Level 4 Progress Report

Tan Ming Sheng 2110022T

# Introduction

Forums have become an important resource on the web due to its increasing richness of user generated information through the contribution by millions of internet users on a daily basis, therefore harvesting forum data can allow analyst to discover useful information to help improve business intelligence. The aim of this project is to develop a real-time automatic crawler with a dashboard user interface to manage and monitor the crawler while it retrieves and store data from various forum sites distributed on the internet.

# Progress

For the first few weeks, I have been researching on related works of web crawlers, followed by studying of research papers that are related to forum crawler. After some review, I came out with a refined set of requirements to improvise and better integrate the dashboard and the crawler together by making it a web application which will be able to handle multiple client connections and each of them is able to monitor and manage their crawlers to crawl whichever forums/websites they want, returning the specific desired data in an organized manner back to the client.

Next, I went on to design and test out the overall architecture for weeks and it was one of the challenge I had because it was complicated. Finally, after numerous attempts, I managed to come out with an architecture integrating Django, Scrapy and MongoDB together in a multi-threaded and also a multi-process environment.

Currently for my progress, I have a planned architecture and also have already set up the necessary environment for the development of the crawler with a basic working prototype which is able to handle multiple client connections with multiple crawler processes. You can view the source code and my commits at this GitHub link: <https://github.com/mingsheng36/Focra>

# Problems

Some of the problems I have encountered during development were:

1. The lack of up-to-date complex technical resources

Although Scrapy’s website is very well documented, some of the more complex technical resources provided by other websites were all using older versions of Scrapy which has already since changed drastically.

1. Integrating Scrapy with Django

Scrapy has its own environment settings and it requires at least one or more processes of its own to operate, but to integrate it with Django environment and allowing different client process to communicate with their own Scrapy processes was hard.

1. Integrating MongoDB with Django

MongoDB is a document based database and therefore it is different from traditional relational database. This has caused some minor problems while trying to store data into MongoDB from Django because of the Object-Relational Mapping (ORM) design of Django.

# Next Step

For the remainder of the project, I will be improving from the basic prototype by implementing the core features into it which includes:

* Allow dynamic selection of data (xpath) to extract
* Provide template for crawling different kinds of website. (Pagination etc.)
* Allow scheduling of the crawler (when to crawl, how often to crawl)

If time permits, I will also implement these additional features:

* Controlling on the number of crawlers and CPU resource for each user
* Auto detection of website template
* Auto detection of structured data based on user input