oxylabs®

Building a Competitor Intelligence System for E-commerce

Competitor intelligence at a glance 3

Competitor Intelligence explained 3

Competitor intelligence architecture 4

Collecting data 5

Web scraping 6

Setting up proxies 6

Creating a fingerprint 7

Sending an HTTP request 8

Parsing 9

Data cleaning and normalization 10

Data analysis 11

Summary 12

Competitor intelligence at a glance

Evaluating the competition has long been ingrained within the essentials of gaining an advantage over competing firms. Yet, not all parts of the competitor analysis are clear-cut. What factors should you consider if you wish to determine the effectiveness of an existing company? Market share, price, and similar features may be an obvious choice, but there are other intricacies to be considered. These, less visible, factors could still play massive roles within various echelons of business.

Competitor intelligence comes into play precisely at this stage, where the need to combine these intricacies and make accurate estimations arises. It overlooks both public information and otherwise unpublished data (within legal limits) so that accurate analysis can be made. The idea of competitor intelligence is to create a "portrait" of the businesses you're competing with.

Competitor Intelligence explained

The purpose of competitor intelligence is deeply valuable, as it is the fundamental foundation upon which opportunities, challenges, and overall business environments are examined. Yet, as mentioned above, it includes not only readily available but also obscure information, making creating a "portrait" difficult.

As such, this whitepaper will focus on both obvious and vague data available on the internet. As a whole, a competitor intelligence system can be roughly divided into four parts (a more detailed analysis is provided in the architecture schema):

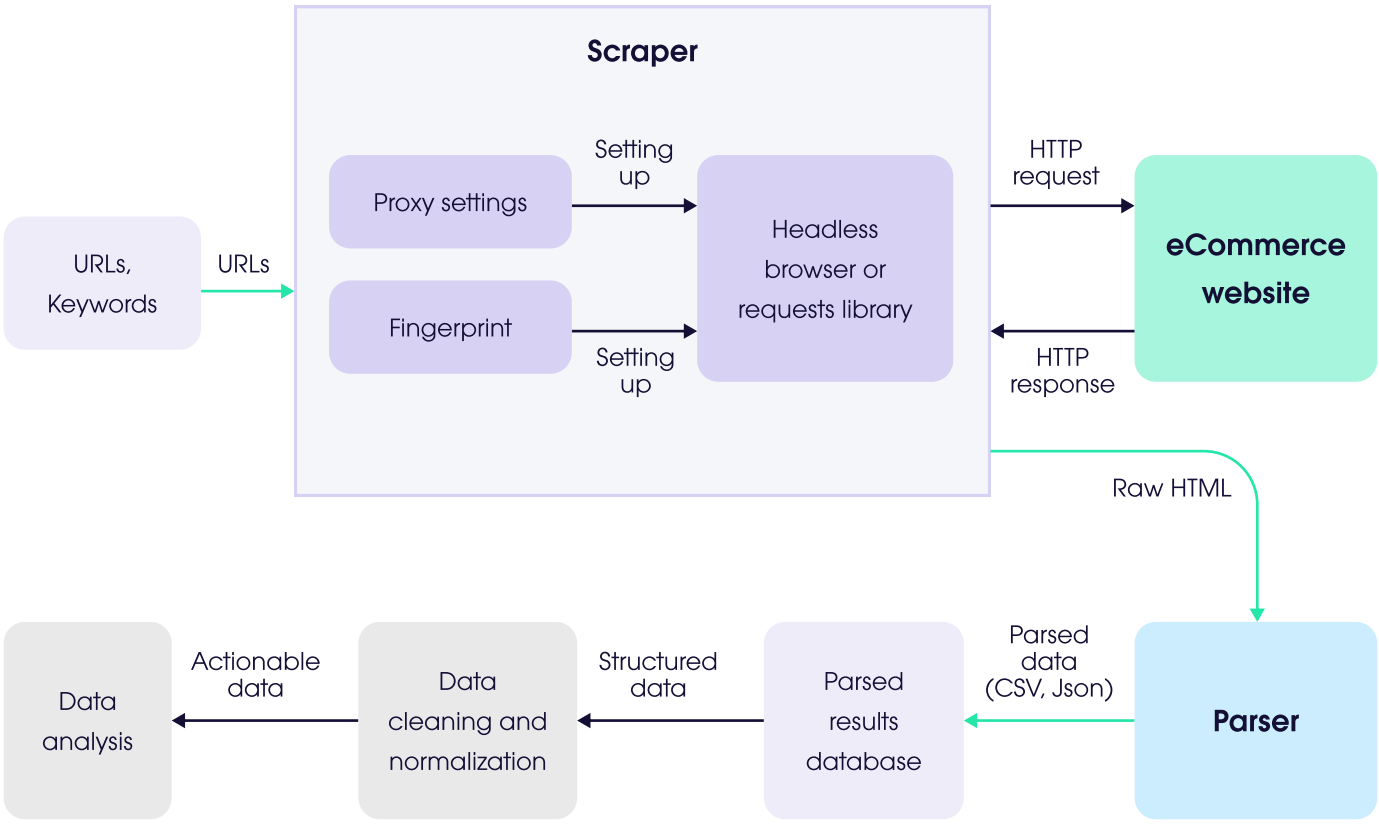
Competitor Intelligence consists of four principal steps:

1. Gathering the data
2. Setting up the system
3. Analyzing the information received
4. Making a decision based on the formatted data

Competitor intelligence architecture

There are five primary steps within a competitor intelligence system. Additionally, each action has various sub-actions, which may vary greatly depending on the purpose of your task and available tools.

1. Collecting data (such as target URLs or keywords)
2. Web scraping
3. Data parsing
4. Data cleaning and normalization
5. Data analysis



Note that the architecture outlined above is highly customizable. You can choose your preferred approach, whether that's automated solutions, hands-on work, or a combination of both.

Collecting data

What is relevant data varies on a case-by-case basis. In most scenarios, it is likely to be keywords and/or URLs, though, in practice, a pre-analysis must be conducted.

Within the said analysis, you should consider certain factors. Evaluate websites so you can deduce how they can be scraped, what ding of bot detection systems they use, how the data is being fetched from the back-end, what proxy works the best for you, and lastly, which factors are crucial for your competitor intelligence project.

Web scraping

Configuring a web scraper involves three steps:

1. Setting up proxies
2. Creating a fingerprint
3. Sending an HTTP request

Setting up proxies

Within the largest E-commerce sites, a variety of data, such as price, is provided based on current location. If a site version is not international, then country or city-level targeting can be conducted (the decision to do so has to be made in the Collecting data part) since proxy providers cover most locations worldwide. The said location should be chosen when targeting a specific E-commerce site.

Furthermore, the need for the above-mentioned virtual connection requires using proxies. Whether that is a more limited location coverage, as seen in Datacenter Proxies, or the globally expansive locations offered by Residential Proxies, one of the two must be chosen.

Let's imagine the priorities of a specific task are lightning-fast scraping speeds and large data volumes. In such a case, Datacenter proxies would likely be the preferred option as they can handle even the largest of data loads. It is assumed, however, that the target website is not capable of blocking multiple traffic instances originating from a singular subnet or cannot detect the IP/Proxy type itself by changing geolocation databases for types.

If the primary concerns are the avoidance of blocks, flagging, and bans, then Residential proxies would be the recommended choice. Due to them being recognized as household devices and their ISP-provided IPs, a website would find it significantly more difficult to detect such a proxy since it closely resembles genuine traffic, ensuring a high level of concealment for any scraping task.

Furthermore, Residential proxies are able to cover nearly the whole world, offering a wide array of locations and subnets, while Datacenter proxies are mainly clustered in and around densely populated areas.

Three prominent aspects must be considered when choosing which one is for you. What's more important is the price and the performance of datacenter proxies, or the anti-scraping avoidance of residential proxies essential for you.

Finally, consider setting up multiple parallel scraping tasks, timing sessions, utilizing protocols (HTTP), and configuring proxy rotation. All of these considerations significantly increase the odds that your entire scraping process will be successful.

> **NOTE:** the exact details of how often proxies need to be configured, which type should be used, and similar issues are highly dependent on scraping targets, frequency of data extraction, and other factors. Proxy personalization is paramount.

Creating a fingerprint

While browsing the internet, there is a constant exchange of requests and responses between browsers and websites. These requests allow the server to understand what content is requested by the user and how it should be delivered. Parameters such as layout preferences, language settings, operating system, and device in use are all determined by the browser/client. At the same time, the server tries to deliver the closest approximation to the machine request.

Browser fingerprinting aims to track such information by going through users themselves. It gains responses based on HTTP headers, GPU and CPU specifications, and other data. This data is then combined and gathered into one larger picture, which then becomes the "digital fingerprint." Yet there are other priorities as well.

For web scraping, it is the creation of organic HTTP headers which emulate organic traffic. When visiting a website, a browser sends a set of HTTP requests to the server. Lastly, User-Agents are combined with associated headers, therefore ensuring the browser is sending successful requests.

Sending an HTTP request

The process starts with the need to make an HTTP request. As such, the URL, proxy settings, and headers should be sent through a request library to an E-commerce website. There are other ways of making an HTTP request, primarily through a vast number of headless browsers, which is quite an advantage since most popular browsers are also available in headless mode.

For your web scraping efforts, any support you can get is certainly beneficial. Crucially, most programming languages have in-depth HTTP libraries for making requests. Here are some popular ones for the most common web scraping languages:

- **JavaScript:** Request, Axios
- **Golang:** Colly
- **Python:** Requests, aiohttp
- **R:** rvest
- **PHP:** Goutte, Panther
- **Ruby:** HTTParty, Kimurai

NOTE: Selenium, an automation tool for browsers, works well with most of these languages.

If you wish to avoid making requests yourself and want more automation, then interacting and using an API (automated programming interface) would be recommended as it simplifies some of the time-consuming tasks. The primary benefits of Scraper APIs when compared to regular automation tools are:

- Easy scalability
- Lack of extra coding
- Already available tools (proxy rotator)
- 100% return rates per successful requests
- Automated web scraping process

If your scraping tasks encounter failure, check response evaluation, see that the proxy type/location is not being blocked, handle errors (settings, set up), and re-try. If no errors appear and the HTML file has been successfully extracted, it is then moved to a parser.

Parsing

Upon extraction, the HTML is structured, though it cannot be used for analysis yet, as specific information still needs to be extracted. A parser makes sense of all this information by pulling the HTML file from the scraper and forming it into an easily readable format such as JSON or CSV.

The structuration process should be tailored to parse the specific data elements by locating their HTML attributes from a particular E-commerce site. A parser determines what information from an HTML string is useful based on predefined rules.

Most E-commerce retailers display these data elements:

- Images
- Delivery time
- Title
- Price
- Discounted price
- Product code (SKU)
- Product URLs
- Related products
- Review ratings
- Stock availability

Parsers may require constant maintenance to deliver results. Primarily, this happens because E-commerce sites have frequent layout changes, among other reasons.

Factoring in which of the data elements are essential for your competitor intelligence system is key. Only then can a parser be configured in a way that provides desired results in a structured manner.

Data cleaning and normalization

Clarity of data is vital within competitor intelligence. Thus, a final step of quality control must be performed. It includes normalizing found data, currency conversions if necessary, and converting specific text-based data points to digits.

Lastly, all the formatted data can be passed on to further analysis, upon which major strategic competition-related plans can be drawn up.

Data analysis

The value of the entire process discussed above lies within the data analysis step. While, technically, the competitor intelligence system's functions stop at this stage, getting to data analysis is why creating such a system is often seen as vital.

A scientific article on the topic can provide evidence as to why this is the case. The paper uses automatic text summarization to gather textual data from various organizations. Authors conclude that an integrated system through which these texts were compiled allowed practicing managers to make more accurate competitive decisions, which is precisely the goal that a successful competitor intelligence system aims to achieve.

Summary

Competitor intelligence is often seen as the only way to stay on top of your advantages while creating accurate, data-centric plans. The amount of information one can gather on an organization is immense, but doing so effectively allows for creation of predictable business patterns.

Completing all actions shown in this white paper would provide you with competitor intelligence scraping software that efficiently collects E-commerce data. Such an intelligence system would allow you to extrapolate otherwise unseen insights.

We hope that the architecture shown in this paper will help you gather E-commerce insights. If there are sections or actions that you're still unsure of, we recommend referring to an expert who could assist in finding the optimal solution based on individual budgets and activities.

oxylabs®

Want to Know More?

If you would like to know more about any of the topics mentioned in this white paper or **learn about our products**, please get in touch! Our team is ready to answer any of your questions and offer you the best solution for your business needs.

Get in touch with Oxylabs

Our Mission

Our mission is to share all the know-how that we collected over the years in the industry in order to create the future where big data is accessible to all businesses. We seek to create a healthy environment for everyone to grow and thrive in.

Our Values

As a leading company in the proxy and web scraping industry, we ensure that the highest standards of business ethics lead all our operations. Our core values guide us toward achieving our mission. **Learn more**