**Data Pipeline**

**Introduction:**

The purpose of this project is to extract data from multiple sources, transform it, and then load it into a data storage solution. The targeted subject is "A.I News".

Extraction from .CSV files, .PDF files, and web scraping will be implemented.

1. **Data Ingestion:**

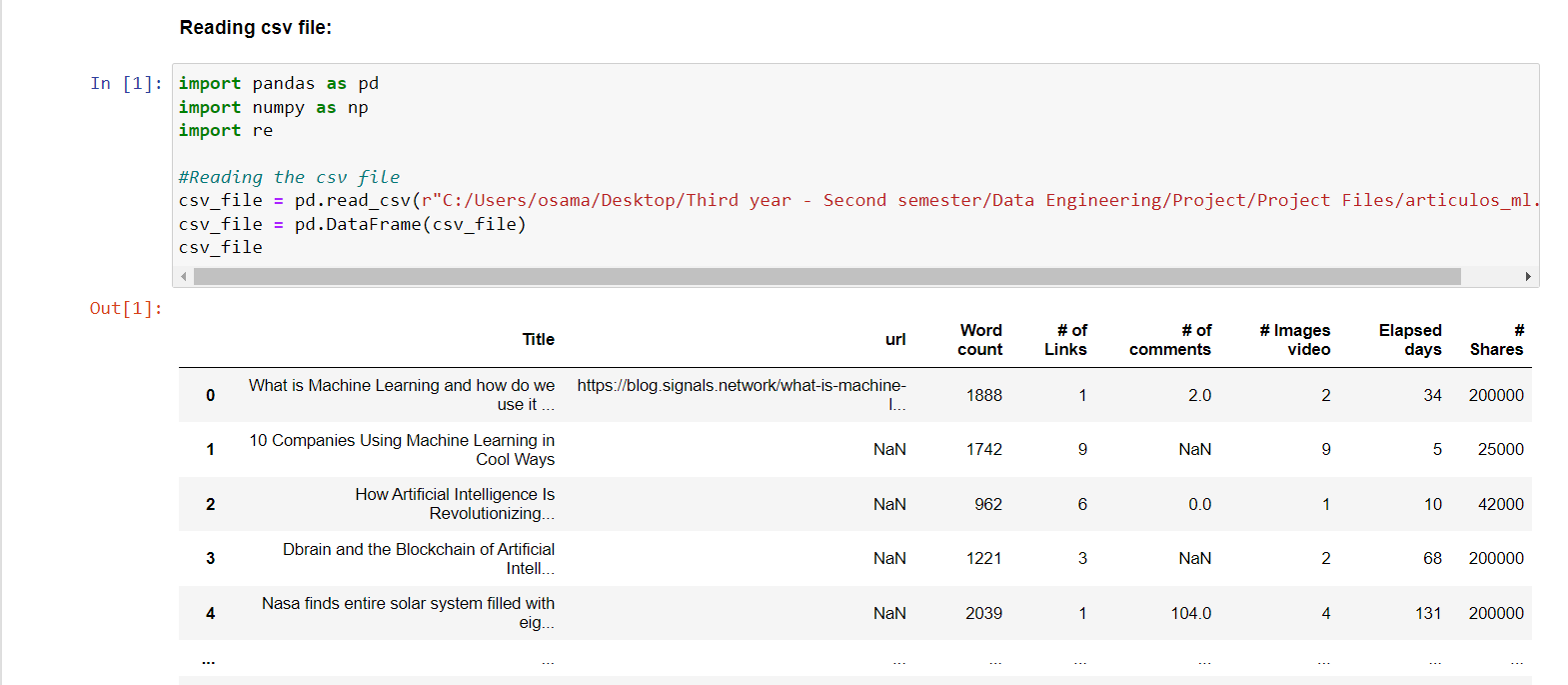
* **CSV Extraction:**

**File link:** https://github.com/jbagnato/machine-learning/blob/master/articulos\_ml.csv

The csv file used contained relevant and irrelevant data to A.I/A.I News, we're going to extract the relevant data.

**Figure 1:** Reading the csv file:

**Figure 1, Reading the csv file:**



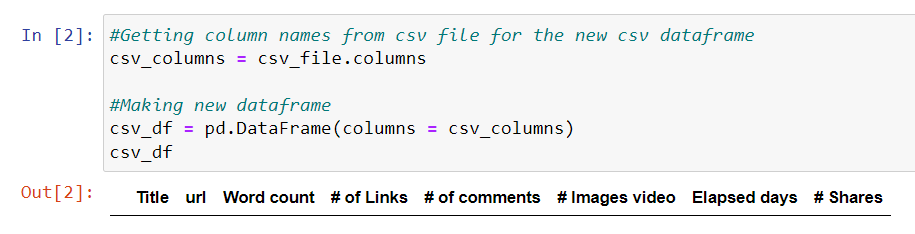
After reading the csv file we need to clean and transform the data, refer to the figures below.

**Figure 2**: Creating the data frame using the columns from the csv file, which we will use to save the transformed data.

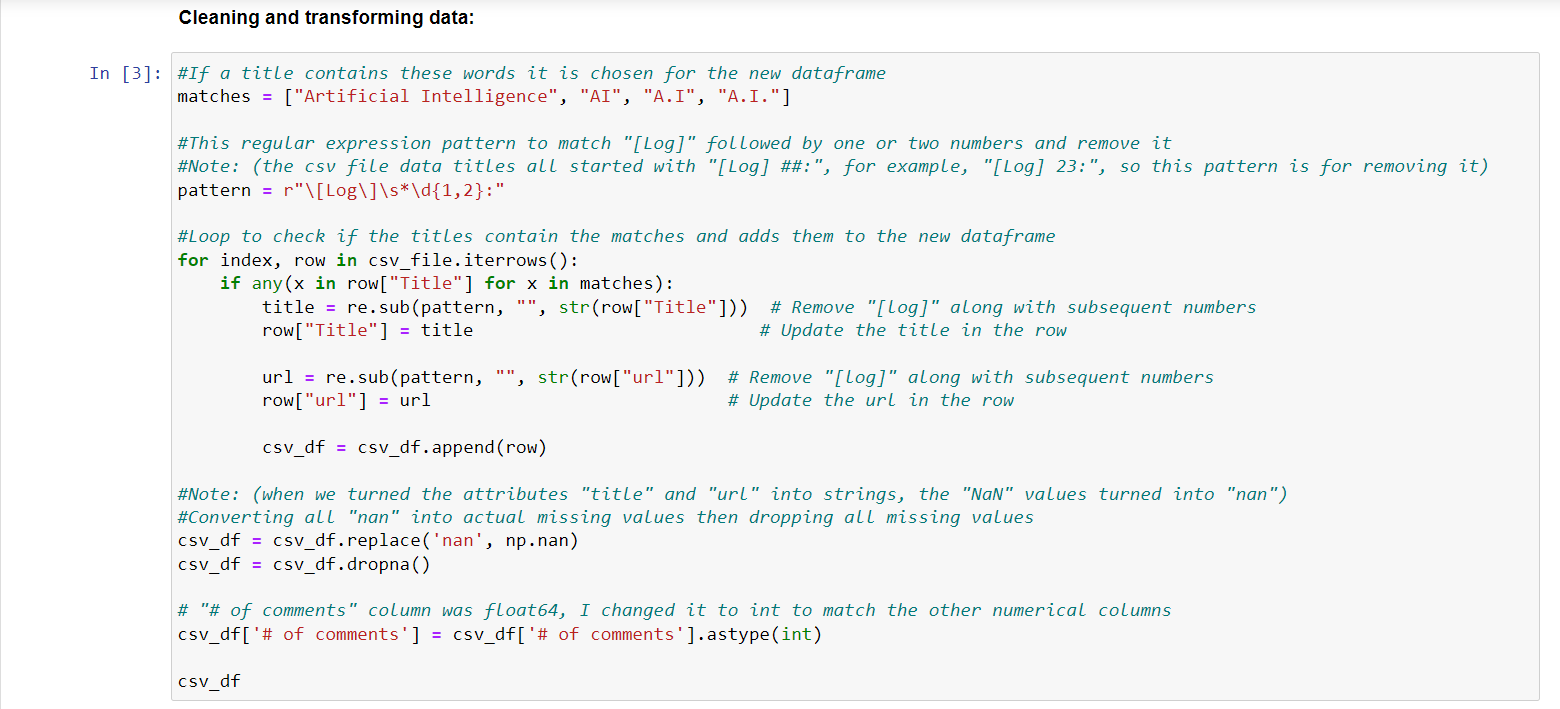
**Figure 3:** Cleaning and transforming the csv file data, the data contained missing values and inconsistencies.

**Figure 4:** Displays the data after transformation.

**Figure 2, Data frame creation.**



**Figure 3, Cleaning and transforming.**



**Figure 4, Data after cleaning and transformation.**

A screenshot of a computer

Description automatically generated with low confidence

* **Web Scraping Extraction:**

**Website:** <https://www.artificialintelligence-news.com/>

Used BeautifulSoup library to scrap from the website which contains A.I News articles, the data scraped from each article is: Title, Description, Date, Genre, URL.

**Figure 5:** Creating the data frame which will store the finalized data, and the temporary containers will be used to store the scraped data directly.

**Figure 6:** details of the functions used for web scraping.

**Figure 7:** The AiNews\_create() function is used once when initializing web scraping.

**Figure 8:** The AiNews\_add() function will be used later in a thread to continually add new data to the data frame.

**Figure 9:** The AiNews\_create() function is called to initialize web scraping, then the AiNews\_add() function repeats every 2 hours to keep the data up to date.

**Figure 5, data frame and containers creation.**



**Figure 6, functions used in extracting the data.**



A picture containing text, screenshot, font

Description automatically generated

**Figure 7, The AiNews Data frame creation function.**

A picture containing text, screenshot, font

Description automatically generated

**Figure 8, The AiNews adding function.**



**Figure 9, web scrape initialization and addition every 2 hours.**



* **PDF Extraction:**

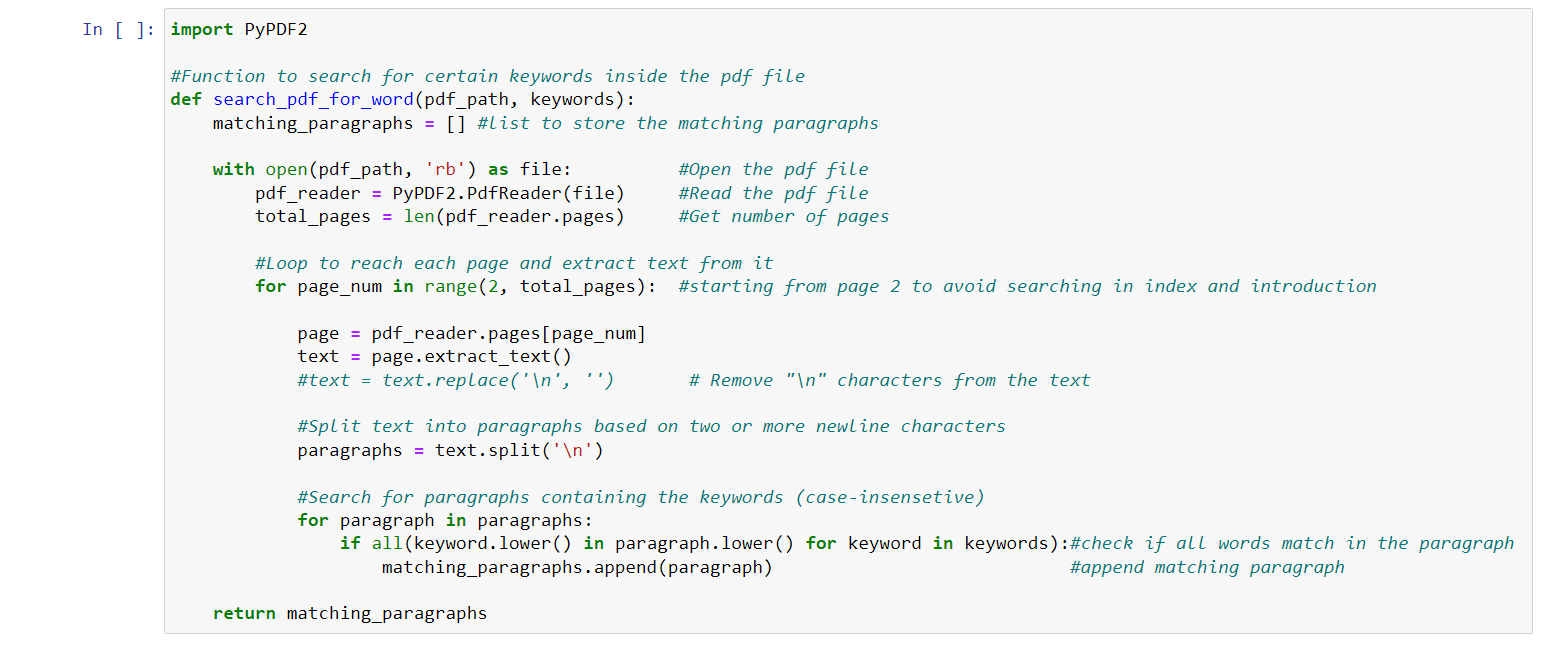
**File Link:** <https://www.tandfonline.com/doi/pdf/10.1080/21670811.2022.2063150>

The pdf file has information about A.I in the news, we’re going to extract the paragraphs which contain the words [“AI”, “News”] and store them in a data frame.

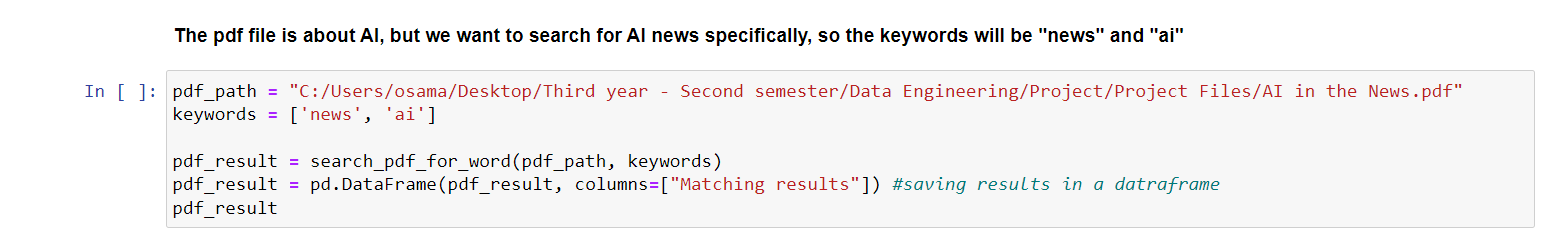
**Figure 10:** Used PyPDF2 to read a pdf file and then extract the relevant paragraphs to our subject (A.I news).

**Figure 11:** Called the search\_pdf\_for\_word() function with the desired pdf path and keyword, which we then stored the result of in a data frame.

**Figure 10, Function to search for matching words in paragraphs.**



**Figure 11, Calling the function using the pdf file path and the keywords we're searching for**



1. **Data Storage:**

* **Storing data in MongoDB:**

Used NoSQL MongoDB to store the data in separate collection depending on where they came from (CSV, web scraping, PDF).

**Figure 12:** Convert all data frames to dictionaries to be saved in JSON format inside MongoDB, and then connect to the MongoDB and store the data in the appropriate collection.

**Figure 12, Building model for data storing to MongoDB.**



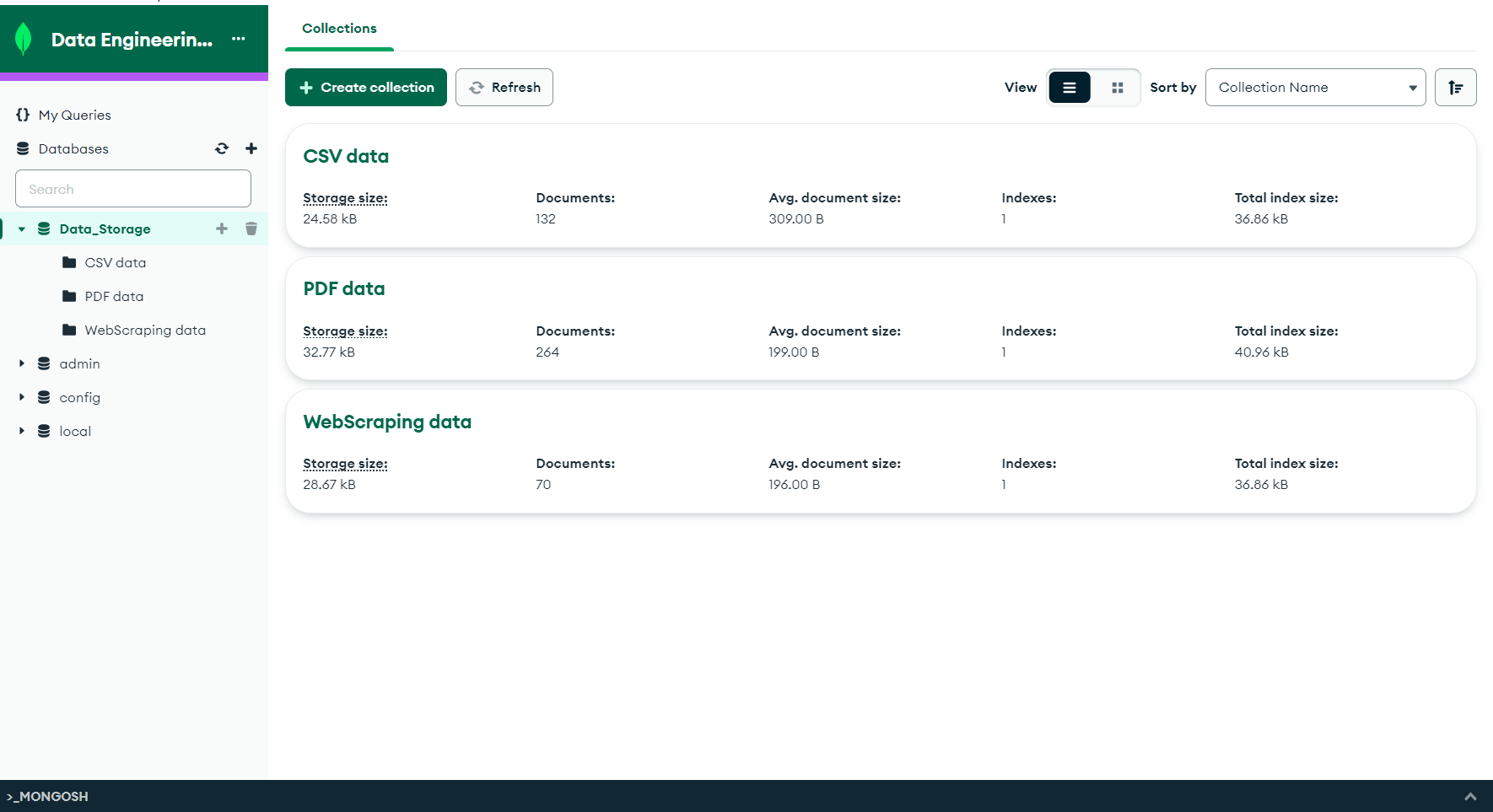
* **MongoDB:**

The database and collections are automatically made by the python code when you visit the MongoDB:

**Figure 13:** Overview of the database and collections.

**Figure 14:** Example of data inside the collection.

**Figure 13, Overview.**



**Figure 14, CSV Data extracted.**

