Investigating the Illicit Trade of Cultural Property with an Automated Data Pipeline Architecture

Nicholas Landi, Elizabeth Lee, Karolina Naranjo-Velasco, and Felipe Barraza
School of Data Science, University of Virginia, Charlottesville, United States
nyl4gw@virginia.edu, ewl3dv@virginia.edu, kn3cs@virginia.edu, fb7yg@virginia.edu

***Abstract*—The scale of global art crime has been difficult to quantify due to the vast number of transactions and varying methods of trade. Although online marketplace platforms such as eBay offer promising data to study and track this illicit market, this relationship has not been systematically studied due to the highly technical nature of compiling and wrangling these data. This research project partners with the Cultural Resilience Informatics and Analysis (CURIA) Lab to design a robust data pipeline that collects, processes, and stores data from eBay to quantify and analyze the network mobility of illicit cultural property. The data pipeline consists of a template for accessing eBay's API, understanding API documentation, and collecting necessary features for network analysis. This process represents the first data pipeline architecture to our knowledge that collects data from listings across categories of interest, and stores features in a SQLite database through an automated, recursive script for social science research. The metadata for building and maintaining the data pipeline is recorded in an in-depth guide. The result of this data pipeline framework is a replicable blueprint for interacting with an online marketplace's API environment. This project will act as a precursor to begin research regarding the global trade of illicit cultural property through subsequent network and spatial analysis.**

***Index Terms*—Data Pipeline, Automated, Data Engineering, Database, End-to-End**

I. Introduction

The growing demand of the legal market for antiquities has been fueling the looting of historically protected sites by criminal actors [1]. Although there have been decades of qualitative research on this topic [2], there has been limited quantitative research due to the data-intensive nature of the network. The global illicit trades market is increasingly active on online marketplaces, such as eBay [3], which presents an opportunity to collect large amounts of data on postings and transactions.

Studies on the illicit trades market have been largely qualitative, with some research centered around the market value of stolen items [4]. Also, in the art crime community, the movement of individual artifacts is complex with illicit goods often being listed a significant amount of time after they are stolen. There often lies a time lag between the point where goods are seized and listed on the Internet for sale [3]. This makes it difficult to monitor illegal activity related to cultural heritage and motivates the implementation of an automated pipeline that can keep track of listings over long periods of time. This data pipeline allows for data engineering and social science to collaborate and research the trade market of cultural objects.

In collaboration with the Cultural Resilience Informatics and Analysis (CURIA) lab [5], this paper presents an automated data pipeline connected to eBay's Application Programming Interface (API) that extracts, processes, and stores data from categories that likely contain illicit antiquities and goods. The following sections will discuss the essential components of building this automated data collection pipeline that extracts listings of antiquities from eBay's online marketplace environment. Additionally, a theoretical framework of the selected eBay APIs and the ethical challenges of collecting the data is explained. Finally, the key insights from the data are presented, including potential biases, results, and use cases of this pipeline.

II. Background

The UNESCO 1970 Convention on the Means of Prohibiting and Preventing the Illicit Import, Export, and Transfer of Ownership of Cultural Property - hereinafter the UNESCO 1970 Convention – created a framework to prohibit and prevent the illicit trafficking of cultural property that it is illegally excavated from a heritage site or existing collection, among others forms of spoilage. The convention requires states to develop policies to restrict illicit cultural trade by regulating different forms of acquisition - transactions - of such objects. However, cross-border internet transactions through online platforms such as eBay have highlighted the challenges in monitoring and regulating cross-national transactions in a massive global market [4] [6]. Similarly, scholars Fabiani and Marrone [2], have argued that the lack of observable data limits the study of the market dynamics for archaeological objects, diminishing the measurability of actual damages caused by illegal forms of trade on cultural heritage. The marketization of cultural property has been a focal point of interest among scholars studying how the online marketplace shapes sales. A critical component missing in this conversation is a large-scale analysis seeking to understand market trends on popular selling platforms such as eBay [3]. Another barrier to the understanding of the phenomenon is a lack of computational tools available to researchers to systematically understand the nature of the cultural market. Hence, building a foundation to study the illicit online market can potentially provide insights into this industry [1].

The Internet market in antiquities has grown into a sophisticated and diversified commercial operation. Online transactions make it financially viable to trade in low-value and potentially high-volume material while also bringing geographically distant buyers [3]. Previous qualitative research on the online illicit markets have focused on eBay. We choose to build on research on this platform, rather than other marketplaces such as Facebook Marketplace or Craigslist, as this electronic commerce is most known by social science researchers to permit the trade of illicit goods [3].

Existing tools to access the eBay's Application Programming Interface (API) system [7] come from individualized wrapper packages created by dedicated development teams. While code for these wrappers is readily available online, many of them suffer from deprecation issues, and they serve only to answer specific business cases. A lack of homogeneity in the programming languages used to develop these wrappers also presents another barrier to less experienced researchers.

Additionally, eBay's system has allowed third parties to access various data sources and applications with the intent of helping sellers manage their eBay businesses at scale. Expansion and deprecation issues resulted in a revamp of the developers' program in 2016, which delivered a new family of modern and consistent RESTful APIs [8] . However, dense and inconsistent documentation has hindered widespread adoption of these web services for research use cases [9], and the developers program still primarily targets existing companies looking to obtain strategic business advantages [10].

Lastly, some provenance studies have manually leveraged scraping methods to generate data from antiquities sales catalogs [15]. Traditional forms of data extraction introduce the possibility of human error while also requiring extensive labor and time resources.

The developed pipeline aims to alleviate chronic issues of data paucity by aggregating information related to online vendors who sell cultural property on eBay into a single, multipurpose data repository. It intends to replace the need for human intervention by recursively running a data collection script every 24 hours.

The automated nature of this process enables the user to capture listings from a customized set of target categories. Due to API call limits, the pipeline has been structured to bias listings posted within the last 24 hours. However, if allowed to run long enough, this data collection method will provide a holistic snapshot of the market environment over a user-defined period of interest. The Pythonic wrapper presented in this proposal, while robust, offers flexible usage through customizable feature parameters. Supporting meta documentation presents a replicable system that can be readily integrated by other research groups in order to interface with eBay and other online market platforms.

III. Data Pipeline Architecture

The goal of this project was to create an automated data pipeline that captured relevant features from eBay listings for social science research. The data collected is stored in a database that is accessible through a high-level dashboard created with the Datasette open-source tool [11]. The general pipeline architecture involves building a Flask app to support an API, narrowing down on eBay APIs of interest, feature selection, database creation, and maintenance. The automated portion of the data pipeline runs every 24 hours collecting data from predefined categories of interest while being in compliance with call limits. As shown in Figure 1, after the initial set-up of the pipeline is completed, a Simple Linux Utility for Resource Management (SLURM) workload manager works to collect data daily.

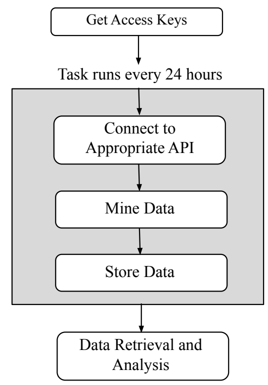

Fig. 1. General data pipeline architecture connecting eBay to database

A. Data Description

The data pipeline architecture from eBay to data storage is formed with the goal of social science research in mind. Currently, most use-cases of eBay's APIs and developer's program are for merchants and consumers looking to manage their marketplace and purchases [8]. There does not exist a pipeline, to our knowledge, that extracts data from eBay to track the illicit trades market. This required extensive research on feature selection. eBay's developer's program contains many APIs, each containing data relevant to different marketplace performance indicators or listing information. Due to the nature of the project focusing on listing information instead of marketplace performance indicators, we focus on the Finding API and Shopping API to collect features relevant to tracking the network mobility of illicit cultural properties. These variables were selected to provide comprehensive qualitative and quantitative data on items listed under predefined categories of interest. The features selected have the goal of identifying the listing and potential trade of illicit cultural properties.

Table 1 is a final list of features selected, as well as a description of the corresponding feature. The data type listed in parenthesis is the data type going into the database.

B. Key Parameters: Categories and Time

Data collection began by defining the categories of interest to social research. eBay sellers and users alike use categories to identify goods that are important to them. By defining categories of interest, we are limiting the data collected to

TABLE I
DATA DESCRIPTION

Feature	Description (Type)
Item_ID	ID that is uniquely generated by eBay when an item is listed. Typically 9-12 digits in length. (Integer)
Product_Title	Item name as it appears in the listing title, or in the search and browse results. (String)
CategoryID	ID that uniquely identifies the main category associated with the item listing. (Integer)
Price	A listing's current price converted to US dollars. (Float)
Item_Condition	A list showing the name and unique ID of the item's condition. (String)
Listing_Time	GMT time stamp of when listing is made available on eBay. (DateTime)
Item_Specifics	Item specifics key-value pairs for a listing. Keys vary depending on the listing and listing category. (String)
Seller_ID	Hash code encrypted ID that identifies unique sellers. (String)
Country	Two-letter ISO 3166 country code to indicate the country where the item is located. (String)
Zip_Code	Postal code indicating where the item is listed from. Format varies by seller. (String)
Image_URL	The URL of a specific listing on eBay. (String)
SKU	Unique SKU (Stock Keeping Unit) which identifies items, not the listing. (String)

categories where listings will likely contain illicit cultural antiquities.

We use category IDs as the first of two key parameters when building the automated data pipeline. Categories on eBay exist in a nested manner. For example, the "Antiques" category might have subcategories such as "Asian Antiques" and "Tapestries". We use a combination of overarching parent categories and child subcategories to create a list of categories as comprehensive as possible that encompass the trade of antiquities. eBay's categories also have a corresponding category ID. The category ID is an unique numerical identifier for each category that is consistent throughout eBay. However, category IDs might change on a rare occasion if eBay deems that category to be growing or deprecating. While each item can theoretically belong in many categories, each listing is associated with a primary category and category ID that is not limited to parent or child categories. This means that when parsing through eBay listings with categories, there will be no duplicate listings across different categories.

The second parameter necessary for accessing relevant eBay data is time. The goal of this project is to collect data across categories within a specific time-frame. Thus, a time parameter is included as an input when running the data collection script every 24 hours. This time parameter allows the corresponding API to collect data from listings listed only in the previous 24 hours at 12:00 a.m. EST.

C. Methodology

The script of this pipeline was written in Python with automation put in place with a batch script submitted to Slurm. Before writing the script of this data pipeline, we had to get access keys to eBay's developer's program. eBay's developer's program gives users direct access to tap into eBay marketplace and listing data. Access to eBay's developer's program gives a platform that streamlines the data collection process while being in compliance with eBay user privacy agreements.

Access to eBay's developer's program requires users to build their own application. We constructed a Flask app that provides eBay with a communication channel to notify developers of marketplace deletion notifications. The application is able to receive eBay challenge code and respond with an output that hashes together the challenge code, the user's eBay verification token, and the application's endpoint. This application creates an API that is paired with a domain that can handle Hyper Text Transfer Protocol Secure (HTTPS) requests and communicate with eBay.

After access is established, the eBay Finding API and Shopping API are used to collect features of interest. First, we used the Finding API to filter across categories of interest in a specific timeframe. The keys parameters, categories and time, are implemented as an input in the findItemsbyCategory call under the Finding API. Call types under APIs are a particular way in accessing the same data. We use the findItemsbyCategory call because it allows us to loop through our predefined categories of interest. The Finding API outputs variables of interest including Item_ID, Product_Title, Country, Price, and Zip_Code. By collecting data on these variables, we are able to identify unique listings and what the object is. These features are outputted in JSON format, so we convert them to a dataframe, which is an intuitive format for social science research. We also include the corresponding CategoryID as a column.

While the variables from the Finding API are insightful, there were missing gaps for data that social science is interested in. A unique seller identifier and other descriptive information about the details of the object cannot be found in this API. Thus, we use the Shopping API to get more data corresponding to the listings extracted from the Finding API. To create an end-to-end data pipeline, we use the Item_ID feature from the Finding API as our parameter input for the Shopping API. This allows us to collect data for listings we already decided were important — based on category — from the Finding API.

Similar to the Finding API, the Shopping API also contains different calls that make for accessing data based on different inputs. The GetMultipleItems call takes a list Item_ID as input, and outputs features that further identify and describe objects from listings. Features from the Shopping API include Item_Condition, Listing_Time, Item_Specifics, Seller_ID, Image_URL, and SKU. Similarly to the Finding API, these features are outputted in JSON format. We clean the output and concatenate them to the dataframe created from the Finding API. A description of each feature in our final dataframe can be found in Table 1.

A limitation faced in data collection under eBay's developer's program is call limits. These call limits limit the number of times we can send a request to eBay to collect data each day, resetting at 12:00 a.m. PST. Both APIs implemented have a call limit of 5000 calls. Based on the architecture of our pipeline, one Finding API call occurs for each page of data for a specific category. For the Shopping API, the GetMultipleItems call allows us to get data for up to 20 items with one call to eBay. This implementation allows us to maximize the number of observations we get in a single day. Although we are maximizing the number of calls, some categories have more new listings each day than our call limits allow us to collect data for. We use a stratified sample by category to collect data across all categories each day. Depending on the number of new listings each day, our data pipeline collects about 5000 new observations each day.

D. Data Automation, Storage, and Retrieval

The data collection process is automated by a batch script. We submit an sbatch file to the Slurm workload manager where can can run the python script that makes calls to the eBay API and cleans the data. The slurm script runs on a daily basis set up by parameters including beginning run time at 12:00 a.m. EST, and the resulting output — a SQLite database. Once the new data has successfully been appended to the database table, all changes are saved and the connection to the database is closed. This process is repeated the next day when call limits are reset.

The cleaned data frame from the features collected from the Finding and Shopping API are connected to a SQLite database. This database contains two entities. The first entity is updated daily with new listings from our data pipeline. This entity, named Item_Specs, contains all the features defined in Table 1. The second entity titled Category_Info has the attributes Category_Name and CategoryID for reference. We note that the database does not exist in a normalized or optimized form. However, for the case of social research this simple relationship allows social scientists with limited data extraction knowledge to access the data.

In order to make the database more accessible for applied social science researchers, the data stored in the database is also accessible through a high-level dashboard created with the Datasette open-source tool. This tool creates a general drop-down menu that allows for filtering of the data based on columns of interest. With this module, the resulting table can be downloaded as a CSV file or JSON file. This allows social scientists to use a more intuitive system to access the data.

The python script, sbatch script, resulting database, and Datasette module all exist on a high performance computing system. This central location allows for results and analysis to be deployed flexibly.

E. Metadata Documentation

To make the pipeline accessible, replicable, and applicable to social science research, a detailed documentation was created. The documentation discusses the general framework of building an eBay API and the specific use-cases of this pipeline in further detail. The metadata provides information on choices made in building the data pipeline. The documentation also discusses ethical implications of this project, including API and eBay-specific compliance. The comprehensive documentation is hosted on GitHub Pages [12] as a JupyterBook [13]. The link to the completed metadata documentation can be found here: https://nmicp.github.io/ebay_api/1_chapter.html [14].

IV. Results

The automated data pipeline architecture and the metadata documentation were the main products in this project. This data pipeline allows social science researchers to collect larger amounts of data than what is currently obtainable. The final database also allows researchers to narrow in on specific data of interest. To explore preliminary relationships and gain insight into the collected features, we performed exploratory data analysis. This analysis showcases how the final database can be used by researchers to extract data of interest.

To create a more intuitive interface to query the data from the database, we use the Datasette module. We are interested in listings of antiques that might contain more illicit goods than other categories. The following use-case of the database demonstrates how to alter the query to refine the collected listings.

We first select the eBay listing database and search along with the item category. For instance, the Greek Coins are under Category ID 4738. The search can be further refined by the seller's residence. In this use-case, we refined searches that matched a United Kingdom (GB) country code.

Fig. 2. Example Query from Datasette Module

The search above can be restricted to listings with their price below or at a maximum of 500 USD. All prices in the database are converted to USD. In addition to selecting an amount of money, a search can be filter a range of times. A filter for dates greater than 2022-01-01 and less than 2022-03-01 limits the time range from January 1st, 2022, to March 1st, 2022. Results will reflect the listings specified in that range and exclude those not in the selection. Executing this query produces results, such as that in Figure 2.

While the database is not normalized, the Item_Specifics column can be used to refine the listing based on more detailed information. String parsing and regular expressions can be applied to search for a specific string in the text that determines the intended characteristic of the listing. Databases do not allow complex data types, such as dictionaries, to exist as entries; thus, the data was flattened into strings that can be parsed. Parsing strings is beneficial because keywords can quickly identify the selected listing, providing more comprehensive information such as descriptions and values.

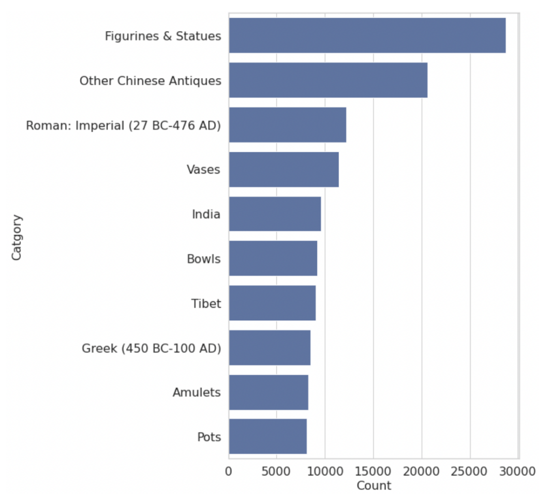

Fig. 3. Distribution of Top 10 Categories in Database as of April 7, 2022

Executing queries through SQL or an interface such as Datasette allows listings to be filtered and retrieved in a tabular format. This data can be then saved to a file or joined to another table for further analysis such as network analysis.

We also perform some exploratory data analysis to get some general insights of the data collected. In Figure 3, we see the distribution of top 10 categories in the comprehensive database that is updated daily. The nature of our data collection process samples from predefined categories of interest every 24 hours. Based on this plot, we can see that some categories such as "Figurines and Statues" might get many more new listings as compared to categories such as "Amulets" and "Pots". The categories with a smaller total count of listings also often child categories, which highlights that more listings have a primary category of a parent category.

Further, the illicit cultural trade market is a global phenomenon [1]. Figure 4, representing the frequency counts of top 5 countries in the database, gives a better representation of this trade. From Figure 4, the top 5 countries with listings in the database are the United States, China, Japan, United Kingdom, and Thailand. This highlights a bias in the data collection process. This bias can be defined as a bias from the predefined categories of interest we set. Different categories selected will likely result in a different distribution of country counts. On the other hand, this bias might also be attributed to an innate bias due to the nature of eBay. eBay is a marketplace platform based in the United States, so there might exist a home-country bias when looking at it's users and ultimately listing country [16].

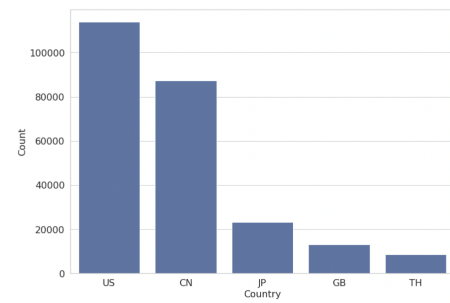

Fig. 4. Frequency Counts of Top 5 Countries as of April 7, 2022

Exploratory graphs can show preliminary relationships in the data collected without running any analysis. This exploratory stage will help set the stage for future works such as network analysis or other social science research. The results of our research show that the automated data pipeline has opportunities for research that can be looked at both quantitatively through features such as price, or qualitatively through an text search on features like item specifics. The data pipeline will help those interested in marketplace research, such as in the scope of illicit trade, to gather more data for a more comprehensive research.

V. Discussion

The end-to-end data pipeline architecture that collects data from relevant eBay APIs and stores this data in a database allows for the quantitative research of illicit cultural property trade. Previously, a lack of observable data has limited comprehensive and quantitative studies of the underlying market dynamics influencing the online trade of cultural property. In that vein, the pipeline we have outlined remedies this chronic issue of data paucity by generating a centralized data set that interested third parties can publicly access.

Moreover, this architecture aligns with the development of new research techniques, in particular, machine learning algorithms. Applications that require large amounts of observable data, such as deep learning, can be trained to solve specific use cases by leveraging the data available in our growing repository. For example, machine recognition techniques can be applied to the Image URL parameter, which would allow researchers to systematically track the life cycle of an antique based on the recurrence of an item's image. Features, such as Price, CategoryID, and Country, can be isolated and trained to reveal the distribution of different characteristics associated with antiquities across the online marketplace. Additionally, Natural Language Processing (NLP) can be applied to reveal patterns in the title of antiquities, which can be expanded upon to predict the item's country of origin or the transaction's authenticity.

Due to the challenges of the deprecation of existing tools to interface with eBay's API system and a lack of homogeneity in programming languages utilized to build these applications, this pipeline was built to provide researchers with customizable, end-to-end data extraction method. The automated nature of this program replaces traditional data collection methods, reducing the possibility of human error, time, and labor resources to eliminate barriers to third parties.

The code documentation to build an automated data pipeline from eBay is publicly available, allowing third parties to alter parameters in the request calls or interface with different APIs. The underlying pipeline architecture can also be used as a foundation to study other online marketplace environments. The pipeline, supplemented by detailed meta documentation, acts to further bridge the gap between social scientists and the growing field of data science.

VI. Conclusion

In this paper, we have illustrated the construction of a robust data pipeline for capturing information of antiquities on eBay's online market environment. By developing our application, we have successfully interfaced with eBay's Finding and Shopping APIs to extract data of up to 5,000 listings a day from over 40 categories of interest. While this pipeline is an automated process of collecting listings from predefined categories of interest, it does not denote which goods are actually illicit. In order to determine which goods are illicit, researchers such as archaeologists can use the data from this pipeline to begin identifying illicit goods. The automated nature of this pipeline creates an updated database for investigation and discussion where antiquities and illicit goods likely exist. The data pipeline architecture also does not capture previous listings of items that have already been on sale either on eBay or other environments. This highlights a bias in the dataset from when the pipeline began collecting data. Although bias and limitations exist in the data pipeline, this architecture forms a link between the large amounts of data and social science research. Researchers can use the structure of this pipeline for future projects that use Machine Learning to identify illicit and repeated listings.

This growing repository has been made publicly accessible through integration with the Datasette open-source tool. This is the first end-to-end architecture developed to research online market environments, and the automated system is intended for widespread usage in the research community, where it will replace more traditional data collection methods. This pipeline is also supplemented with extensive meta documentation, which entails the process of constructing a Flask application, connecting to eBay's API system, and extracting and cleaning data. This meta documentation was collated so that other researchers could both replicate and build upon our work to begin the process of studying other marketplace environments. The resulting end-to-end automated data pipeline architecture links data science with social science research, and presents opportunities for more comprehensive studies on the online marketplace network interactions.

Acknowledgment

The authors would like to thank Professor Jonathan Kropko for his continuous feedback and support as an advisor throughout this project. The authors would also like to thank Professor Fiona Greenland and Professor Michelle Fabiani for sponsoring this capstone project and providing their subject-area expertise when necessary. Finally, the authors would like to thank Jacalyn Huband, Ph.D., Computational Research Support Specialist at the University of Virginia, for her support in navigating high-performance computing.

References

[1] M. Altaweel, "The Market for Heritage: Evidence From eBay Using Natural Language Processing," Social Science Computer Review, vol. 39, no. 3, pp. 391–415, Jun. 2021, doi: 10.1177/0894439319871015. [Accessed March 31, 2022].

[2] Fabiani M.D., Marrone J.V. (2021) Transiting Through the Antiquities Market. In: Oosterman N., Yates D. (eds) Crime and Art. Studies in Art, Heritage, Law and the Market, vol 1. Springer, Cham.

[3] M. Altaweel and T. G. Hadjitofi, "The sale of heritage on eBay: Market trends and cultural value," Big Data Society, vol. 7, no. 2, p. 205395172096886, Jul. 2020, doi: 10.1177/2053951720968865.[Accessed April 1, 2022].

[4] F. Greenland, J. V. Marrone, O. Topçuoğlu, and T. Vorderstrasse, "A Site-Level Market Model of the Antiquities Trade," Int J Cult Prop, vol. 26, no. 1, pp. 21–47, Feb. 2019, doi: 10.1017/S0940739119000018. [Accessed April 1, 2022]

[5] The Cultural Resilience Informatics and Analysis Lab (CURIA Lab). [Online]. Available: https://curialab.org/

[6] N. Brodie, M. M. Kersel, S. Mackenzie, I. Sabrine, E. Smith, and D. Yates, "Why there is still an illicit trade in cultural objects and what we can do about it," Journal of Field Archaeology, vol. 47, no. 2, pp. 117–130, 2021. doi: 10.1080/00934690.2021.1996979.[Accessed April 4, 2022].

[7] eBay's Developer Program. [Online]. Available: https://developer.ebay.com/

[8] R. Rischpater. eBay Application Development. Apress, 2004.

[9] M. Altaweel. (2020). eBay Scraper [Source code]. https://github.com/maltaweel/eBayScraper

[10] A. Hochstein, A. Schwinn and W. Brenner, "Business Opportunities with Web Services in the Case of Ebay," 2009 42nd Hawaii International Conference on System Sciences, 2009, pp. 1-7, doi: 10.1109/HICSS.2009.100.

[11] "Datasette: An open source multi-tool for exploring and publishing data," Datasette v0.61.1. [Online]. Available: https://datasette.io/. [Accessed: 07-Apr-2022].

[12] GitHub Pages. [Online]. Available: https://pages.github.com/. [Accessed: 07-Apr-2022].

[13] "Books with Jupyter," Jupyter Book. [Online]. Available: https://jupyterbook.org/intro.html. [Accessed: 07-Apr-2022].

[14] "EBay's API Data Pipeline," eBay's API Data Pipeline - eBay's API Data Pipeline. [Online]. Available: https://nmicp.github.io/ebay_api/1_chapter.html. [Accessed: 08-Apr-2022]

[15] Yakutskiy, A. Ruskevych, V., Somfai, M. (2015) Ebay Api Bundle [Source code]. https://github.com/WebConsul/EbayApiBundle

[16] A. Hortaçsu, F. Asís Martínez-Jerez, J. Douglas. "The Geography of Trade on eBay and Mercado Libre," working paper, The Networks, Electronic Commerce, and Telecommunications (NET) Institute, 2006 [Online]. Available: https://archive.nyu.edu/bitstream/2451/28452/2/Hortacsu_06-09.pdf