of service and responsible data usage practices. The extracted dataset can support various applications such as business directory creation, sentiment analysis, geographic data visualization, and machine learning models for predictive analytics.

Keywords: Web Scraping Google Maps, Python, Selenium, BeautifulSoup, Data Extraction, Automation

I INTRODUCTION

In the digital era, location-based services have become an integral part of everyday life, enabling users to search for businesses, navigate routes, read reviews, and make informed decisions. One of the most widely used mapping platforms is **Google Maps**, which provides extensive geographic and

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

business-related information across the globe. The platform contains valuable data such as business names, addresses, phone numbers, ratings, reviews, operating hours, and

geographic coordinates. This vast repository of structured and semi-structured information makes it a powerful source for data-driven research, market analysis, and business intelligence.

Web scraping is a technique used to automatically extract data from websites for analysis and processing. In recent years, Python has emerged as a preferred programming language for web scraping due to its simplicity, flexibility, and availability of powerful libraries. Tools such as BeautifulSoup, Requests, and Selenium enable developers to collect and process web content efficiently, including dynamically loaded data. Since Google Maps primarily relies on JavaScript to render its content, automated browser interaction is often required to retrieve complete and accurate information.

Scraping data from Google Maps can support multiple applications such as competitor analysis, customer sentiment analysis, geographic trend mapping, and the development of business directories. By converting unstructured web data into structured datasets like CSV or Excel files, organizations can perform statistical analysis, visualization, and predictive modeling. However, it is also essential to consider ethical and legal aspects while collecting data, ensuring compliance with platform policies and responsible data usage practices.

This project aims to demonstrate how Python-based tools can be used to automate the extraction of relevant business information from Google Maps and transform it into meaningful insights for academic and commercial use.

II RELATED WORK

Several studies and practical implementations have explored web scraping techniques for extracting large-scale data from online platforms, particularly location-based services such as **Google Maps**. Researchers have highlighted the importance of automated data collection methods for business intelligence, market research, and geographic information system (GIS) applications. Early approaches to web scraping primarily relied on static HTML parsing using libraries such as BeautifulSoup and Requests. However, with the increasing adoption of JavaScript-based dynamic content rendering, more advanced tools like Selenium and headless browsers have become essential for extracting complete and accurate data from interactive platforms.

Previous works have demonstrated the use of scraping techniques to collect business listings, customer reviews, ratings, and geolocation data for sentiment analysis and trend prediction. Some studies focused on competitor analysis by extracting business profiles and comparing pricing strategies, service categories, and customer feedback. Others applied machine learning models to scraped review data to classify sentiments and predict customer satisfaction levels. In the context of smart cities and urban planning, researchers have utilized location-based datasets to analyze commercial density, service accessibility, and regional development patterns.

Additionally, various open-source frameworks and APIs have been developed to simplify data extraction from mapping services. While official APIs provide structured access to certain datasets, they often come with limitations such as request quotas, usage restrictions, and associated costs. As a result, researchers have explored browser automation and scraping techniques to gather publicly available information beyond API constraints. However, literature also emphasizes the importance of adhering to legal guidelines, ethical considerations, and terms of service when performing web scraping activities.

III LITERATURE REVIEW

Web scraping has been widely studied as an effective method for collecting structured and unstructured data from web platforms for research and commercial applications. With the rapid growth of location-based services, platforms such as **Google Maps** have become significant sources of business and geographic information. Several researchers have explored automated techniques to extract business listings, customer reviews, ratings, and contact details to support decision-making processes in areas such as marketing, urban planning, and data analytics.

Earlier literature primarily focused on static web scraping methods using Python libraries like BeautifulSoup and Requests to parse HTML content. These approaches were effective for websites with fixed page structures but faced limitations when dealing with dynamic content rendered through JavaScript. As modern web applications increasingly rely on asynchronous loading techniques, researchers introduced browser automation tools such as Selenium and headless browsers to simulate user interactions and retrieve fully rendered data. These advancements significantly improved the efficiency and accuracy of data extraction from interactive platforms.

Several studies have utilized scraped data for sentiment analysis, where customer reviews were processed using natural language processing (NLP) techniques to identify positive, negative, or neutral opinions. Other research focused on competitor analysis by collecting business information such as pricing, service categories, and customer ratings to evaluate market trends. In the field of geographic information systems (GIS), extracted location coordinates have been used for spatial analysis and visualization, enabling researchers to study commercial distribution and accessibility patterns.

The literature also discusses the limitations and challenges associated with web scraping, including anti-scraping mechanisms, data inconsistency, and legal considerations. Researchers emphasize the importance of ethical data usage and compliance with platform policies. Overall, existing studies confirm that Python-based scraping methods provide a powerful approach for gathering and analyzing location-based data when implemented responsibly and efficiently.

IV EXISTING SYSTEM

The existing system for collecting business and location-based data from platforms such as **Google Maps** primarily relies on manual data collection or the use of official APIs. In many cases, users manually search for businesses on Google Maps and record details such as name, address, phone number, ratings, and reviews. This manual approach is time-consuming, labor-intensive, and prone to human error, especially when dealing with large volumes of data. It is not suitable for large-scale analysis or real-time data processing.

Another common method in the existing system is the use of the Google Maps API, which provides structured access to specific datasets such as place details, geocoding, and directions. While APIs offer reliable and well-formatted data, they often come with certain limitations. These include request quotas, usage restrictions, pricing plans, and limitations on the amount of information that can be retrieved per request. For researchers and small-scale businesses, these constraints may restrict large-scale data collection and analysis.

In addition, some third-party tools and software solutions provide pre-extracted business directories. However, such solutions may not always offer updated information, customization options, or flexibility in selecting specific data fields. They may also involve subscription costs and limited data control.

DISADVANTAGES

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

The existing system of collecting data from **Google Maps** has several limitations that affect efficiency and scalability. Manual data collection is highly time-consuming and requires significant human effort, especially when gathering large volumes of business information. This process is also prone to human errors such as incorrect data entry, missing fields, or duplication, which can reduce the accuracy and reliability of the dataset. Additionally, relying on official APIs introduces constraints such as request limits, usage quotas, and subscription costs, which may not be feasible for extensive data extraction or academic research purposes. Third-party data providers may offer pre-collected datasets, but these are often outdated, incomplete, or lack customization options based on specific requirements. Furthermore, existing approaches may not effectively handle dynamically loaded content, leading to incomplete data retrieval. Scalability is another major concern, as traditional methods cannot efficiently manage large-scale data extraction tasks. These disadvantages highlight the need for a more automated, flexible, and scalable solution for extracting structured information from Google Maps.

V PROPOSED SYSTEM

The proposed system aims to develop an automated and efficient data extraction framework for collecting publicly available business information from **Google Maps** using Python. Unlike the existing manual and API-based approaches, this system leverages web scraping and browser automation techniques to gather large-scale data in a structured and organized manner. The system utilizes Python libraries such as Requests and BeautifulSoup for parsing web content, along with Selenium to handle dynamically loaded JavaScript content by simulating real user interactions. This enables the extraction of detailed information including business names, addresses, phone numbers, ratings, reviews, and geographic coordinates.

The proposed solution focuses on automation, scalability, and data accuracy. It includes modules for search query input, automated scrolling and data loading, content extraction, data cleaning, and structured storage in formats such as CSV or Excel. The system also incorporates basic error handling and rate control mechanisms to ensure smooth operation. By converting unstructured web data into a structured dataset, the proposed system enables advanced analysis such as sentiment analysis, market trend evaluation, and geographic visualization.

ADVANTAGES

The proposed system for scraping data from **Google Maps** offers several significant advantages over traditional manual and API-based methods. One of the primary benefits is automation, which reduces human effort and minimizes errors in data collection. The system can extract large volumes of business information quickly and efficiently, making it suitable for large-scale research and market analysis. Unlike API-based solutions that impose request limits and usage costs, the proposed scraping approach provides greater flexibility in selecting specific data fields such as business name, address, ratings, reviews, and geographic coordinates.

Another major advantage is scalability, as the system can handle multiple search queries and locations systematically. It also supports dynamic content extraction using browser automation tools, ensuring more complete and accurate data retrieval. The collected data can be stored in structured formats like CSV or Excel, enabling easy analysis, visualization, and integration with machine learning models. Additionally, the system is customizable, allowing users to modify search parameters based on their specific requirements. Overall, the proposed system enhances efficiency, flexibility, and data accessibility for business intelligence and analytical applications.

VI METHODOLOGY

The methodology for scraping data from **Google Maps** using Python follows a systematic and structured approach to ensure accurate and efficient data extraction. The process begins with defining the search query, such as a specific business category and location. Based on this input, the system automatically launches a web browser using Selenium to access Google Maps and simulate user actions such as searching, scrolling, and loading additional results. Since Google Maps content is dynamically generated using JavaScript, browser automation ensures that all relevant data is fully rendered before extraction.

Once the required results are loaded, the system identifies and captures essential data fields such as business name, address, contact number, ratings, number of reviews, and geographic coordinates. The extracted raw data is then processed using parsing techniques with libraries like BeautifulSoup to structure the information properly. Data cleaning steps are applied to remove duplicates, handle missing values, and ensure consistency in formatting.

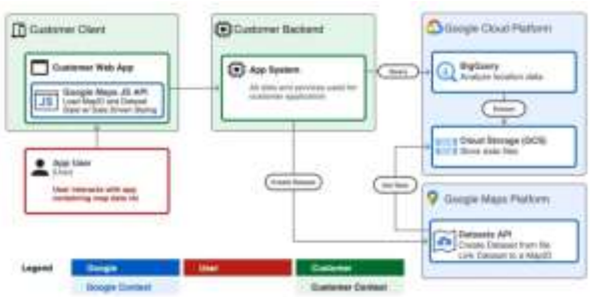
After processing, the cleaned data is stored in structured formats such as CSV or Excel files for further analysis. Basic error handling and delay mechanisms are incorporated to maintain smooth execution. This methodology

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

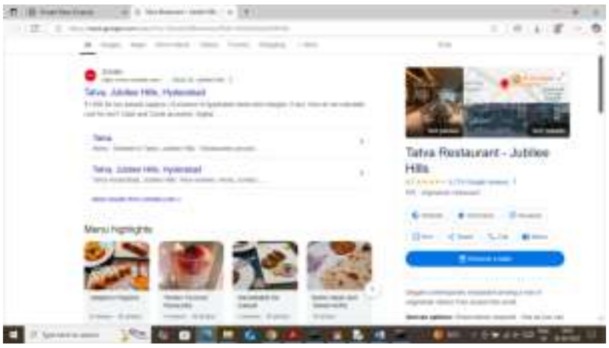
ensures automation, scalability, and organized data storage, enabling effective use of extracted information for analytical and research purposes.

VII SYSTEM MODEL

SYSTEM ARCHITECTURE



VIII RESULT AND DISCUSSIONS



Zestera PUBLICATIONS
International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

IX CONCLUSION

In conclusion, scraping data from **Google Maps** using Python provides an effective and automated approach for collecting large-scale location-based business information. Traditional manual data collection methods are time-consuming and prone to errors, while API-based solutions often impose limitations such as request quotas and usage costs. The proposed Python-based scraping system overcomes these challenges by utilizing automation tools and parsing techniques to extract structured data efficiently.

By integrating libraries such as Selenium for handling dynamic content and BeautifulSoup for parsing web elements, the system ensures accurate retrieval of important details including business names, addresses, ratings, reviews, and contact information. The extracted data can be cleaned, organized, and stored in formats like CSV or Excel, enabling further analysis such as market research, sentiment analysis, and geographic visualization.

REFERENCES

1. Google Maps. (2024). *Google Maps Platform Documentation*. Google Developers.
2. Selenium. (2024). *Selenium Documentation*. SeleniumHQ.
3. Beautiful Soup. (2024). *Beautiful Soup Documentation*. Crummy.
4. Requests. (2024). *Requests: HTTP for Humans Documentation*. Python Software Foundation.
5. Mitchell, Ryan. (2018). *Web Scraping with Python* (2nd ed.). O'Reilly Media.
6. Chacon, Scott&Straub, Ben. (2014). *Pro Git*. Apress.

International Journal of
AI Electronics and Nexus Energy
Peer Reviewed, Referred & Indexed Journal
ISSN: 3070-0515 www.zesterapublications.com Original Research Paper

7. Python Software Foundation. (2024). *Python Documentation*.
8. Google. (2024). *Google Maps Platform Terms of Service*.
9. Bird, S., Klein, E., & Loper, E. (2009). *Natural Language Processing with Python*. O'Reilly Media.
10. Tan, P. N., Steinbach, M., & Kumar, V. (2019). *Introduction to Data Mining* (2nd ed.). Pearson.
11. Sharma, P., & Gupta, R. (2023). AI and Geo-Fencing Based Smart Tourist Safety Framework. International Journal of Advanced Computer Science.
12. Sharma, S., & Kaur, R. (2019). Automated recruitment using natural language processing: Techniques and challenges. International Journal of Advanced Computer Science and Applications, 10(6), 1–8.
13. Dayal, P. S., Chandra, B. R., Keerthi, M., Sruthi, M., Venkatesh, K., Appalaraju, G., & Eswari, G. (2013). Design of Pyramidal Horn Antenna at 10GHz Using WIPL-D Optimizer. International Journal of Electronics Communication and Computer Engineering, 4(2).
14. Viswanathan, V., Polagani, S. S., Agarwal, R., Akula, S., Dey, S., & Kashyap, R. (2025, September). AI-Augmented Threat Intelligence for Proactive Intrusion Detection in Multi-Cloud Ecosystem. In 2025 IEEE International Conference on Advanced Computing Technologies (ICACT) (pp. 567-572). IEEE.
15. Sruthi, M. V., Sree, V. U., & Soundararajan, K. (2012). Specific removal of motion artifacts in medical image processing. IJECCE, 3(3), 227-229.
16. Viswanathan, V., Shah, A. K., Kubam, C. S., Dontu, S., Gandhi, A., & Singla, P. (2025, August). Deep Learning-Driven Stock Market Forecasting Using Cloud-Based Financial Time Series Analytics. In 2025 International Conference on Emerging Trends in Networks and Computer Communications (ETNCC) (pp. 1-6). IEEE.
17. Viswanathan, V. (2025). Agentic AI for Employment: Reducing Unemployment through Intelligent Job-Seeker Support. LEX LOCALIS–Journal of Local Self-Government.
18. Viswanathan, V. (2024). Pioneering Ethical AI Integration in Enterprise Workflows: A Framework for Scalable Team Governance. Available at SSRN 5375619.

19. Sruthi, M. V., Soundararajan, K., & Sree, V. U. (2012). Accurate Multimodality Registration of medical images. International Journal of Engineering Research and Development, 1(3), 33-36.
20. Ranjbareslamloo, S., Dzukeya, G. A., Muhit, M. M. I., & Qattawi, A. (2025). Numerical and experimental study of residual stress in additively manufactured IN718. Manufacturing Letters, 44, 915–927. https://doi.org/10.1016/j.mfglet.2025.915927
21. Mahtabi, M., Roshan, M., Muhit, M. M. I., Behvar, A., & Haghshenas, M. (2026). Cryogenic ultrasonic fatigue: Mechanisms, advancements, and insights. Cryogenics, 153, 104257. https://doi.org/10.1016/j.cryogenics.2025.104257
22. Kotte, G. (2025). Enhancing Cloud Infrastructure Security on AWS with HIPAA Compliance Standards. SSRN Electronic Journal. https://doi.org/10.2139/ssrn.5283660
23. GIRISH KOTTE. (2025). ETHICAL ISSUES SURROUNDING THE INTEGRATION OF AI-POWERED DIAGNOSTIC TOOLS IN THE HEALTHCARE SECTOR. American Journal of AI Cyber Computing Management, 5(4), 329–334. https://doi.org/10.64751/ajaccm.2025.v5.n4.pp329-334
24. Kumara, S. (2025). Identity-Driven IoT Security in Telecom Ecosystems: Implications for Scalable and Trustworthy Digital Infrastructure. Int. J. Appl. Math, 38(12s), 2797-2816.
25. Poojari, R. INTELLIGENT SYSTEMS+B108 AND APPLICATIONS IN ENGINEERING.
26. Cyril, H. P., & Kumara, S. (2026, February). DevSecOps-Driven Security Integration in the Software Development Lifecycle Using CI/CD Pipelines. In 2026 IEEE 5th International Conference on AI in Cybersecurity (ICAIC) (pp. 1-6). IEEE.
27. Prodduturi, S. M. K. To Secure Your Paper as Per UGC Guidelines We Are Providing A ElectronicBar code.
28. Santthosh Saai Reddy Purmani. (2026). Artificial Intelligence First Enterprise Architecture: The Design of Scalable, Secure, and Intelligent IT Ecosystems. American Journal of AI Cyber Computing Management, 6(1(2)), 1–8. https://doi.org/10.64751/ajaccm.2026.v6.n1(2).pp1-8

29. Purmani, S. S. R. (2025). Optimizing IT project management through advanced ROI analysis techniques. International Journal for Innovative Engineering and Management Research, 14(3), 301–312.
30. Patyrykin, K. (2025). CANCEL CULTURE PROBLEM. Lex Localis: Journal of Local Self-Government, 23.
31. Kalae, U. K. (2021). Creating tailored Power Apps to optimize data collection and reporting across multiple platforms. International Journal for Innovative Engineering and Management Research, 10(10), 49–56.
32. Patel, S., & Patyrykin, K. (2025). Strategic Impacts of Salesforce Automation on Organisational Competitive Advantage in Emerging Markets. Journal of Posthumanism, 5(12), 357–372. https://doi.org/10.63332/joph.v5i12.3782
33. Vasagam, M., Kumar, A., & Garg, A. (2026). Learning Execution Plan Embeddings for Multi-Dimensional Query Resource Prediction. IEEE Access.
34. Kalae, U. K. (2023). Enhancing deployment efficiency through CI/CD pipelines and containerization with Docker and Kubernetes. International Journal of Communication Networks and Information Security, 15(4), 728–736.
35. Poojari, R. Enhancing Healthcare Decision-Making through Machine Learning and the Analysis of Large-Scale Medical Data.
36. Akhilaiswarya, B., Sree, B. T., Lilly, K., Chowdary, K. H., & Sruthi, M. (2023). Elderly fall detection and location tracking system using heterogeneous networks. Journal of Engineering Sciences, 14(05).
37. Reddy, S. K. R. Developing a Modular AI Framework to Enhance Scalability and Personalization in Next-Generation Reward Platforms.