Used Car Data Extraction

BAX-422: Data Design and Representation

Avantika Goyal
Charles Wang
Shalagha Mundepi

Table of Contents

Executive Summary

In the dynamic landscape of the automotive industry, the used car market represents a significant segment, offering vast opportunities for buyers and sellers alike. Platforms like autolist.com have become central to connecting buyers to affordable, pre-owned vehicles while also providing a marketplace for car-owners looking to make a sale. If we had seamless access to all the vehicle data, we'd be able to draw key insights about the types of vehicles being sold, in which city, at what price, and many other factors. This data can help buyers make informed decisions and dealerships better understand the competition in the market.

This project report outlines the development and execution of a web-scraping and databasing approach to collect detailed information on used cars from autolist.com. We've leveraged tools such as Python, Selenium's ChromeDriver, and MongoDB to develop a smooth data extraction and storage process that can be expanded and modified for any context. The dataset we've built not only provides a thorough view of the current used cars in the San Francisco Bay Area but also serves as a foundation for answering meaningful business questions in the automotive industry.

Background and Domain Context

Market Growth and Demand: The demand for used cars has surged in recent years throughout the country. Contributing factors include economic uncertainties making new cars more expensive and difficult to acquire, high depreciation rates of new vehicles, and the increased reliability of cars prolonging their lifetimes. Moreover, environmental consciousness in the San Francisco Bay Area region has made buyers less inclined to purchase new vehicles, especially gas-guzzlers.

Technology: Digital marketplaces have permeated every industry and the used car market is no exception. Websites like autolist.com offer extensive listings and granular search tools, making it easier for buyers to browse and find their desired car. These platforms also store valuable data on pricing trends, vehicle specifications, and market demand, which enables informed decision making and targeted selling methods.

Pricing Trends: Prices in the automotive industry have been quite volatile due to supply chain disruptions, new car inventory shortages, and changing consumer preferences. The used car market would reflect similar trends since car-owners can sell their car at a premium if new vehicles are much more expensive.

Consumer Preferences: The San Francisco Bay Area demonstrates a pronounced preference for compact and environmentally friendly vehicles, particularly electric vehicles (EVs). By analyzing data extracted from autolist.com, we can determine whether these general preferences are indeed reflected within the used car market.

Data Sources and Web Scraping Techniques

Used Car Website

Since we were interested in analyzing trends in the used car market, we came up with a list of a few websites that fit this objective. Unfortunately, many of these websites, such as *carmax.com* and *cargurus.com*, have restrictions in place that prevent web scraping and automated browser tools from collecting their data. After visiting several websites, we decided on *autolist.com*, a top-rated car buying app and site that aggregates listings from a plethora of other websites across the country.

Search Parameters

As graduate students attending school in the heart of San Francisco, we wanted to set search parameters that resonated with us. We wanted to find which cars were on the market in the Bay Area and within an affordable price range.

Zip Code: 94102

Radius: Within 25 miles

Price: $0-$20,000

These parameters can be easily modified for other use cases (e.g. different city, demographic) if needed but our approach would remain the same.

Methodology

1. Save Search Results as HTML Files

The first stage in the project involved navigating the Chrome browser to visit autolist.com, setting the search filters, and saving the first 50 pages of the search results on our local drive as

HTML files. We used the Python library Selenium and its webdriver package to navigate through the website. We set *time.sleep()* commands between each line of code with random durations between 5-10 seconds to avoid bot detection. We used various selectors to click on the filters and buttons on the website. The figure below shows a snippet of the code we used to configure our filters.

```
# Visit autolist.com
driver.get('https://www.autolist.com/')

# Click on the Price search button
time.sleep(5)
driver.find_element(By.XPATH, '//button[text()="Price"]').click()

# Select the drop down menu and input $$20000
time.sleep(7)
maxprice = driver.find_element(By.CSS_SELECTOR, "input[placeholder='$100000']")
maxprice.clear()
maxprice.send_keys('20000')

# Click the Search button
time.sleep(10)
driver.find_element(By.XPATH, "//button[contains(text(), 'Search')]").click()
```

In order to save our search results as HTML, we used *driver.page_source* to get the HTML script and wrote to a local file named page{page_number).html.

2. Extract Data For Each Car

The second stage of our project involved parsing these HTML files to extract all the available data for each car. We wrote a nested loop that first iterated through each HTML file and then within each file iterated through each listing. We extracted the following data: Listing Title, Year, Make, Model, Mileage, Location, Days on Market, Price, Est. Monthly Pay, Listing URL, Trim, Transmission, Engine, Drive Type, Exterior Color, Interior Color, and Car Style.

One challenge that we faced was that some of the data was only available after clicking into the listing URL and not on the thumbnail. Initially, we attempted to implement a GET request to visit the URL and search the HTML using BeautifulSoup but that was not feasible. When we parsed the BeautifulSoup object after making the GET request it did not match the HTML we saw on the browser. To circumvent this, we utilized Selenium to visit each listing's URL, extracted the page HTML using *driver.page_source*, and finally parsed that HTML using BeautifulSoup. We defined a new function that implemented this process and nested it in our loop. The loop stored information for each car in a dictionary and each dictionary was appended to a list.

3. Store All Data in MongoDB Database

The last step in our project was to save all the data to a database. We chose MongoDB for its seamless integration with the dictionary data structure and because it allows for documents of different lengths and attributes. For our project we only parsed *autolist.com* and used one set of filter parameters, but if we want to scale this project in the future to collect data from different sources or criteria, MongoDB would be able to support that. Moreover, a relational database like SQL requires keys that relate tables to one another but our data doesn't have a natural 'key' or identifier, further validating our decision to use MongoDB. Lastly, if dealerships want to use this dataset for customer segmentation, MongoDB's aggregation and query capabilities will allow them to easily segment data on various criterias like location, car features, and price range. The figure below shows how one document appears in Studio 3T.


```
{
    "_id" : ObjectId("65fb234ae2bca42e10d2092f"),
    "title" : "2015 Ford Transit Connect XLT",
    "year" : "2015",
    "make" : "Ford",
    "model" : "Transit Connect XLT",
    "miles" : NumberInt(42692),
    "location" : "Colma, CA",
    "days_on_market" : NumberInt(21),
    "price" : NumberInt(18491),
    "est_monthly_pay" : NumberInt(267),
    "URL" : "https://www.autolist.com/listings#city=San%20Francisco&latitude=37.7786871&limit
    "trim" : "XLT",
    "transmission" : "automatic",
    "engine" : "Duratec 2.5L I4",
    "drivetrain" : "FWD",
    "ext_color" : "gasoline",
    "int_color" : "Tectonic Silver Metallic",
    "style" : "42692"
}
```

Application of Data in Business Domain

Customized Customer Experience: The dealership/website can use the extensive dataset to offer tailored experiences to customers, enhancing satisfaction and engagement. For instance, by examining specific details such as Car Style, Transmission, and Color, a dealership could quickly recommend the perfect match for a customer seeking a sporty sedan with a manual transmission in a black exterior color.

Pricing and Valuation Insights: The dealership can utilize the dataset's granular details on Year, Make, Model, and Mileage, alongside Price and Days on Market, to gain a nuanced understanding of how used cars are valued. For example, by observing that late-model sedans with under 50,000 miles tend to sell within 7 days on the market at a particular price point, dealers can adjust their pricing strategies accordingly.

Market Research and Analysis: Industry researchers can develop comprehensive market research reports tailored for client dealerships in the automotive sector, pinpointing opportunities and challenges. For instance, if analysis reveals a lack of affordable electric vehicles (EVs) in urban cities, automotive dealers can seize the opportunity by pricing EVs more competitively.

Informed Decision-Making: Buyers can use this data to enhance purchasing decisions by pinpointing vehicles that offer the best value within specific budget constraints. For example, a buyer looking for a durable and efficient used car under $15,000 can use insights from the analysis to choose between models like the Toyota Corolla or Honda Civic, known for their reliability and low maintenance costs.

Summary

In the rapidly evolving automotive industry, particularly within the used car market, businesses face the challenge of staying competitive and meeting consumer demands efficiently. This report outlines our approach to leveraging web-scraping technologies and database management to gather and analyze data from autolist.com, a prominent aggregator of vehicle listings. This project is aimed at providing dealerships and buyers with data that can inform pricing strategies, marketing segmentation, and buying decisions.

Our data collection process involved the use of web scraping techniques to extract comprehensive information from *autolist.com*, covering various attributes like Make, Model, Year, Price, Color, and more. Utilizing Python libraries such as Selenium and BeautifulSoup, we automated the collection of data across multiple search result pages, ensuring a rich dataset for analysis. MongoDB was our database of choice due to its flexibility, scalability, and integration with the dictionary data structure, enabling an expansion of this project in the future.

Employing this data collection and storage process can provide businesses and potential buyers in the used car market with a powerful toolkit for navigating market challenges. By enabling a data-driven approach to inventory management, customer engagement, and strategic planning, businesses can achieve a significant competitive edge, responding agilely to market dynamics and consumer trends.