IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

DATA ENGINEERING PIPELINE

[1]Jyothiradithya M, [2]Prajith P G, [3]Sarathkrishnan P V, [4]Vidul K R, [5]Minu Augustine

[1]Student, [2]Student, [3]Student, [4]Student, [5]Assistant Professor (CSE)

[1]Computer Science and Engineering Department,

[1]Nehru College of Engineering and Research Centre (NCERC),Thrissur, India

Abstract: This project entails the development of a robust data pipeline for OLX data leveraging Python. The pipeline employs web scraping techniques or OLX APIs for data extraction, followed by meticulous data transformation in Python to ensure data integrity and relevance. The processed data is stored in a chosen repository, utilizing Python libraries and frameworks for seamless integration. Automation and scheduling are implemented using Pythonscripts.The pipeline incorporates error-handling mechanisms, and data quality checks are enforced throughout the process. The Python-based pipeline not only provides an efficient and scalable solution for handling OLX data but also facilitates easy integration with popular analytics tools, empowering users to derive actionable insights for strategic decision-making. The OLX Data Pipeline ensures a systematic and efficient processing of JSON-encoded data, laying the groundwork for insightful analytics and informed decision-making based on the structured information within OLX listings**.**

1 INTRODUCTION

The digital age has ushered in a transformative era in commerce, with online market- places playing a pivotal role in connecting buyers and sellers seamlessly. Within this landscape, OLX has emerged as a cornerstone platform, facilitating the exchange of goods and services on a global scale. As businesses increasingly recognize the indispensable role of data in shaping strategic decisions, this project embarks on the development of a sophisticated data pipeline tailored for OLX data. Leveraging the versatility and power of Python, the objective is to construct a comprehensive framework that efficiently extracts, transforms, and loads OLX data, laying the groundwork for insightful analytics and informed decision-making.

The first facet of this project revolves around the intricate process of data extraction. In this digital ecosystem, where information is dispersed across diverse platforms and interfaces, the project explores techniques such as web scraping and API integration to harvest relevant data from OLX. The second critical phase involves data transformation, where Python takes center stage. Through meticulous cleaning, normalization, and enrichment, the extracted data is refined into a structured format conducive to in-depth analysis. The final stage encompasses storage and automation, ensuring that the processed data is not only accessible but also dynamically updated through scheduled tasks, allowing for a seamless and timely flow of information.

Looking beyond the immediate implementation, the future scope of this project is marked by adaptability and innovation. The dynamic nature of online marketplaces suggests that continuous refinement of data extraction methods will be essential to keep pace with evolving OLX structures. Furthermore, the integration of advanced analytics and machine learning techniques holds the promise of uncovering deeper insights into user behavior, market trends, and predictive modeling. This project serves as a foundational step toward a dynamic and evolving solution, poised to navigate the shifting landscapes of online commerce and contribute to the ongoing narrative of data-driven decision-making in the digital era.

Data pipeline with the data of an e com platform like OLX have many uses like trend analysis Marketplace analysis involves the systematic examination of data within an online marketplace, like OLX, to glean valuable insights into user behavior, listing trends, competitive dynamics, and overall platform performance. By

studying key performance indicators, user feedback, and market trends, businesses can make informed decisions to optimize user experiences, enhance marketing strategies, and adapt to changing market conditions. This analysis encompasses understanding user engagement patterns, evaluating the popularity of product listings, assessing the competitive landscape, and monitoring critical metrics to gauge the success of promotional campaigns. Furthermore, marketplace analysis aids in fraud detection, ensuring the integrity and security of the platform. Through a comprehensive understanding of marketplace dynamics, businesses can strategically position themselves, optimize operations, and meet the evolving demands of the online marketplace ecosystem.

2 RELATED WORK

Web scraping is a process of extracting valuable and interesting text information from web pages. Most of the current studies targeting this task are mostly about automated web data extraction , According to [1] Web pages are made of HTML elements and data between these elements. Web scraping is a process of extracting specific data between these elements for providing data for other applications such as online price change monitoring, product review analyzing, weather data monitoring, tracking online presence, gathering articles and so on. The internet is a rich source of ‘‘big data’’ for these applications. Most of the studies in ‘‘big data’’ concentrate on the time efficiency of deep learning models. However, to increase the time efficiency of obtaining data is an important issue. The Uzun Ext approach consists of two main components: crawling web pages and extracting data from them. It uses the additional information obtained from web pages during the crawling process for increasing the extraction time efficiency.

Information extraction (IE) is a challenging task, particularly when dealing with highly heterogeneous data. State-of-the-art data mining technologies struggle to process information from textual data. Therefore, various IE techniques have been developed to enable the use of IE for textual data. However [2] states that, each technique differs from one another because it is designed for different data types and has different target information to be extracted. This study investigated and described the most contemporary methods for extracting information from textual data, emphasizing their benefits and shortcomings.

In case of [3] we can see that In service-oriented architectures, data exchange relies on some choice of resource representation, with XML or JSON being the most popular ones. XML and JSON are structured documents where structure tags specify the type of the data elements they annotate, enabling the independent services to share commonly agreed upon semantics. Element and attribute tags in XML and, correspondingly, keys in JSON serve as annotations to the actual data values being exchanged. These annotations tags are invariably repeated several times in the request and, especially response, documents. While this repetition arguably makes XML and JSON self-descriptive, it also contributes to verbosity of these documents. The main purpose of compressing structured documents such as XML and JSON is to reduce their size without information loss.XML and corresponding JSON representation of the same structured data returned by a web service are comparable, with JSON being slightly smaller because it does not require ‘‘closing tags’’ for the various data elements.in our literature review on compression of structured documents, most work has been on the compression of XML and relatively limited attention has been given to JSON. The method we describe in this paper is applicable to structured documents in general, including XML and JSON, relying on the notion that the documents in question exhibit a regular structure composed of free text encapsulated in regular pat- terns of labels, i.e., words from a limited vocabulary.

According to [4] Large quantities of semistructured documents are appearing on the Web. In recent years, Web data-extraction techniques have been applied in many automatic agent systems, such as price comparison and recommendation systems. They access Web documents to extract and integrate data and to provide data services to users. Unlike free-text documents, semistructured documents have embedded structures. However, there is no explicit schema that comes with these documents, making them difficult to be processed automatically.
The next process we can see is data processing, in [5] the data processing work is detailed. Data preprocessing includes a variety of tasks such as cleaning, encoding, scaling and dimensionality reduction [11]. The task of resolving data issues is referred to as data cleaning. Typical data issues include missing values, inaccurate datatypes, and repeated rows. Data cleaning is intended to clean the data with missing values, inconsistencies,

and noisy data . Data preprocessing is also intended to perform feature encoding, scaling, and Dimensionality reduction.

3 METHODOLOGY

The existing system reveals a notable gap in the organization's capability to effectively harness OLX data for strategic decision-making. Currently, the absence of a dedicated and efficient process for Extracting, Transforming, and Loading (ETL) OLX data impedes the organization's ability to derive meaningful insights in a timely and efficient manner. Manual data retrieval methods or sporadic scripts are utilized, leading to labor-intensive processes prone to errors. Moreover, the lack of a structured pipeline results in inconsistent data quality and hampers accurate analysis. Without automation, maintaining up-to-date information becomes a challenge, exacerbating the inefficiencies within the system. Error-handling mechanisms are absent, further exacerbating the risk of data discrepancies.

In response to these challenges, the proposed OLX Data Pipeline seeks to revolutionize the organization's approach to handling OLX data. By leveraging Python-based web scraping techniques or OLX APIs, the pipeline will enable efficient data extraction, ensuring a steady influx of up-to-date information. Meticulous data transformation processes will be implemented to guarantee data integrity and relevance, addressing the current inconsistencies in data quality. Automation and scheduling functionalities will streamline the data processing workflow, reducing manual intervention and enhancing operational efficiency.

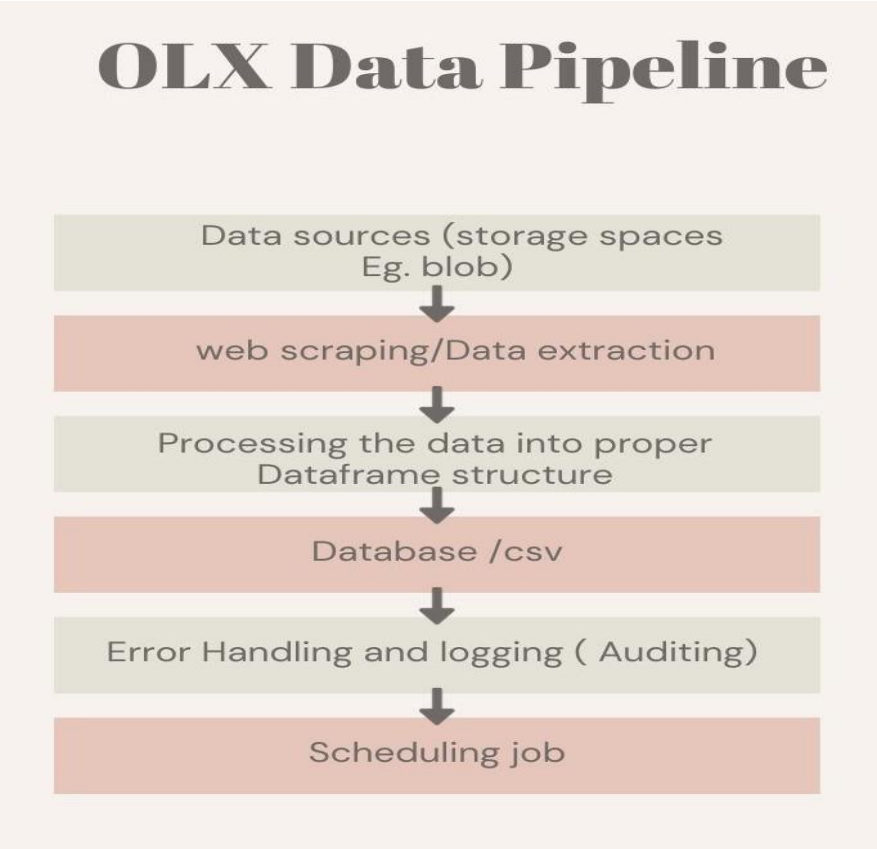
Central to the proposed system is the integration of error-handling mechanisms and data quality checks, mitigating the risk of data discrepancies and ensuring the reliability of the processed data. The Python-based pipeline will not only provide a scalable solution for handling OLX data but also facilitate seamless integration with popular analytics tools. This integration empowers users to derive actionable insights for strategic decision-making, thus unlocking the full potential of OLX data.

In essence, the proposed OLX Data Pipeline represents a paradigm shift from the inefficiencies of the existing system to a streamlined, reliable, and scalable solution for harnessing OLX data. By addressing the shortcomings of the current setup and leveraging advanced technologies, the proposed system sets the stage for informed decision-making and strategic growth.

Objectives:

- Develop a reliable data pipeline for extracting, transforming, and storing OLX data.
- Ensure data integrity and relevance through meticulous data transformation.
- Implement automation and scheduling for seamless data processing.
- Enforce error-handling mechanisms and data quality checks throughout the pipeline.

System Architecture:



The system will consist of:

- Data extraction module using web scraping techniques or OLX APIs.
- Data transformation module in Python to ensure integrity and relevance.
- Data storage module utilizing Python libraries and frameworks for seamless integration.
- Automation and scheduling module implemented using Python scripts.
- Error-handling mechanisms and data quality checks integrated throughout the pipeline.

Data Extraction Module

Responsible for extracting data from OLX, utilizing web scraping tools (e.g., Beautiful Soup, Selenium) or OLX APIs.Ensures reliable and periodic retrieval of the latest OLX data.

Data Transformation Module

Cleans, normalizes, and enriches the extracted data using Python libraries such as Pandas, NumPy, Prepares the data for structured storage and subsequent analysis.

Data Storage Module

Selects and implements a suitable database management system (e.g., PostgreSQL, MongoDB) to store processed OLX data, Ensures scalability and accessibility of data storage, considering the increasing volume of OLX data

Automation and Scheduling Module

Utilizes workflow orchestration tools like Apache Airflow or simple automation tools (e.g., cron jobs) to schedule and automate pipeline tasks enables regular and automated execution of the entire data pipeline.

Error Handling and Logging Module

Implements mechanisms for logging using Python's logging module to track the pipeline's execution. includes error-handling components to grace- fully manage unexpected issues during data extraction or transformation.

Web Scraping / API Calls Module

Conducts web scraping activities if chosen as the data extraction method, utilizing tools like Beautiful Soup or Selenium. Handles API calls to OLX if APIs are available for a more structured data retrieval process.

Notification module

Using the Bot token and Chat id of a Telegram bot , the system will sent a message which tells that the data retrieved successfully

4 PROBLEM STATEMENT

In the current landscape of leveraging OLX data for strategic decision-making, there exists a notable deficiency in the absence of a systematic and efficient data pipeline. The manual or sporadic methods employed for data extraction, coupled with the lack of a standardized transformation process, hinder the organization's ability to harness the full potential of OLX data. The absence of an automated and well-orchestrated data pipeline results in inefficiencies, including time-consuming data retrieval processes, inconsistent data quality, and the potential for errors. These limitations impede the organization's capacity to derive meaningful insights, track marketplace trends, and make informed decisions in a timely manner. Recognizing the critical role that data plays in today's digital marketplace, the problem at hand is to design and implement a robust Python-based data pipeline for OLX data, addressing these inefficiencies and providing a scalable solution that optimizes the extraction, transformation, and loading (ETL) processes, ultimately empowering the organization to unlock the full analytical potential of OLX data.

5 SYSTEM REQUIREMENTS

HARDWARE REQUIREMENT

Ram: 4 GB or above
Processor: Intel core I3 or above
Hard Disk Space: 320 GB
Sufficient storage capacity to store both raw and processed OLX data

SOFTWARE REQUIREMENTS

Python, selenium: These are used for web based data extraction.

Python: Used for processing the data and enhancing the data to a data frame structure.

Kubernetes / airflow :In future Automate the whole process and schedule a job (daily, weekly, monthly)

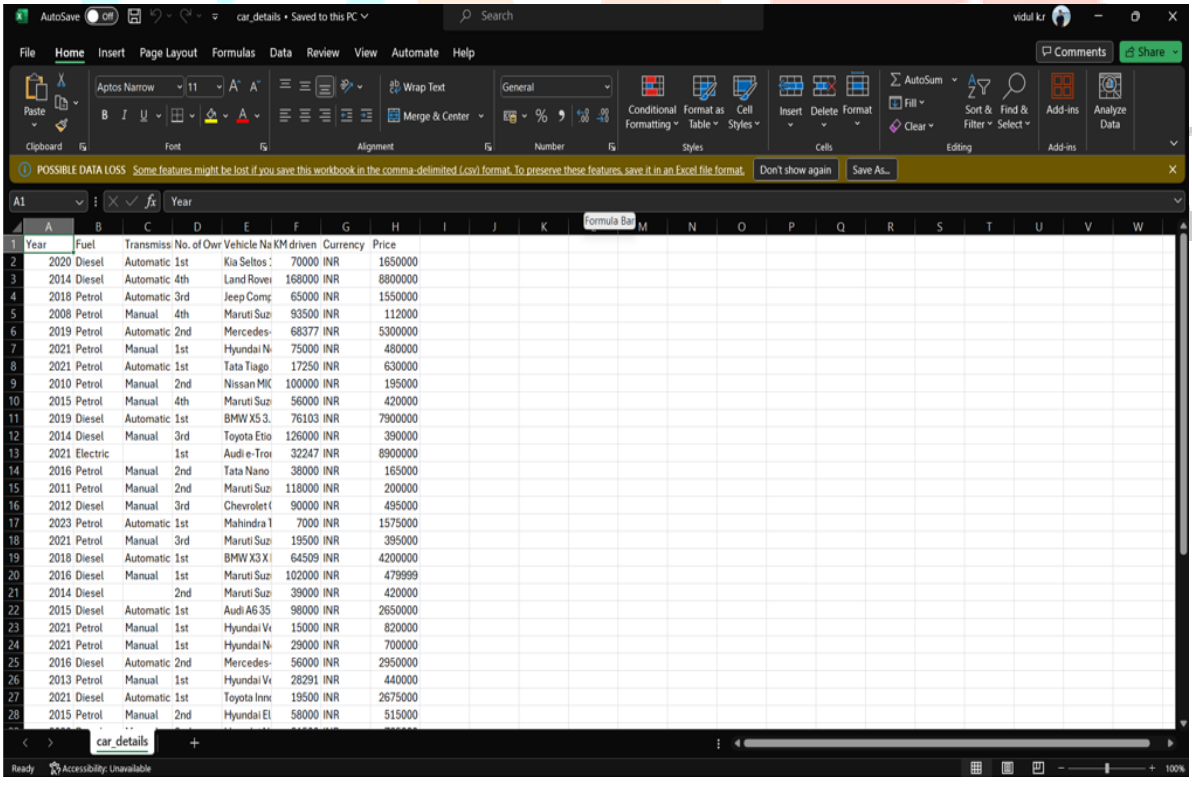
Database: Used to store the data in to table format.

Auditing: Based on the job run understand the errors, failure, etc.....

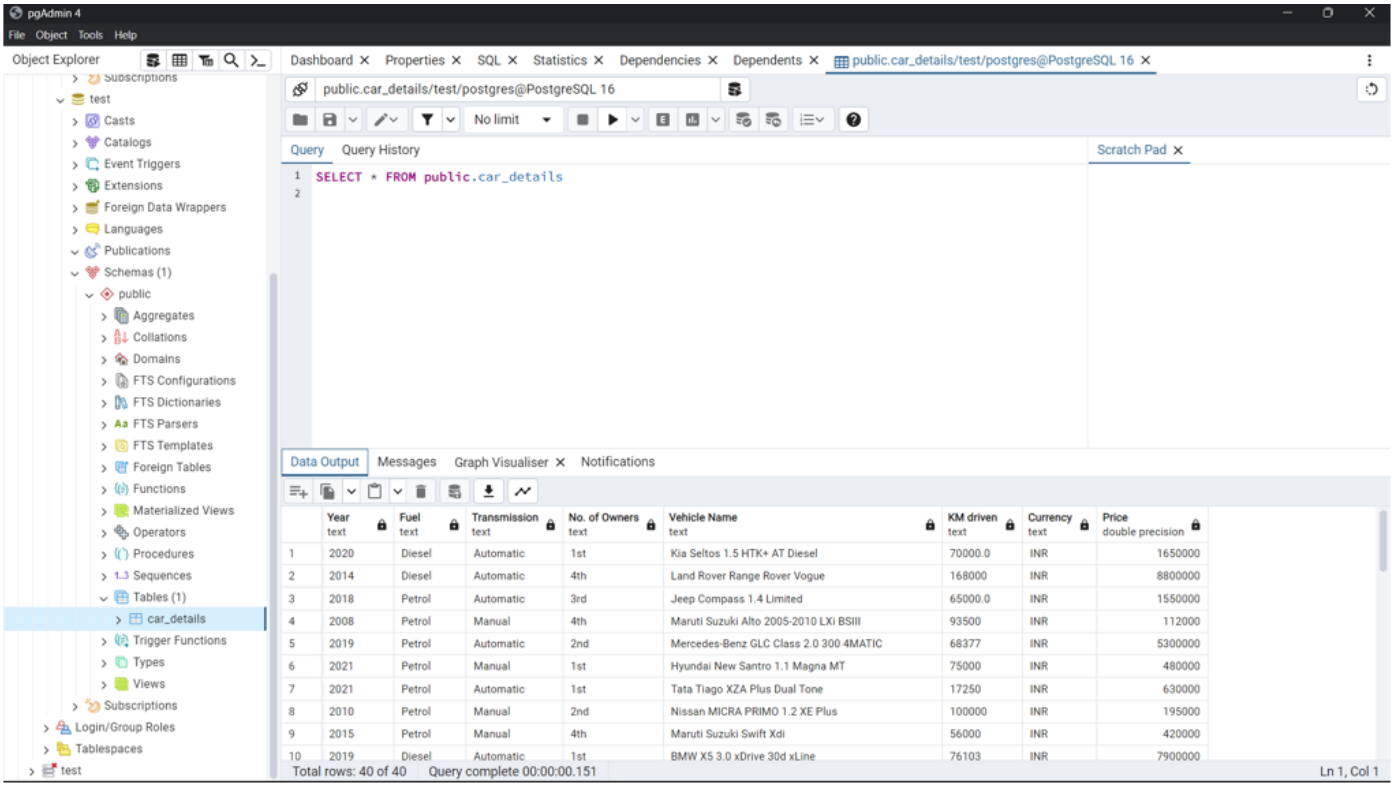
Sending notification: to telegram/teams/ other platforms about the status of job run

6. RESULTS AND DISCUSSION

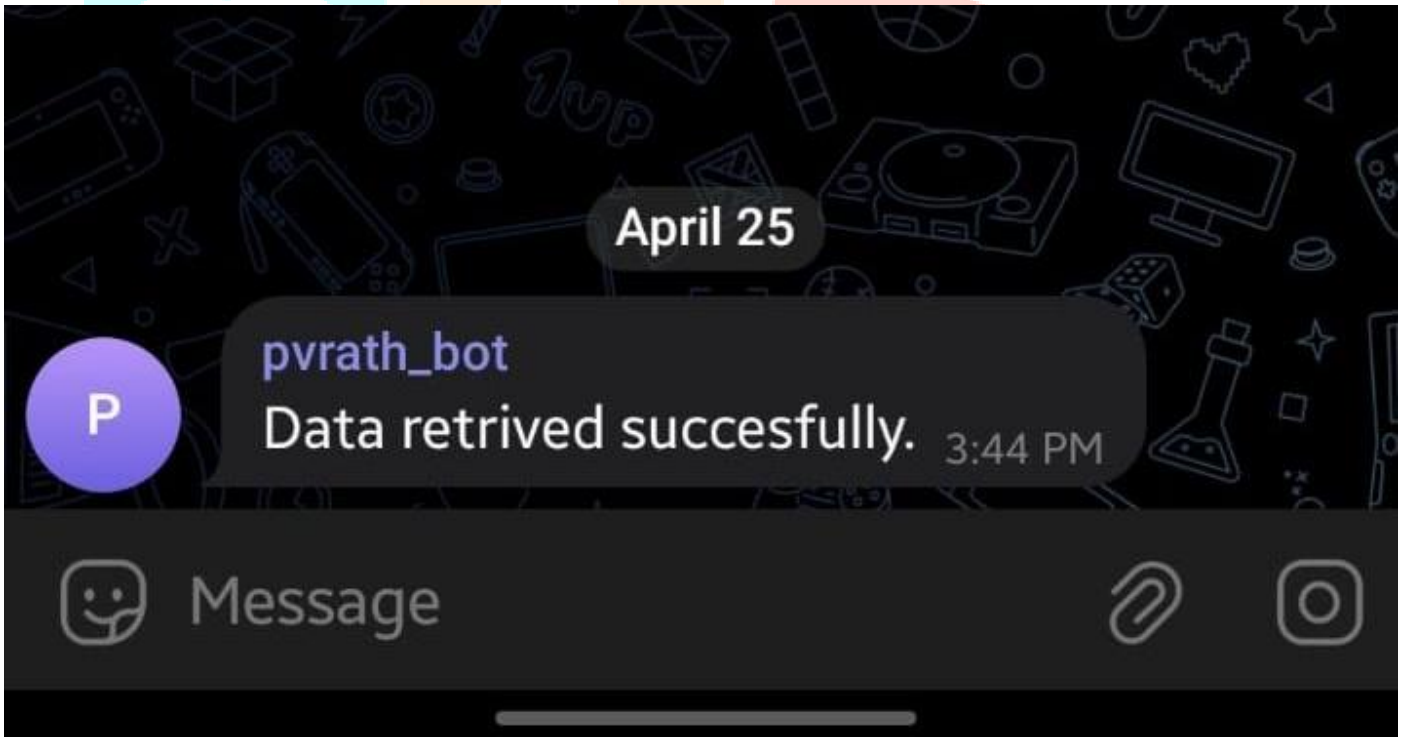

```
{
  "version": "108.0",
  "data": [
    {
      "id": "1760497281",
      "score": 1.0,
      "spell": {
        "id": 112,
        "key": "DISCERN_LOCATION_ES",
        "version": "1",
        "main": false,
        "facet_disabled": false,
        "default_sorting": "DEFAULT"
      },
      "status": {
        "status": "active",
        "allow_edit": true,
        "display": "active",
        "translated_display": "Active",
        "flags": {
          "hot": false,
          "new": false
        },
        "allow_deactivate": true
      },
      "category_id": "84",
      "ad_id": "1760497281",
      "favorites": {
        "count": 1,
        "count_label": "1",
        "count_label_next": "2"
      },
      "has_phone_param": false,
      "description": "ADDITIONAL VEHICLE INFORMATION:\n\nInsurance Type: Comprehensive\nMake Month: March\nRegistration Place: KL",
      "created_at": "2024-02-06T10:26:31+05:30",
      "inspection_info": null,
      "title": "Kia Seltos HTK Plus AT D, 2020, Diesel",
      "multi_location": null,
      "platform": null,
      "car_body_type": "SUV",
      "partner_id": null,
      "user_type": "Regular",
```

Year	Fuel	Transmiss	No. of Owr	Vehicle Na	KM driven	Currency	Price
2020	Diesel	Automatic	1st	Kia Seltos	70000	INR	1650000
2014	Diesel	Automatic	4th	Land Rover	168000	INR	8800000
2018	Petrol	Automatic	3rd	Jeep Comp	65000	INR	1550000
2008	Petrol	Manual	4th	Maruti Suzi	93500	INR	112000
2019	Petrol	Automatic	2nd	Mercedes-	68377	INR	5300000
2021	Petrol	Manual	1st	Hyundai N	75000	INR	480000
2021	Petrol	Automatic	1st	Tata Tiago	17250	INR	630000
2010	Petrol	Manual	2nd	Nissan MIC	100000	INR	195000
2015	Petrol	Manual	4th	Maruti Suzi	56000	INR	420000
2019	Diesel	Automatic	1st	BMW X5 3.	76103	INR	7900000
2014	Diesel	Manual	3rd	Toyota Etio	126000	INR	390000
2021	Electric		1st	Audi e-Tror	32247	INR	8900000
2016	Petrol	Manual	2nd	Tata Nano	38000	INR	165000
2011	Petrol	Manual	2nd	Maruti Suzi	118000	INR	200000
2012	Diesel	Manual	3rd	Chevrolet (	90000	INR	495000
2023	Petrol	Automatic	1st	Mahindra T	7000	INR	1575000
2021	Petrol	Manual	3rd	Maruti Suzi	19500	INR	395000
2018	Diesel	Automatic	1st	BMW X3 X	64509	INR	4200000
2016	Diesel	Manual	1st	Maruti Suzi	102000	INR	479999
2014	Diesel		2nd	Maruti Suzi	39000	INR	420000
2015	Diesel	Automatic	1st	Audi A6 35	98000	INR	2650000
2021	Petrol	Manual	1st	Hyundai Ve	15000	INR	820000
2021	Petrol	Manual	1st	Hyundai N	29000	INR	700000
2016	Diesel	Automatic	2nd	Mercedes-	56000	INR	2950000
2013	Petrol	Manual	1st	Hyundai Ve	28291	INR	440000
2021	Diesel	Automatic	1st	Toyota Inn	19500	INR	2675000
2015	Petrol	Manual	2nd	Hyundai El	58000	INR	515000



	Year text	Fuel text	Transmission text	No. of Owners text	Vehicle Name text	KM driven text	Currency text	Price double precision
1	2020	Diesel	Automatic	1st	Kia Seltos 1.5 HTK+ AT Diesel	70000.0	INR	1650000
2	2014	Diesel	Automatic	4th	Land Rover Range Rover Vogue	168000	INR	8800000
3	2018	Petrol	Automatic	3rd	Jeep Compass 1.4 Limited	65000.0	INR	1550000
4	2008	Petrol	Manual	4th	Maruti Suzuki Alto 2005-2010 LXi BSIII	93500	INR	112000
5	2019	Petrol	Automatic	2nd	Mercedes-Benz GLC Class 2.0 300 4MATIC	68377	INR	5300000
6	2021	Petrol	Manual	1st	Hyundai New Santro 1.1 Magna MT	75000	INR	480000
7	2021	Petrol	Automatic	1st	Tata Tiago XZA Plus Dual Tone	17250	INR	630000
8	2010	Petrol	Manual	2nd	Nissan MICRA PRIMO 1.2 XE Plus	100000	INR	195000
9	2015	Petrol	Manual	4th	Maruti Suzuki Swift Xdi	56000	INR	420000
10	2019	Diesel	Automatic	1st	BMW X5 3.0 xDrive 30d xLine	76103	INR	7900000



7. CONCLUSION

In concluding the OLX Data Pipeline project, we reflect on the journey from the initial recognition of inefficiencies in OLX data handling to the development of a robust, Python- based solution. The project successfully addresses the challenges associated with manual data extraction and lack of standardization by introducing a systematic approach to data pipeline management. By leveraging web scraping, APIs, and Python's versatile capabilities, we have created a dynamic framework capable of extracting, transforming, and loading OLX data efficiently. The modular architecture ensures scalability, flexibility, and adaptability to evolving marketplace dynamics.

The future scope of the project is promising, with opportunities for integration with advanced technologies, mobile application development, and enhanced user interfaces. The project's reusability is underlined by its

potential to serve as a template for other online marketplaces and the possibility of open-sourcing components for broader community contribution. The inclusion of emerging technologies, micro services architecture, and cloud-native services further positions the project to meet the ever-evolving demands of the digital marketplace ecosystem.

As we conclude, we envision the OLX Data Pipeline as more than a solution to immediate challenges; it stands as a foundation for continuous improvement and innovation. By empowering organizations to derive meaningful insights from OLX data, the project contributes to informed decision-making, strategic planning, and the overall enhancement of user experiences within the dynamic landscape of online marketplaces. The journey continues as we remain committed to refining, expanding, and adapting the project to embrace the limitless possibilities that the future holds in the realm of data-driven excellence.

REFERENCES

[1] ERDINC¸ UZUN "A Novel Web Scraping Approach Using the Additional Information Obtained From Web Pages"-date of publication March 31, 2020

[2] MOHD HAFIZUL AFIFI ABDULLAH , (Graduate Student Member, IEEE),NORSHAKIRAH AZIZ , SAID JADID ABDULKADIR , (Senior Member, IEEE),HITHAM SEDDIG ALHASSAN ALHUSSIAN, AND NOUREEN TALPUR "Systematic Literature Review of Information Extraction From Textual Data: Recent Methods, Applications, Trends, and Challenges" January 2023

[3] Gyan P Tiwari, Eleni Stroulia ,Abhishek srivastava. "Compression of XML and JSON API Responses"-2021

[4] Zhao Li, Wee Keong Ng, Aixin Sun-"Web data extraction based on structural similarity "DOI 10.1007/s10115-004-0188-z Springer-Verlag London Ltd. 2005 Knowledge and Information Systems (2005)

[5] Hussain A Jaber,Hadeel Kassim Aljobouri , l.cankaya -"Preparing fMRI Data for Postprocessing: Conversion Modalities, Preprocessing Pipeline, and Parametric and Nonparametric Approaches"- 2019

[6] Dunlu peng, Wenjie X u, -Using JSON for Data Exchanging in Web Service Application:Article• December 2011

[7] B. Bilalli, A. Abello´, T. Aluja-Banet, and R. Wrembel, "Automated data pre-processing via meta-learning," in Automated Data Pre-Processing Via Meta-Learning. New York, NY, USA: Springer, 2016, pp. 194–208.

[8] A. Alasadi and W. S. Bhaya, "Review of data preprocessing techniques in data mining," J. Eng. Appl. Sci., vol. 12, no. 16, pp. 4102–4107, 2017.

[9] S. Sakr, "XML compression techniques: A survey and comparison," J. Comput. Syst. Sci., vol. 75, no. 5, pp. 303–322, Aug. 2009.

[10] H.LiefkeandD.Suciu,"XMill:AnefficientcompressorforXMLdata," in Proc. ACM SIGMOD Int. Conf. Manage. Data (SIGMOD), 2000, pp. 153–164.