Bachelor Project Plan

s204164

September 2023

1 Project Description

1.1 Motivation

1.1.1 Purpose

Usually a tool can be either flexible but complicated or simple to use but can only be used to specific use cases. In terms of web scraping a python library for web scraping is very flexible. It can be used to scrape almost any site. Unfortunately it requires the user to know programming, have knowledge about the library and spend a lot of time writing specialized code for each specific use case. On the other hand there are simple tools designed to let the user scrape data quickly without writing any code. The drawback is that these tools are not very flexible and cannot perform any scraping task. I believe that with the emergence of large language models that can write code, it will be possible to find a new middle ground. Given a web scraping task, by analyzing HTML code, a LLM can write specified scraping code for that task and website. The problem is that the HTML of a page cannot always be given directly to the model as part of the promt, because there is often too much HTML code. Also, it is often difficult to interpret the HTML code of a page, since elements are not always properly labelled. For example the title of a product on amazon has the class "a-size-medium a-color-base a-text-normal". The class doesn't descripe that this is a product title. This means chat-gpt and other LLM models are not always useful for web scraping. Therefore instead of just giving the text model the raw HTML code as a promt, it is necessary to build a scraping tool.

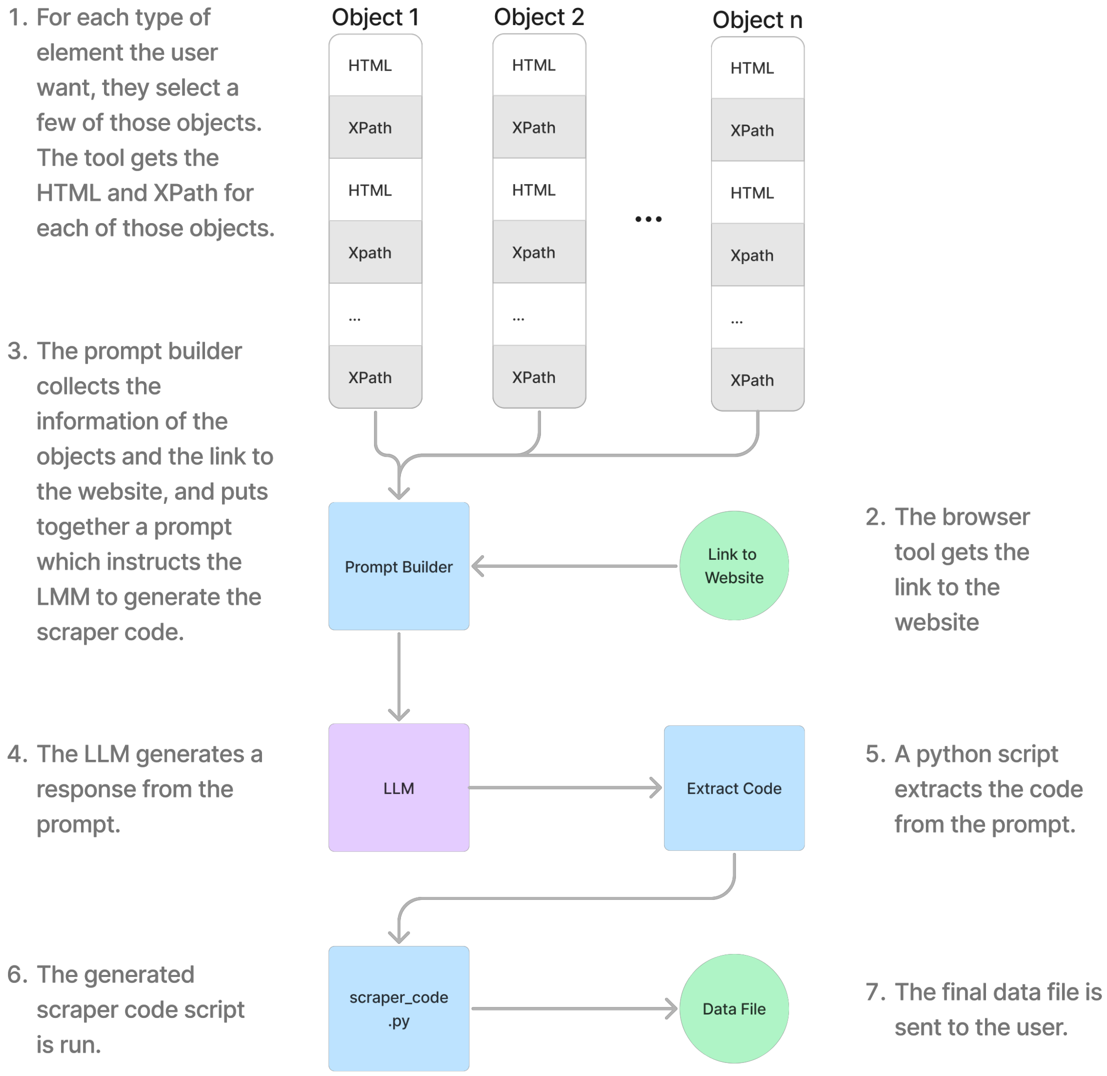
1.1.2 Target Customer

Small businesses might not have the resources to do advanced data collection. For them a web scraping tool which is simple yet still somewhat flexible would be useful for gaining insight into their competition and do market analysis.
Digital Marketing Professionals might not have a lot of knowledge about coding, but they still need to collect data. With this tool they can collect data for SEO analysis, content research, and customer research.
E-commerce Businesses would probably be the main target market for this web scraper. Online retailers need to monitor competitor prices, product reviews, and customer sentiment to adjust their strategies. Many people why are new in the field of e-commerce don't have great technical skills. Therefore a very intuitive web scraper would be useful to get them started.
Data Scientists and Machine Learning Engineers could use this tool to gather large datasets for training predictive models in less time. It would also be useful for students doing free-time projects.

1.1.3 Societal Impact

Democratization of Data: Large companies collect lots of data, which gives them a big advantage. If web scraping is made simpler it would help small companies and individuals get access to more data.
Innovation: Researchers and academics often need to collect a lot of data. This is especially important for fields like healthcare, finance, and social sciences.

1.1.4 Scraper Diagram



1. For each type of element the user want, they select a few of those objects. The tool gets the HTML and XPath for each of those objects.

2. The browser tool gets the link to the website

3. The prompt builder collects the information of the objects and the link to the website, and puts together a prompt which instructs the LMM to generate the scraper code.

4. The LLM generates a response from the prompt.

5. A python script extracts the code from the prompt.

6. The generated scraper code script is run.

7. The final data file is sent to the user.

1.2 Project Goals

1.2.1 Web Scraper Backend with OpenAI API

A simple program that will put together a prompt and feed it to a GPT model to generate a python script. Then it will run this script, to obtain the scraped data. Ideally the script should be able to do more than just analyzing the HTML code. For example it should be able to click around on the website. If there is time it should additionally be able to (legally) bypass CAPTCHA and other difficulties of scraping a website.

1.2.2 Evaluation of the Models

An evaluation dataset should be made for testing the capabilities of the scrapers. It should contain about 20 examples of web-sites and prompts. There should be easy tasks and difficult tasks. For each task there should also be a correct scraping script to compare with the results of the models.

1.2.3 Transfer Learning for Smaller LLM

Calling the OpenAI API every time a website should be scraped can be expensive. Therefore it would be a good idea to use a GPT model to generate a dataset of web scrapers. The dataset should contain a lot of prompts and web scrapers written from those prompts. This dataset can be used to fine-tune a smaller pre-trained LLM, which is small enough to run on a single computer.

1.2.4 Web Scraper Browser Extension (Optional)

This can be made if there is time left after making everything else. A browser extension for the web-scraper. The user should be able to click on examples of what they want to scrape, and give a text description of what should be scraped and how. This information will be sent to web scraper backend, where a promt will be generated.

2 Learning Objectives

2.1 General Learning Objectives for Bachelor Projects

- Understand the experiment as a source of new knowledge.
- Work independently and can structure a larger task, including adhering to schedules and organizing and planning the work.
- Summarize and interpret technical information and master technical problem-solving through project work.
- Master a technical language in Danish.
- Be able to work with all phases of a project, including proposal development, solution implementation, and documentation.
- Think systematically and be able to acquire new knowledge based on the ability to learn, read, and listen, and also critically assess acquired knowledge.
- Perform relevant and critical information retrieval and, based on that, find the appropriate methods to address the current problem.

2.2 Specific Learning Objectives for the Project

- Understand transfer learning.
- Understand fine-tuning.
- Understand the structure of transformer models and how they function.
- Can explain what web scraping is, what it is used for, and how it works.

3 Timeline

Name	Start Date	End Date
Research og få overblik	Sep 1	Sep 5
Projektplan	Sep 5	Sep 11
Prototype af webscraper backend	Sep 5	Sep 11
Evaluerings Dataset	Sep 11	Sep 25
Transfer Learning	Sep 18	Oct 16
Første Udkast af Rapport	Oct 16	Nov 13
Browser Extension	Nov 4	Nov 25
Bypass CAPTCHA	Nov 4	Nov 25
Færdig Rapport	Nov 14	Dec 15