NUCES FAST ISB, SOFTWARE ENGINEERING

NLP CS4063 Assignment 2

i201847-Daniyal Khan

September 21, 2023

1 Introduction

This LaTex document provides an explanation of the Python code for web scraping using Scrapy.

2 Code

```
import scrapy

#WE CAN REMOVE ANY OTHER YIELD TAGS, AND THAT WON'T Output any links and story names
# I have outputted the story names as well as their links for better manuevering.


class I201847UrduStoriesSpider(scrapy.Spider):
    name = "i201847_urdu_stories_spider"
    start_urls = ["https://www.urduzone.net"]

    def parse(self, response):
        # Extract story links from the main page
        stories = response.css('h3.entry-title')
        for story in stories:
            # Extract the link to the story
            link = story.css('a::attr(href)').get()

            # Create a new Scrapy request to follow the story link and invoke the parse_story callback
            yield scrapy.Request(link, callback=self.parse_story)

    def parse_story(self, response):
        # Extract the text inside the <p> element within the div with class "tdb-block-inner"
        paragraph_text = response.css('div.tdb-block-inner p::text').get()

        # Clean and yield the extracted data
        yield {
            'story_title': response.css('h1.entry-title::text').get(),  # Extract the story title
            'paragraph_text': paragraph_text.strip() if paragraph_text else None,  # Extracted paragraph text
            # Add more data extraction logic as needed
```

```
                }
```

3 Code Explanation

The provided Python code is for web scraping using Scrapy. It has two main functions:

- `parse` **Method:** This method finds links to articles using the appropriate CSS selector and then loops over these links to Create a new Scrapy request to follow the story link and invoke the `parse_story` parse_story callback.
- `parse_story` **Method:** This method Extract the text inside the ¡p¿ element within the div with class ”tdb-block-inner” and then Clean and yield the extracted data.

4 Data Storage

The Scrapped data of the articles is later on stored in a csv file in the scrapy project root directory.

5 Challenges Faced

- Using Scrapy for the first time, Elsewise done webscraping with be.requests and beautifulSoup.
- Understanding the format of the `www.urduzone.net` i.ie the website to be scrapped and finding the right css selectors to extract the data from.
- Filtering out the non-urdu words or characters from the extracted article text.
- Using LaTeX

6 Conclusion

1. We're creating a web scraping tool (a spider) that's going to visit a website called ”https://www.urduzone.net.”
2. First, the spider goes to the main page of the website and looks for stories listed there.
3. It finds the titles of these stories and the links that take you to the full story.
4. For each story, it clicks on the link to the full story to see what's inside.
5. Once it's inside a full story page, it looks for a specific part of the story, which is usually a paragraph of text.
6. It takes that paragraph of text and saves it along with the title of the story.
7. It does this for all the stories it finds and saves all the titles and paragraphs of text.
8. The spider continues doing this until it has gone through all the stories.
9. It's like reading all the stories on the website and jotting down the titles and a part of each story in a notebook.