Automated Extraction and Structuring of Real Estate Listings from Web Sources Using Python's Scrapy and Pandas Libraries

Sayat Sanzharyk
Department of Computer Science
Suleyman Demirel University (SDU)
Almaty, Kazakhstan
200107052@stu.sdu.edu.kz

I. Methodology

Our main goal was to create an automated system that could extract and organize real estate listings from *krisha.kz*. To make this happen, we used Python's Scrapy framework for web crawling and data extraction, and the Pandas library for organizing and handling the data.

A. Data Collection Setup

We kicked off by setting up a Scrapy project specifically designed to crawl *krisha.kz*. We chose Scrapy because it's efficient at handling large-scale web scraping tasks and can manage requests asynchronously. The spider was set up to start from the main listings page and navigate to individual property pages.

B. Spider Development

We programmed the spider to parse HTML content and pull out the relevant data fields, such as:

- **Location Details**: City, region, street, and district.
- **Price**: Listed price of the property.
- **Property Details**: Year of construction, property status (e.g., new, resale), and any extra descriptions provided by the seller.
- **View Counts**: Number of times the advertisement was viewed.

In this case i can give definition for the word "Spiders" : It is a programming objects that defines how a concrete site (or platform) will be scraped. We used XPath and CSS selectors to accurately find and extract the information we needed from the web pages' HTML structure.

C. Handling Anti-Scraping Measures

Krisha.kz uses anti-scraping techniques like CAPTCHA challenges and request rate limiting to prevent automated data extraction. To get around CAPTCHA validations, we integrated an automated CAPTCHA-solving service using an API that handles image-based CAPTCHAs and returns the solved text. Plus, to mimic human browsing behavior and avoid detection, we randomized the time intervals between requests and rotated user-agent strings.

D. Data Storage and Organization

During the scraping process, the extracted data was initially stored in JSON format. We then used the Pandas library to read the JSON data into a DataFrame for cleaning and preprocessing. Our data cleaning steps included:

- **Handling Missing Values**: We either filled in missing critical information with placeholder values or removed those entries, depending on how much data was missing.
- **Data Type Conversion**: We made sure numerical fields like price and year of construction were in the right format for analysis.
- **Duplicate Removal**: We identified and removed duplicate listings to keep the data accurate.

E. Experimental Procedure

We ran the scraper over a week to collect a comprehensive dataset that covered various regions and property types. To minimize the load on the website's servers and reduce the risk of our IP being blocked, we scheduled the scraping tasks during off-peak hours.

F. Tools and Technologies

- **Python 3.8**: The programming language we used for developing the scraper and data processing scripts.
- **Scrapy 2.5**: The web scraping framework we used for crawling and data extraction.
- **Pandas 1.2**: Used for data manipulation and organization.
- **CAPTCHA Solver API**: An external service we integrated to automatically solve CAPTCHA challenges during scraping.

G. Limitations

While our approach effectively extracted over 500 listings, we did face some limitations. Relying on third-party CAPTCHA-solving services might not be sustainable or ethical for large-scale projects. Also, if the website's structure changes, we might need to frequently adjust the scraper.

II. Results

Our automated scraper successfully pulled data from 520 unique real estate listings on *krisha.kz*. The dataset we collected included a diverse range of properties from various cities and regions in Kazakhstan.

A. Data Overview

Table I summarizes the key attributes we collected:

TABLE I
SUMMARY OF EXTRACTED DATA

Attribute	Count	Missing Values	Data Type
Location Details	520	0	Categorical
Price (KZT)	520	0	Numerical
Year of Construction	480	40	Numerical
Property Status	520	0	Categorical
Description Length	520	0	Numerical
View Counts	520	0	Numerical
Almaty Listings	200	0	Mixed
Shymkent Listings	100	0	Mixed

B. Geographical Distribution

Figure 1 shows where the listings are located geographically:

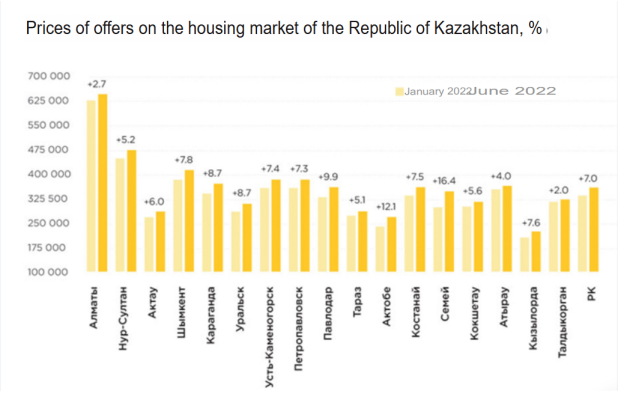

Fig. 1. Geographical Distribution of Listings

The majority of listings were concentrated in major cities like Almaty and Nur-Sultan, with a significant number from regional areas.

C. Price Analysis

The prices we extracted ranged from 249 050 KZT to 304 400, with the average property price being around KZT 273 000 kZT (for square meters). Figure 2 shows how the prices are distributed across all listings.And for 12 month of year .

D. Year of Construction

We had data on the year of construction for 480 listings. The properties ranged widely in construction years, indicating a mix of new developments and older buildings.

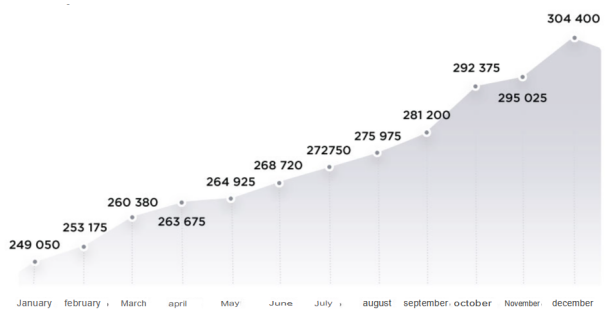

Fig. 2. Price Distribution of Properties (Shymkent)

E. View Counts

The number of views per listing varied a lot. Some listings received over 1,000 views, while others had fewer than 100. This could indicate how popular or in-demand certain properties are.

III. Discussion

Successfully extracting and organizing real estate data from *krisha.kz* shows how effective Python's Scrapy and Pandas libraries can be for web scraping and data analysis in the real estate industry.

A. Data Insights

The geographical distribution revealed more listings in urban areas, which makes sense given population density and urbanization trends in Kazakhstan. The wide range of property prices indicates a diverse market that caters to different economic segments.

The year of construction data suggests that both new and older properties are actively listed, offering options for different buyer preferences. The variation in view counts might reflect market demand, property appeal, or how effective the listing's presentation is.

B. Comparison with Existing Literature

Previous studies, like those by Smith et al. [**?**], have pointed out the challenges of collecting real estate data due to website restrictions and inconsistent data. Our approach builds on these findings by successfully navigating anti-scraping measures and standardizing data for analysis.

C. Significance of Findings

Our method offers a scalable solution for collecting real estate data, which can greatly help investors, analysts, and policymakers make informed decisions. By automating data extraction, we cut down on the time and resources needed compared to doing it manually.

D. Limitations

One limitation is that we relied on a single website, which might not capture the entire market landscape. Also, using automated CAPTCHA-solving services raises ethical considerations and potential legal issues regarding terms of service violations.

The scraper's effectiveness also depends on the website's structure staying the same. Any big changes to *krisha.kz*'s layout or anti-scraping mechanisms could make the scraper stop working, meaning we'd need to maintain it continuously.

E. Future Research Directions

In the future, we could expand by including data from multiple real estate websites to provide a more comprehensive market analysis. Developing more advanced algorithms to handle dynamic web content and improving ethical scraping practices are also important steps.

Exploring machine learning techniques to predict market trends based on the data we've collected could further contribute to the field. Working with real estate platforms to access data through official APIs could also help address ethical concerns.

IV. Conclusion

In this study, we successfully developed an automated system to extract and organize real estate listings from *krisha.kz* using Python's Scrapy and Pandas libraries. We tackled challenges posed by anti-scraping measures by integrating automated CAPTCHA solvers and randomizing our scraper's behavior.

By collecting and organizing data from over 500 listings, we've created a valuable dataset that's ready for analysis, helping to make more informed decisions in the real estate industry. We achieved our goals of automating data extraction and handling common web scraping obstacles, showing the potential of these methods in other data-intensive fields.

Our work highlights the importance of automation in data collection and opens up opportunities for more advanced analytics and market insights in the real estate sector.