ETL Process Plan

Phone Data Harvest: scraping and analysing mobile device prices

Group 2 (EN)

Daniel Herrera (daniel.herrerarussert@stud.hslu.ch)

Ramon Burkhard (alainramon.burkhard@stud.hslu.ch)

Jack Brown (jack.brown@stud.hslu.ch)

Module: CIP (Data Collection, Integration and Preprocessing), F24

March 14, 2024

Contents

1. Introduction and Motivations

Project ideation

During the initial brainstorming phase for our project topic, we investigated several data sources. We explored the possibility of extracting data from retail giants like Amazon, MediaMarkt, and Galaxus. We also considered analysing trends on social media platforms such as Facebook and Instagram. Additionally, we looked into sports activity data from reputable sources like Wikipedia, Olympics.com, and laliga.com.

After some consideration, we decided that delving into data associated with retail companies would be more interesting for the entire group. At this point, the focus shifted to deciding where and what to scrape. Initially, we contemplated utilizing the "Mediamarkt" group as our project source, motivated by its widespread presence across different European markets. This approach would enable us to analyze diverse country markets, including Spain, Germany, and Switzerland.

An alternative idea within the same theme emerged: exclusively scraping data from the Swiss market using platforms like "Galaxus.ch," "Mediamarkt.ch," and "Interdiscount.ch." Ultimately, we opted for the latter, specifically sourcing data from Switzerland domains. The main reason behind this decision lay in the similar structures of data presentation across domains of the "Mediamarkt" group. This would influence the group to adopt a uniform approach, consequently, the data manipulation process would be nearly identical.

Opting for domains from various companies proved more intriguing, as each student would encounter distinct structures, potentially leading to varied challenges. This diversity is expected to enhance the overall learning experience for the group.

Regarding the "What?" aspect, the group unanimously decided to focus on scraping data related to notebooks or smartphones. This decision was straightforward, driven by the observation that smartphones, across all pages, offered more diversity in models, quantity, ratings, and reviews. Additionally, smartphones are ubiquitous devices with a wide range of features and specifications, making them an ideal choice for data analysis.

Contextualization of the Project

To provide context for our project and facilitate its development, give meaning to our inquiries, and justify our decisions, we present a story below:

The "TipTopClub" is an application and platform that charges a monthly or yearly fee from its users. It offers registered users the convenience of searching for products in various categories such as sports equipment, books, construction tools, office materials, and electronics. The platform then presents search results, already highlighting the best options based on specified parameters (price, specifications, delivery time, ratings, and reviews). Users can select a product from the results, complete their request, and the "TipTopClub" takes charge by placing the order on behalf of the user. This eliminates the need for users to register on multiple websites and manage an address book. Another advantage of the club is the option to participate in collective orders with other members, potentially reducing costs. The club then handles the separation and dispatches the products to the user.

In this context, the "TipTopClub" has tasked its data science team with delving into the next product category listed in their database: smartphones. To fulfil this request, the board has asked for a preliminary study from the data science team, focusing on extracting data from three different websites to evaluate the project's viability.

The project should encompass a comprehensive report detailing all technical procedures for scraping the data, conducting data analysis, and developing a small program. This program should be capable of comparing and returning the best phone result in the data frame based on specifications provided by the user input.

Context diagram

3 Sources -> Extraction -> Transformation -> Storage

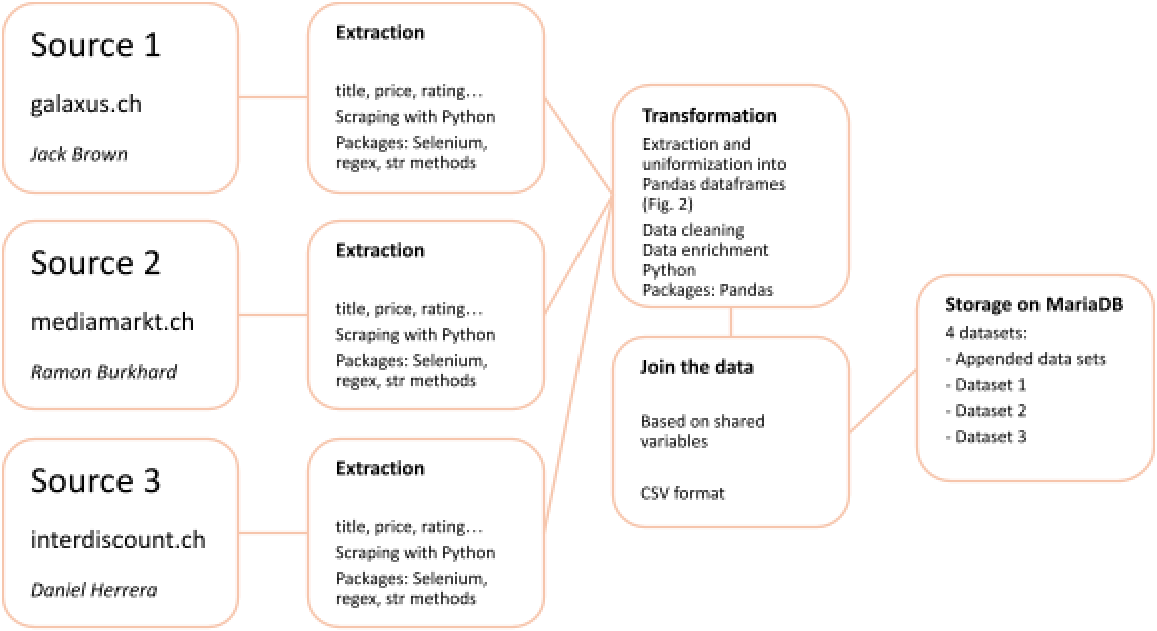


*Figure 1 SEQ Figure * ARABIC 1: Context Diagram for ETL Process*

Sources

We plan on scraping data from three Switzerland based websites (dynamic web pages): galaxus.ch, mediamarkt.ch, and interdiscount.ch. These platforms offer an immense variety of phones, with information like make, model specifications, delivery time, ratings, and type of sale (new or refurbished). With this data, we aim to enable users to find the best product given their prioritized specifications (e.g. best price and largest phone) while incorporating aspects like review ratings and delivery time. The data can be merged by phone maker and corresponding attributes.

galaxus.ch:

For our analysis, we use (https://www.galaxus.ch/en/s1/producttype/smartphones-24?

take=). This provides necessary filters to set us onto the phones page. Galaxus is one of the biggest online electronic sellers in Switzerland ("Galaxus," n.d.) which means we will have a wide variety of phone data to extract. To be exact, as of March 14, 2024, there are currently 2,087 products that can be scraped.

After inspecting Galaxus's robot.txt file, crawlers like sogou spider, Yandex, eventmachine httpclient, and niki-bot are not allowed to scrape any url with "/" in it (does not apply to us). We also wouldn't be allowed to access things like /cert, /Files, or /management (which also does not apply to us), so nothing will stop us from scraping this website.

mediamarkt.ch:

Mediamarkt is the second website to be scraped. Mediamarkt is a German multinational chain of stores selling consumer electronics with over 1000 stores in ten countries in Europe ("Mediamarkt," n.d.). In Switzerland, it operates 25 physical stores and an online commerce platform.

Due to its extensive capillarity in the European market, Mediamarkt is a highly competitive retailer, offering prices comparable to the Swiss market. Regarding smartphones, the website "mediamarkt.ch" contains 354 new units from 17 different brands listed in its dedicated smartphone category (https://www.mediamarkt.ch/de/category/_smartphone-680815.html).

Additionally, it offers approximately 1440 used smartphones, available in the refurbished smartphones section (https://refurbished.mediamarkt.ch/ch_de/unsere-refurbished-smartphones).

The robots.txt file for the website disallows web crawling and indexing. However, since our intention does not involve either of these actions, we are compliant with the website's policies.

interdiscount.ch:

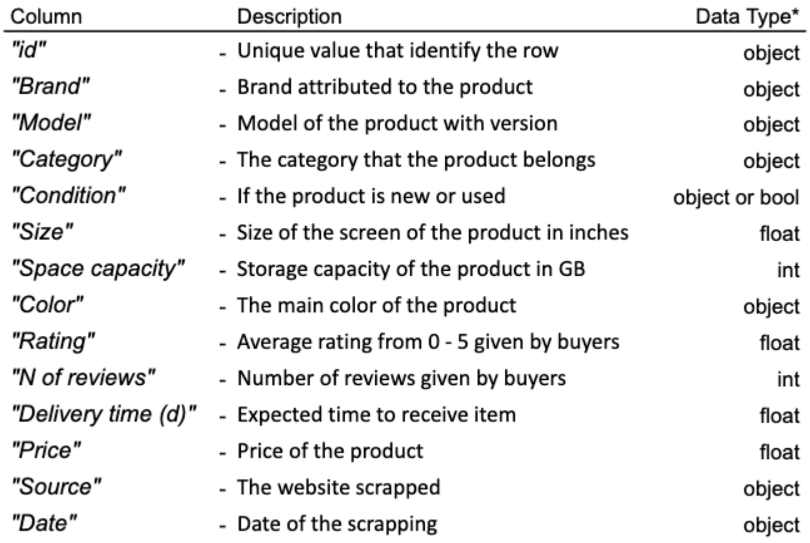
As a third data source, we refer to interdiscount.ch, which according to the own portal is 'the leading Swiss retailer for consumer electronics, offering a vast selection of top brands and new trends at competitive prices'. They boast the most extensive network in Switzerland with roughly 170 stores and a user-friendly online shop ("Interdiscount," n.d.).

In order to access their offer in mobile devices available on the website we use the URL address 'https://www.interdiscount.ch/de/search?search=Smartphone', in a similar manner to the two other retailers. As of March 14, 2024, there are 26'136 items available on the website, which also includes accessories and other phone-related components. A further step will be to reduce this source to only the mobile devices.

Expected Result (CSV)

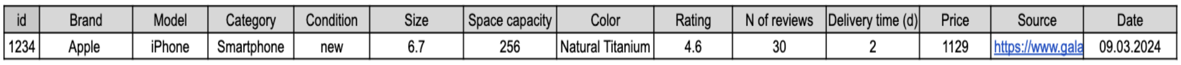
In this section we present our preliminary expected structure of the CSV uploaded from each group member in MariaDB. The structure will be used as a parameter for all group members. the idea behind it is that the group members after scraping and the data manipulation can achieve a data frame structured in this way.

Table 1: Column Description



Column	Description	Data Type*
"id"	- Unique value that identify the row	object
"Brand"	- Brand attributed to the product	object
"Model"	- Model of the product with version	object
"Category"	- The category that the product belongs	object
"Condition"	- If the product is new or used	object or bool
"Size"	- Size of the screen of the product in inches	float
"Space capacity"	- Storage capacity of the product in GB	int
"Color"	- The main color of the product	object
"Rating"	- Average rating from 0 - 5 given by buyers	float
"N of reviews"	- Number of reviews given by buyers	int
"Delivery time (d)"	- Expected time to receive item	float
"Price"	- Price of the product	float
"Source"	- The website scrapped	object
"Date"	- Date of the scrapping	object

* All data types referring to Python language



id	Brand	Model	Category	Condition	Size	Space capacity	Color	Rating	N of reviews	Delivery time (d)	Price	Source	Date
1234	Apple	iPhone	Smartphone	new	6.7	256	Natural Titanium	4.6	30	2	1129	https://www.gala	09.03.2024

Figure 2: Data Frame

After merging the data from the three retailers we expect to have a data frame of the size 4000 rows x 14 columns

Research questions

Based on the already explained premise, we set ourselves obtain insights through the ETL process which may provide answers for the following research questions:

- How consistent is the data across different resellers?
  A first observational assessment of the data from different sources may provide insights into which phone-related specs or information items are relevant to each retailer.

- How do electronic product prices vary across different platforms?
  The most straightforward observation would be a direct comparison of pricing for the same models or a combination of specs.

- Can we identify patterns in consumer ratings and preferences based on reseller of choice?
  By looking at user ratings and related data, we could narrow down on possible correlations between pricing and buyer-satisfaction.

Process Description and Challenges

Since our project revolves around scraping phone data, it would be challenging to find a pre-existing CSV dataset to merge with ours. To address this, each group member will be responsible for scraping a webpage. We have chosen to conduct our entire process on a virtual machine running Linux. Our team utilizes a GitHub repository to share information and collaborate effectively. The primary tools we will use include PyCharm and Jupyter Lab.

Given that our target webpages are dynamic, we have opted to use the Selenium package for web scraping employing Chrome WebDriver and Firefox WebDriver, parsing data using Class, XPath and CSS selectors (Figure 3,4 and 5 appendix). While most of the required data can be collected from the landing page, including brand, model, version, and price, we anticipate needing to scrape information from individual webpages to obtain details such as delivery time, reviews, availability, and ratings. All this process can be very resource-consuming to the servers scrapped, in respect of that we plan introducing delays in code, such as using '*time.sleep()'.*

During the transformation phase, we will employ regex (Figure 6 appendix) and regular functions for string manipulation, including methods like strip() and split() (Figure 7 appendix). We will create a dictionary from the scraped data (Figure 8 appendix), manipulate it as needed, and adjust data types and structure to align with our expected result CSV format shown in Figure 2 - Data Frame.

During the load phase, we will independently load our CSV files into MariaDB. Since we expect that at the end of the transformation phase, we will have similarly structured datasets, we will join our dataframes by appending one dataframe to another. Additionally, we will reset our ID column as part of this process.

Finally we will analyse the final dataframe, using data analysing procedures, using pandas for dataframe manipulation, visualization libraries and a cloud visualization tool (Streamlit) to illustrated part of our analysis.

We have outlined the proposed ETL (Extract, Transform, Load) workflow and undertaken a preliminary assessment of the websites chosen. It is foreseeable that challenges may arise during the scraping procedure. Since the main objective of our project is to collect data pertaining to the topics at hand, a solution in the event of unexpected issues would be to switch to a different source of data. A consequent additional challenge would be the identification of an adequate replacement website, which considering the broad topic of the project and the abundance of mobile phone resellers, should be straightforward.

Some additional challenges have been identified in the dynamic nature of these online resellers, with fluctuations of price data and ratings over time. Also, the consideration of foreign websites adds another layer of dynamic behavior, with variations in currency exchange data, and differences in pricing between countries. This is however a predictable circumstance and partly the focus of our research questions.

As far as the characteristics of the data, naming conventions of variables and strings could differ within one dataset, as well as between sets from different portals and countries. With some thorough and systematic data cleaning, all three sources could be set to an equivalent standard of quality. This will also provide a significant aid for the next stage of the project, when all three sources are merged into one larger data set. An advisable approach would be to establish standardizations for variable naming and model-specific contents and rating codes, which will be elaborately described in a chapter of the final report. Only with a systematic review and standardization of the data can all three sources be acceptably combined and compared on equal terms.

Reference list

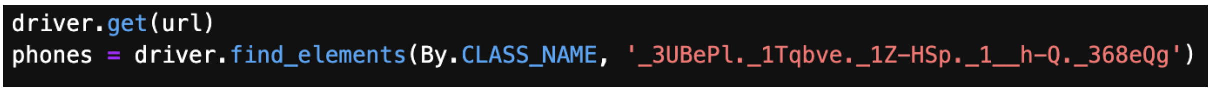
Galaxus. (n.d.). In Wikipedia. Retrieved March 9, 2024, from https://en.wikipedia.org/wiki/Digitec_Galaxus

Mediamarkt. (n.d.). In Wikipedia. Retrieved March 9, 2024, from https://en.wikipedia.org/wiki/MediaMarkt

Interdiscount. (n.d.). In Wikipedia. Retrieved March 9, 2024, from https://de.wikipedia.org/wiki/Interdiscount

List of Tables and Figures

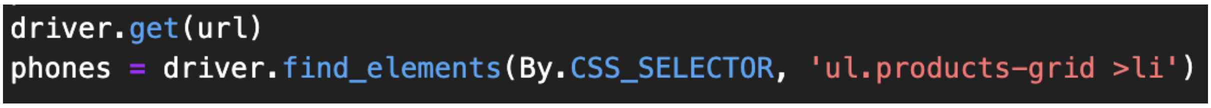
Figure 3: Class Selector



```
driver.get(url)
phones = driver.find_elements(By.CLASS_NAME, '_3UBePl._1Tqbve._1Z-HSp._1__h-Q._368eQg')
```

source: Code snippet from author

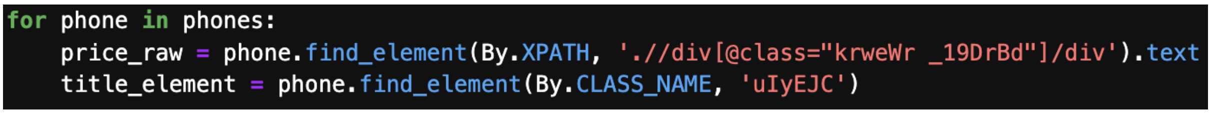
Figure 4: CSS Selector



```
driver.get(url)
phones = driver.find_elements(By.CSS_SELECTOR, 'ul.products-grid >li')
```

source: Code snippet from author

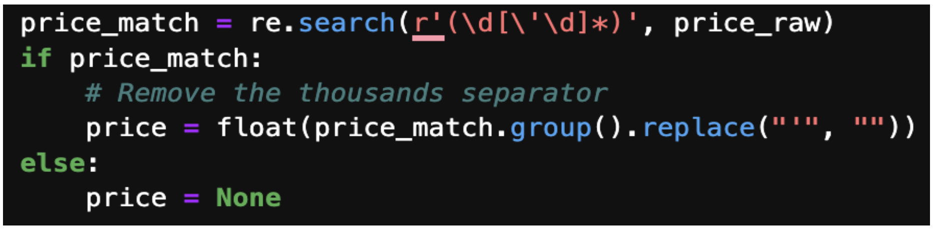
Figure 5: Xpath Selector



```
for phone in phones:
    price_raw = phone.find_element(By.XPATH, './/div[@class="krweWr _19DrBd"]/div').text
    title_element = phone.find_element(By.CLASS_NAME, 'uIyEJC')
```

source: Code snippet from author

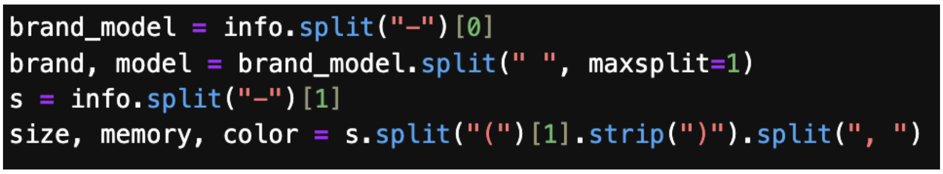
Figure 6: Regex



```
price_match = re.search(r'(\d[\'\d]*)', price_raw)
if price_match:
    # Remove the thousands separator
    price = float(price_match.group().replace("'", ""))
else:
    price = None
```

source: Code snippet from Autor

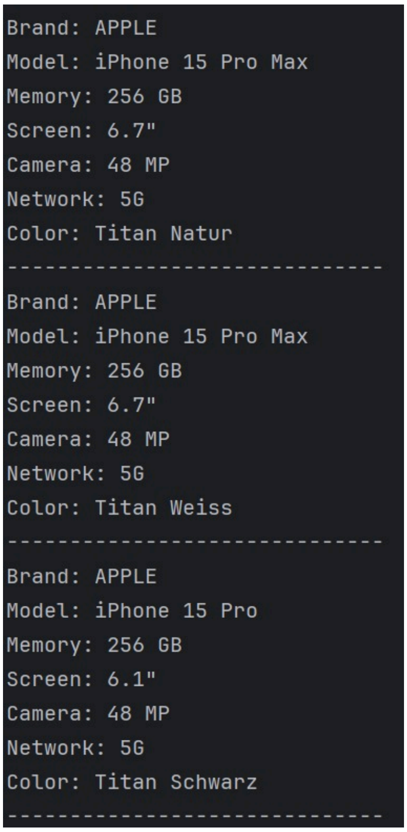
Figure 7: Str mehods



```
brand_model = info.split("–")[0]
brand, model = brand_model.split(" ", maxsplit=1)
s = info.split("–")[1]
size, memory, color = s.split("(")[1].strip(")").split(", ")
```

source: Code snippet from author

Figure 8: Dict data



```
Brand: APPLE
Model: iPhone 15 Pro Max
Memory: 256 GB
Screen: 6.7"
Camera: 48 MP
Network: 5G
Color: Titan Natur
------------------------------
Brand: APPLE
Model: iPhone 15 Pro Max
Memory: 256 GB
Screen: 6.7"
Camera: 48 MP
Network: 5G
Color: Titan Weiss
------------------------------
Brand: APPLE
Model: iPhone 15 Pro
Memory: 256 GB
Screen: 6.1"
Camera: 48 MP
Network: 5G
Color: Titan Schwarz
------------------------------
```

source: code snippet from author