MA5851 Assignment 3, Document Two

April 25, 2021

1 Website Selection

1.1 Website Selection Justification

Only trusted Australian news sources were considered, which meant first examining their machinery of care, transparency, expertise and agenda (Caulfield, 2017). Furthermore, only news sources which provided publicly available data and consent for web crawling their data were pursued in order to remain ethical. The combination of these two factors significantly limited the pool of Australian news source candidates to consume to just one: https://www.onthehouse.com.au. Hence, the selected website for this Project was https://www.onthehouse.com.au. In terms of the layout of this website, each news article was displayed as a single block on the page and there were 136 property news pages at the time of this study.

1.2 Base Data Required

The base data required for predicting the relative strengths of the property market across all states and territories in Australia, as a function of time, was deemed to comprise of the publication date and entire body text of each property news article. Returning the entire body text allowed for the identification of all GPE and LOC data present as an input to the sentiment analysis and returning the publication date allowed for tracking of these sentiments over time.

1.3 Data Supplementation

The base data was supplemented by also extracting the title of the news article and the tags available on most articles, which listed the states and territories discussed in the article. The title was acquired in order to obtain as much context about the news article as possible for further analysis. Moreover, the tags were acquired to act as the true labels for validating the Named Entity Recognition (NER) and geocoding models in the NLP pipeline. While these place tags did exist, it was discovered that a significant number of these tags were missing from various property news articles as detailed further in the Exploratory Data Analysis (EDA) section of this Report. This finding further underlined the necessity to incorporate the NER and geocoding models into the proposed NLP pipeline.

1.4 Ethics of Data Extraction

Consent for web crawling the data on this Australian news source was confirmed by consulting with the website's copyright protection notice and the robots.txt file found at the root of the website. While this website did not openly acknowledge previously allowed web scraping, the only disallow directive present in the robots.txt file was for any data under

https://www.onthehouse.com.au/real-estate-agents, however, all relevant data for this Project was contained under https://www.onthehouse.com.au/news instead. Furthermore, all of the relevant data was not behind any kind of walled garden, pay or otherwise, which specified that it was publicly available data.

2 Description of Web Crawler Workflow

A class was used to house all of the functions that comprised the web crawler application. This decision to utilise object-oriented programming (OOP) was made as it increased scalability, maintainability and reusuability of the code (Python Software Foundation, 2021). The web crawler class comprised of the following distinct components:

- **Load necessary Python libraries**: the *selenium* package was loaded in order to automate web browser interactions from Python. Examples of required web browser interactions included launching a browser, extracting text from HTML elements found across different websites, clicking through different pages of the website and closing the browser. Additionally, the *pandas* package was loaded to enable the storage of extracted data as DataFrame objects for data manipulation further downstream. Moreover, the *datetime* package was loaded in order to save the web crawler extraction date in the extracted data outputs for tracking purposes.

- **Launch browser and open website**: the *ChromeDriver* was executed during the initialisation of the web crawler class to launch the Google Chrome web browser. It was then possible to execute a *Selenium* script to open the *onthehouse* website from there. The *ChromeDriver* was chosen over the *FirefoxDriver* as Google Chrome accounted for approximately 65% of the browser market share, and so it was more likely that most of the visitors to the *onthehouse* website used Google Chrome (Statcounter, 2021).

- **Obtain and store relevant URL addresses**: before the text corpus and other relevant data could be extracted, a list of URL addresses leading to these property news articles were first extracted from the title listings on the home pages using a *Selenium* script. Obtaining these URL addresses made it easier to navigate to the relevant property news articles on the website. The *get_URLs()* function in the class performed this task. This list of URL addresses was then stored as a CSV file to provide transparency on where the web crawler would extract data from later on. Further details regarding the web navigation process and the data extraction process involved can be found in the subsequent section.

- **Extract and store relevant news data**: looping through each of the URL addresses collected earlier, each website was individually opened and from this point, the title, date, body text, and tags were then extracted using a *Selenium* script under the *get_Content()* function in the class. All of the extracted data was then stored as a CSV file so that the NLP pipeline could be applied to the data without having to re-run the web crawler application.

- **Quite driver**: once the relevant data was collected and stored successfully, the browser was closed and the *ChromeDriver* session was terminated.

3 Data Extraction, Collection Method and Description of Corpus

3.1 Data Extraction and Collection Method

After the initialisation of the web crawler class, the *run* method was the first to be executed. This method linked the *get_URLs()* method with the *get_Content()* method by feeding the output list of URL addresses as an input for extracting the base and supplementary data.

One of the functionalities of the *get_URLs()* method was to control the navigation process through the *onthehouse* website. This was achieved by exploiting the HTML element for the *next page* button using the XPath tool. XPath was chosen for this task as it allowed for easy, readable and direct access to the relevant HTML element without having to go through the entire HTML tree (Datafiniti, 2014; Sahin, 2019).

At the time of this study, the chosen website contained 136 pages of property news articles to navigate through, and the data from all of these website addresses were needed as it was deemed important to afford more power to the NLP models. This max page count was also acquired using XPath, however, additional transformations were needed to attain the desired format. First, the inner HTML was converted to text which returned "Page 1 of 136". Subsequently, the last page was stripped from this string and converted into an integer for use within a while loop using j as the iterator variable. Here, the web crawler would be programmed to jump to the next page repeatedly until this max page was reached. However, a responsible time delay of two seconds was added in between page jumps to prevent any violations to the terms of service of the website (Heydt, 2018). This decision to retrieve a dynamically changing max page count was deemed important in future-proofing the web crawler.

While on each page, the *get_URLs()* method also utilised XPath to retrieve the property news block URL addresses. It was observed that there were always five property news blocks on each page. Consequently, the path expression to select the relevant nodes in the XML document incorporated iteratively changing numbers from one to five to select all of these property news blocks on the page. This was performed within a while loop that used i as the iterator variable. Again, a responsible time delay of two seconds was added in between extractions of each property block URL address. Once all of the URL addresses were collected, the *get_URL()* method concluded by saving the output as a CSV file and returning the URL addresses as a list.

The *run* method then iteratively fed each URL in this list of URL addresses into the *get_Content()* method using a for loop. This for loop used k as the iterator variable. The title, date, body and tag data were then extracted using XPath, and if any information was missing, an 'N/A' value would be returned instead. This data was then appended to an empty DataFrame, row by row. The data extraction date was also appended at this point for tracking purposes. Once again, a responsible time delay of two seconds was added in between extracting the title, date, body and tag data from the news article website.

Prior to invoking the *quite_driver()* method, the *run* method also saved the extracted data as a CSV file and returned the DataFrame.

3.2 Description of Corpus

The final output from the web crawler was a DataFrame containing the following columns:

- **Body**: the body text of the news article as a string.
- **Data_Date**: the date that the data was extracted by the web crawler as a date object.
- **Date**: the date the news article was published as a string.
- **Tags**: the place tags listed on the news article as a string.
- **Title**: the title of the news article as a string.

The screenshot below displays the DataFrame object returned from the web crawler.

	Body	Data_Date	Date	Tags	Title
0	Melbourne dwelling values have surpassed their...	2021_04_23	29-Mar-21	National VIC	Melbourne property values regain COVID loss an...
1	This edition of the Pain and Gain report analy...	2021_04_23	25-Mar-21	National NSW VIC QLD NT TAS ACT WA SA	Profitability in Australian dwellings rose ove...
2	New data suggests changes to mortgage lending ...	2021_04_23	22-Mar-21	National NSW SA VIC QLD TAS WA ACT NT	Housing lending may be cheap, but regulators a...
3	CoreLogic today announced Sydney property valu...	2021_04_23	15-Mar-21	National NSW	Sydney property values reach new record high
4	Australian home values surged 2.1% higher in F...	2021_04_23	1-Mar-21	National NSW QLD VIC NT SA WA ACT TAS	Momentum builds across Australian housing mark...

Figure 1: Corpus Description

3.3 Data Transformation

Preliminary data preprocessing was conducted on the web crawler output so that it could be ingested by the NLP pipeline downstream. This involved converting any missing values or 'N/A' strings to nan values, converting the date strings into datetime, replacing unwanted characters (i.e. $\hat{a}\epsilon^{TM}$), removing excess white spaces and transforming the tag strings into a list of tags per news article.

	Data_Date	Title	Date_Transformed	Body_Transformed	Tags_Transformed
0	2021_04_23	Melbourne property values regain COVID loss an...	2021-03-29	Melbourne dwelling values have surpassed their...	(National, VIC)
1	2021_04_23	Profitability in Australian dwellings rose ove...	2021-03-25	This edition of the Pain and Gain report analy...	(National, NSW, VIC, QLD, NT, TAS, ACT, WA, SA)
2	2021_04_23	Housing lending may be cheap, but regulators a...	2021-03-22	New data suggests changes to mortgage lending ...	(National, NSW, SA, VIC, QLD, TAS, WA, ACT, NT)
3	2021_04_23	Sydney property values reach new record high	2021-03-15	CoreLogic today announced Sydney property valu...	(National, NSW)
4	2021_04_23	Momentum builds across Australian housing mark...	2021-03-01	Australian home values surged 2.1% higher in F...	(National, NSW, QLD, VIC, NT, SA, WA, ACT, TAS)

Figure 2: Transformed Data

Subsequently, this DataFrame was exploded using the tags column so that the relevant sentences discussing distinct place tags could be analysed during the NLP pipeline.

3.4 Exploratory Data Analysis

During the EDA, it was identified that there were 1,287 rows of data in the output DataFrame. Additionally, it was found that the data extracted was dated from 2 May 2017 to 29 March 2021. Furthermore, there were no nan values within the date, body and title data, but there were 541 missing values in the tags data. The proportion of nan values against all non-nan values in the tag data is displayed in the following figure. This finding emphasisd the necessity for the inclusion of NER in the NLP pipeline to locate the place tags in the news articles.

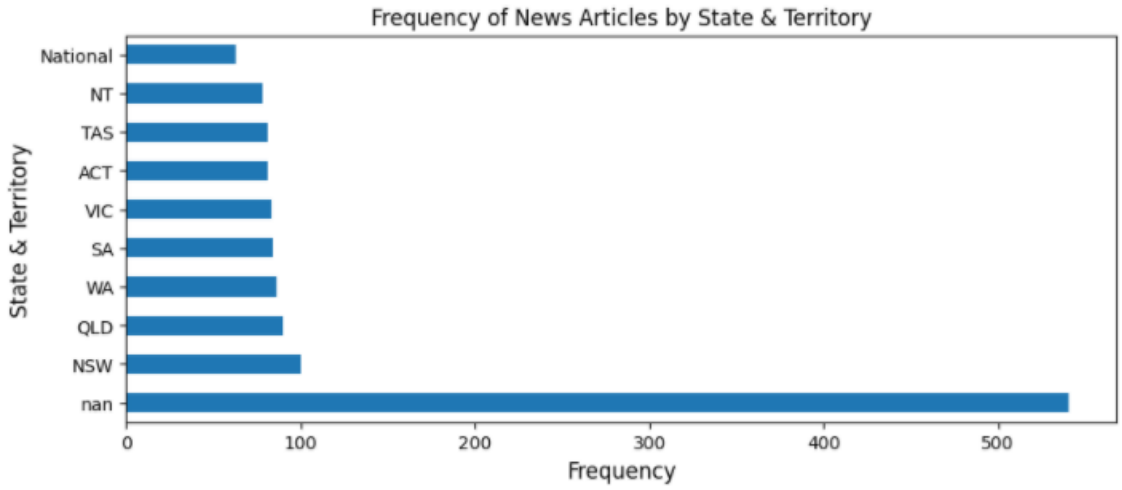


Figure 3: Exploratory Data Analysis

4 Web Crawler Code with Screenshots

4.1 Relevant Code

4.1.1 Web Crawler

[6]:
```
# import packages
import warnings # suppress warnings
warnings.filterwarnings('ignore') # suppress warnings
from selenium import webdriver # for opening webdriver
import time # for implementing time delays
import pandas as pd # for creating dataframes
pd.options.mode.chained_assignment = None  # to suppress SettingWithCopyWarning
from selenium.common.exceptions import NoSuchElementException # for raising␣
 ↪exceptions
from datetime import datetime # for getting current date

# configure pandas
pd.set_option('display.max_columns', None)
pd.set_option('display.max_rows', None)

# define property news web crawler
class Property_News_Web_Crawler:
    def __init__(self):
        # open driver
        self.driver = webdriver.Chrome(r"C:
 ↪\Users\Imran\Documents\chromedriver\chromedriver.exe")
        self.file_path = r'C:/Users/Imran/Desktop/Assignment_3_Repo/'

    def run(self):
        data = pd.DataFrame()
        for idx, url in enumerate(self.get_URLs()):
            data = data.append(self.get_Content(idx, url), ignore_index = True)
        self.quite_driver()
        data.to_csv(self.file_path + 'web_crawled_news_data.csv', index = False)
        return data

    def quite_driver(self):
        self.driver.quit()

    def get_URLs(self):
        # go to website
        self.driver.get('https://www.onthehouse.com.au/news')

        # extract page numbering of website
        element = self.driver.find_element_by_xpath('//
 ↪*[@id="block-views-blockarticles-property-news-block-1"]/div/div/nav/li')
        page_data = element.get_attribute('innerHTML').replace("\n", "")
```

```
        # locate max page limit from page numbering and remove trailing and␣
↪leading white spaces from it
        max_page_limit = int(page_data[page_data.find("of")+2:].split()[0].
↪strip())

        # empty list for urls
        url_list = []

        # seed page number
        i = 1
        # define last page for web crawling urls
        last_page = max_page_limit
        # loop through pages
        while i < last_page:
            # seed news article number on page
            j = 1
            while j <= 5:
                print("i:", i,", j:", j)
                # extract url for news article
                URL = self.driver.find_element_by_xpath('//
↪*[@id="block-views-blockarticles-property-news-block-1"]/div/div/div['+\
                                                    str(j)+']/a').
↪get_attribute("href")
                # take a brief pause so that we remain a nice little web␣
↪crawler for the website
                time.sleep(2)
                # append extracted url to list
                url_list.append(URL)
                # cycle to next news article on the page
                j = j + 1
            # take a brief pause so that we remain a nice little web crawler␣
↪for the website
            time.sleep(2)
            i = i + 1
            # define button for going to next page once all news article urls␣
↪have been scraped on this page
            next_page_button = self.driver.find_element_by_xpath("//
↪a[@rel='next']")

            # click button to next page
            next_page_button.click()

        # convert web crawled url list to dataframe
        url_df = pd.DataFrame(list(zip(url_list)), columns = ['URL'])
```

```
        # save urls to csv (serves as a checkpoint)
        url_df.to_csv(self.file_path + 'url_list.csv', index=False)

        # read in csv saved previously
        return url_list

    # define function for web crawling property news URL
    def get_Content(self, idx, url):
        # go to website
        self.driver.get(str(url))

        # get news article title
        try:
            title = self.driver.find_element_by_xpath('//
↪*[@id="block-corelogic-content"]/article/div[1]/div/h1/span').text
        # if title is missing, put N/A
        except NoSuchElementException:
            title = 'N/A'

        # take a brief pause so that we remain a nice little web crawler for␣
↪the website
        time.sleep(2)

        # get date of news article
        try:
            date = self.driver.find_element_by_xpath("//
↪p[@class='paragraph-active news-date']").text
        # if date is missing, put N/A
        except NoSuchElementException:
            date = 'N/A'
        # take a brief pause so that we remain a nice little web crawler for␣
↪the website
        time.sleep(2)

        # get body text of news article
        try:
            body = self.driver.find_element_by_xpath("//div[@property='schema:
↪text']").get_attribute('innerText')
        # if body is missing, put N/A
        except NoSuchElementException:
            body = 'N/A'

        # take a brief pause so that we remain a nice little web crawler for␣
↪the website
        time.sleep(2)
```

```
        # get labelled location tags for article
        try:
            tags = self.driver.find_element_by_xpath("//
↪div[@class='field_article_state']").text
        # if tag is missing, put N/A
        except NoSuchElementException:
            tags = 'N/A'

        # show progress of web crawler get_Content
        print("k:", idx+1)

        # get date of when data was web crawled
        data_date = datetime.now().strftime("%d-%m-%Y")

        # return values in dictionary for appending into a dataframe
        return {'Title': title, 'Date' : date, 'Body': body, 'Tags' : tags,␣
↪'Data_Date' : data_date}
```

[7]:
```
data = Property_News_Web_Crawler().run()
```

```
i: 1 , j: 1
i: 1 , j: 2
i: 1 , j: 3
i: 1 , j: 4
i: 1 , j: 5
i: 2 , j: 1
i: 2 , j: 2
i: 2 , j: 3
i: 2 , j: 4
i: 2 , j: 5
i: 3 , j: 1
i: 3 , j: 2
i: 3 , j: 3
i: 3 , j: 4
i: 3 , j: 5
i: 4 , j: 1
i: 4 , j: 2
i: 4 , j: 3
i: 4 , j: 4
i: 4 , j: 5
i: 5 , j: 1
i: 5 , j: 2
i: 5 , j: 3
i: 5 , j: 4
i: 5 , j: 5
i: 6 , j: 1
i: 6 , j: 2
i: 6 , j: 3
```

```
i: 6 , j: 4
i: 6 , j: 5
i: 7 , j: 1
i: 7 , j: 2
i: 7 , j: 3
i: 7 , j: 4
i: 7 , j: 5
i: 8 , j: 1
i: 8 , j: 2
i: 8 , j: 3
i: 8 , j: 4
i: 8 , j: 5
i: 9 , j: 1
i: 9 , j: 2
i: 9 , j: 3
i: 9 , j: 4
i: 9 , j: 5
i: 10 , j: 1
i: 10 , j: 2
i: 10 , j: 3
i: 10 , j: 4
i: 10 , j: 5
i: 11 , j: 1
i: 11 , j: 2
i: 11 , j: 3
i: 11 , j: 4
i: 11 , j: 5
i: 12 , j: 1
i: 12 , j: 2
i: 12 , j: 3
i: 12 , j: 4
i: 12 , j: 5
i: 13 , j: 1
i: 13 , j: 2
i: 13 , j: 3
i: 13 , j: 4
i: 13 , j: 5
i: 14 , j: 1
i: 14 , j: 2
i: 14 , j: 3
i: 14 , j: 4
i: 14 , j: 5
i: 15 , j: 1
i: 15 , j: 2
i: 15 , j: 3
i: 15 , j: 4
i: 15 , j: 5
i: 16 , j: 1
```

```
i: 16 , j: 2
i: 16 , j: 3
i: 16 , j: 4
i: 16 , j: 5
i: 17 , j: 1
i: 17 , j: 2
i: 17 , j: 3
i: 17 , j: 4
i: 17 , j: 5
i: 18 , j: 1
i: 18 , j: 2
i: 18 , j: 3
i: 18 , j: 4
i: 18 , j: 5
i: 19 , j: 1
i: 19 , j: 2
i: 19 , j: 3
i: 19 , j: 4
i: 19 , j: 5
i: 20 , j: 1
i: 20 , j: 2
i: 20 , j: 3
i: 20 , j: 4
i: 20 , j: 5
i: 21 , j: 1
i: 21 , j: 2
i: 21 , j: 3
i: 21 , j: 4
i: 21 , j: 5
i: 22 , j: 1
i: 22 , j: 2
i: 22 , j: 3
i: 22 , j: 4
i: 22 , j: 5
i: 23 , j: 1
i: 23 , j: 2
i: 23 , j: 3
i: 23 , j: 4
i: 23 , j: 5
i: 24 , j: 1
i: 24 , j: 2
i: 24 , j: 3
i: 24 , j: 4
i: 24 , j: 5
i: 25 , j: 1
i: 25 , j: 2
i: 25 , j: 3
i: 25 , j: 4
```

```
i: 25 , j: 5
i: 26 , j: 1
i: 26 , j: 2
i: 26 , j: 3
i: 26 , j: 4
i: 26 , j: 5
i: 27 , j: 1
i: 27 , j: 2
i: 27 , j: 3
i: 27 , j: 4
i: 27 , j: 5
i: 28 , j: 1
i: 28 , j: 2
i: 28 , j: 3
i: 28 , j: 4
i: 28 , j: 5
i: 29 , j: 1
i: 29 , j: 2
i: 29 , j: 3
i: 29 , j: 4
i: 29 , j: 5
i: 30 , j: 1
i: 30 , j: 2
i: 30 , j: 3
i: 30 , j: 4
i: 30 , j: 5
i: 31 , j: 1
i: 31 , j: 2
i: 31 , j: 3
i: 31 , j: 4
i: 31 , j: 5
i: 32 , j: 1
i: 32 , j: 2
i: 32 , j: 3
i: 32 , j: 4
i: 32 , j: 5
i: 33 , j: 1
i: 33 , j: 2
i: 33 , j: 3
i: 33 , j: 4
i: 33 , j: 5
i: 34 , j: 1
i: 34 , j: 2
i: 34 , j: 3
i: 34 , j: 4
i: 34 , j: 5
i: 35 , j: 1
i: 35 , j: 2
```

```
i: 35 , j: 3
i: 35 , j: 4
i: 35 , j: 5
i: 36 , j: 1
i: 36 , j: 2
i: 36 , j: 3
i: 36 , j: 4
i: 36 , j: 5
i: 37 , j: 1
i: 37 , j: 2
i: 37 , j: 3
i: 37 , j: 4
i: 37 , j: 5
i: 38 , j: 1
i: 38 , j: 2
i: 38 , j: 3
i: 38 , j: 4
i: 38 , j: 5
i: 39 , j: 1
i: 39 , j: 2
i: 39 , j: 3
i: 39 , j: 4
i: 39 , j: 5
i: 40 , j: 1
i: 40 , j: 2
i: 40 , j: 3
i: 40 , j: 4
i: 40 , j: 5
i: 41 , j: 1
i: 41 , j: 2
i: 41 , j: 3
i: 41 , j: 4
i: 41 , j: 5
i: 42 , j: 1
i: 42 , j: 2
i: 42 , j: 3
i: 42 , j: 4
i: 42 , j: 5
i: 43 , j: 1
i: 43 , j: 2
i: 43 , j: 3
i: 43 , j: 4
i: 43 , j: 5
i: 44 , j: 1
i: 44 , j: 2
i: 44 , j: 3
i: 44 , j: 4
i: 44 , j: 5
```

```
i: 45 , j: 1
i: 45 , j: 2
i: 45 , j: 3
i: 45 , j: 4
i: 45 , j: 5
i: 46 , j: 1
i: 46 , j: 2
i: 46 , j: 3
i: 46 , j: 4
i: 46 , j: 5
i: 47 , j: 1
i: 47 , j: 2
i: 47 , j: 3
i: 47 , j: 4
i: 47 , j: 5
i: 48 , j: 1
i: 48 , j: 2
i: 48 , j: 3
i: 48 , j: 4
i: 48 , j: 5
i: 49 , j: 1
i: 49 , j: 2
i: 49 , j: 3
i: 49 , j: 4
i: 49 , j: 5
i: 50 , j: 1
i: 50 , j: 2
i: 50 , j: 3
i: 50 , j: 4
i: 50 , j: 5
i: 51 , j: 1
i: 51 , j: 2
i: 51 , j: 3
i: 51 , j: 4
i: 51 , j: 5
i: 52 , j: 1
i: 52 , j: 2
i: 52 , j: 3
i: 52 , j: 4
i: 52 , j: 5
i: 53 , j: 1
i: 53 , j: 2
i: 53 , j: 3
i: 53 , j: 4
i: 53 , j: 5
i: 54 , j: 1
i: 54 , j: 2
i: 54 , j: 3
```

```
i: 54 , j: 4
i: 54 , j: 5
i: 55 , j: 1
i: 55 , j: 2
i: 55 , j: 3
i: 55 , j: 4
i: 55 , j: 5
i: 56 , j: 1
i: 56 , j: 2
i: 56 , j: 3
i: 56 , j: 4
i: 56 , j: 5
i: 57 , j: 1
i: 57 , j: 2
i: 57 , j: 3
i: 57 , j: 4
i: 57 , j: 5
i: 58 , j: 1
i: 58 , j: 2
i: 58 , j: 3
i: 58 , j: 4
i: 58 , j: 5
i: 59 , j: 1
i: 59 , j: 2
i: 59 , j: 3
i: 59 , j: 4
i: 59 , j: 5
i: 60 , j: 1
i: 60 , j: 2
i: 60 , j: 3
i: 60 , j: 4
i: 60 , j: 5
i: 61 , j: 1
i: 61 , j: 2
i: 61 , j: 3
i: 61 , j: 4
i: 61 , j: 5
i: 62 , j: 1
i: 62 , j: 2
i: 62 , j: 3
i: 62 , j: 4
i: 62 , j: 5
i: 63 , j: 1
i: 63 , j: 2
i: 63 , j: 3
i: 63 , j: 4
i: 63 , j: 5
i: 64 , j: 1
```

```
i: 64 , j: 2
i: 64 , j: 3
i: 64 , j: 4
i: 64 , j: 5
i: 65 , j: 1
i: 65 , j: 2
i: 65 , j: 3
i: 65 , j: 4
i: 65 , j: 5
i: 66 , j: 1
i: 66 , j: 2
i: 66 , j: 3
i: 66 , j: 4
i: 66 , j: 5
i: 67 , j: 1
i: 67 , j: 2
i: 67 , j: 3
i: 67 , j: 4
i: 67 , j: 5
i: 68 , j: 1
i: 68 , j: 2
i: 68 , j: 3
i: 68 , j: 4
i: 68 , j: 5
i: 69 , j: 1
i: 69 , j: 2
i: 69 , j: 3
i: 69 , j: 4
i: 69 , j: 5
i: 70 , j: 1
i: 70 , j: 2
i: 70 , j: 3
i: 70 , j: 4
i: 70 , j: 5
i: 71 , j: 1
i: 71 , j: 2
i: 71 , j: 3
i: 71 , j: 4
i: 71 , j: 5
i: 72 , j: 1
i: 72 , j: 2
i: 72 , j: 3
i: 72 , j: 4
i: 72 , j: 5
i: 73 , j: 1
i: 73 , j: 2
i: 73 , j: 3
i: 73 , j: 4
```

```
i: 73 , j: 5
i: 74 , j: 1
i: 74 , j: 2
i: 74 , j: 3
i: 74 , j: 4
i: 74 , j: 5
i: 75 , j: 1
i: 75 , j: 2
i: 75 , j: 3
i: 75 , j: 4
i: 75 , j: 5
i: 76 , j: 1
i: 76 , j: 2
i: 76 , j: 3
i: 76 , j: 4
i: 76 , j: 5
i: 77 , j: 1
i: 77 , j: 2
i: 77 , j: 3
i: 77 , j: 4
i: 77 , j: 5
i: 78 , j: 1
i: 78 , j: 2
i: 78 , j: 3
i: 78 , j: 4
i: 78 , j: 5
i: 79 , j: 1
i: 79 , j: 2
i: 79 , j: 3
i: 79 , j: 4
i: 79 , j: 5
i: 80 , j: 1
i: 80 , j: 2
i: 80 , j: 3
i: 80 , j: 4
i: 80 , j: 5
i: 81 , j: 1
i: 81 , j: 2
i: 81 , j: 3
i: 81 , j: 4
i: 81 , j: 5
i: 82 , j: 1
i: 82 , j: 2
i: 82 , j: 3
i: 82 , j: 4
i: 82 , j: 5
i: 83 , j: 1
i: 83 , j: 2
```

```
i: 83 , j: 3
i: 83 , j: 4
i: 83 , j: 5
i: 84 , j: 1
i: 84 , j: 2
i: 84 , j: 3
i: 84 , j: 4
i: 84 , j: 5
i: 85 , j: 1
i: 85 , j: 2
i: 85 , j: 3
i: 85 , j: 4
i: 85 , j: 5
i: 86 , j: 1
i: 86 , j: 2
i: 86 , j: 3
i: 86 , j: 4
i: 86 , j: 5
i: 87 , j: 1
i: 87 , j: 2
i: 87 , j: 3
i: 87 , j: 4
i: 87 , j: 5
i: 88 , j: 1
i: 88 , j: 2
i: 88 , j: 3
i: 88 , j: 4
i: 88 , j: 5
i: 89 , j: 1
i: 89 , j: 2
i: 89 , j: 3
i: 89 , j: 4
i: 89 , j: 5
i: 90 , j: 1
i: 90 , j: 2
i: 90 , j: 3
i: 90 , j: 4
i: 90 , j: 5
i: 91 , j: 1
i: 91 , j: 2
i: 91 , j: 3
i: 91 , j: 4
i: 91 , j: 5
i: 92 , j: 1
i: 92 , j: 2
i: 92 , j: 3
i: 92 , j: 4
i: 92 , j: 5
```

```
i: 93 , j: 1
i: 93 , j: 2
i: 93 , j: 3
i: 93 , j: 4
i: 93 , j: 5
i: 94 , j: 1
i: 94 , j: 2
i: 94 , j: 3
i: 94 , j: 4
i: 94 , j: 5
i: 95 , j: 1
i: 95 , j: 2
i: 95 , j: 3
i: 95 , j: 4
i: 95 , j: 5
i: 96 , j: 1
i: 96 , j: 2
i: 96 , j: 3
i: 96 , j: 4
i: 96 , j: 5
i: 97 , j: 1
i: 97 , j: 2
i: 97 , j: 3
i: 97 , j: 4
i: 97 , j: 5
i: 98 , j: 1
i: 98 , j: 2
i: 98 , j: 3
i: 98 , j: 4
i: 98 , j: 5
i: 99 , j: 1
i: 99 , j: 2
i: 99 , j: 3
i: 99 , j: 4
i: 99 , j: 5
i: 100 , j: 1
i: 100 , j: 2
i: 100 , j: 3
i: 100 , j: 4
i: 100 , j: 5
i: 101 , j: 1
i: 101 , j: 2
i: 101 , j: 3
i: 101 , j: 4
i: 101 , j: 5
i: 102 , j: 1
i: 102 , j: 2
i: 102 , j: 3
```

```
i: 102 , j: 4
i: 102 , j: 5
i: 103 , j: 1
i: 103 , j: 2
i: 103 , j: 3
i: 103 , j: 4
i: 103 , j: 5
i: 104 , j: 1
i: 104 , j: 2
i: 104 , j: 3
i: 104 , j: 4
i: 104 , j: 5
i: 105 , j: 1
i: 105 , j: 2
i: 105 , j: 3
i: 105 , j: 4
i: 105 , j: 5
i: 106 , j: 1
i: 106 , j: 2
i: 106 , j: 3
i: 106 , j: 4
i: 106 , j: 5
i: 107 , j: 1
i: 107 , j: 2
i: 107 , j: 3
i: 107 , j: 4
i: 107 , j: 5
i: 108 , j: 1
i: 108 , j: 2
i: 108 , j: 3
i: 108 , j: 4
i: 108 , j: 5
i: 109 , j: 1
i: 109 , j: 2
i: 109 , j: 3
i: 109 , j: 4
i: 109 , j: 5
i: 110 , j: 1
i: 110 , j: 2
i: 110 , j: 3
i: 110 , j: 4
i: 110 , j: 5
i: 111 , j: 1
i: 111 , j: 2
i: 111 , j: 3
i: 111 , j: 4
i: 111 , j: 5
i: 112 , j: 1
```

```
i: 112 , j: 2
i: 112 , j: 3
i: 112 , j: 4
i: 112 , j: 5
i: 113 , j: 1
i: 113 , j: 2
i: 113 , j: 3
i: 113 , j: 4
i: 113 , j: 5
i: 114 , j: 1
i: 114 , j: 2
i: 114 , j: 3
i: 114 , j: 4
i: 114 , j: 5
i: 115 , j: 1
i: 115 , j: 2
i: 115 , j: 3
i: 115 , j: 4
i: 115 , j: 5
i: 116 , j: 1
i: 116 , j: 2
i: 116 , j: 3
i: 116 , j: 4
i: 116 , j: 5
i: 117 , j: 1
i: 117 , j: 2
i: 117 , j: 3
i: 117 , j: 4
i: 117 , j: 5
i: 118 , j: 1
i: 118 , j: 2
i: 118 , j: 3
i: 118 , j: 4
i: 118 , j: 5
i: 119 , j: 1
i: 119 , j: 2
i: 119 , j: 3
i: 119 , j: 4
i: 119 , j: 5
i: 120 , j: 1
i: 120 , j: 2
i: 120 , j: 3
i: 120 , j: 4
i: 120 , j: 5
i: 121 , j: 1
i: 121 , j: 2
i: 121 , j: 3
i: 121 , j: 4
```

```
i: 121 , j: 5
i: 122 , j: 1
i: 122 , j: 2
i: 122 , j: 3
i: 122 , j: 4
i: 122 , j: 5
i: 123 , j: 1
i: 123 , j: 2
i: 123 , j: 3
i: 123 , j: 4
i: 123 , j: 5
i: 124 , j: 1
i: 124 , j: 2
i: 124 , j: 3
i: 124 , j: 4
i: 124 , j: 5
i: 125 , j: 1
i: 125 , j: 2
i: 125 , j: 3
i: 125 , j: 4
i: 125 , j: 5
i: 126 , j: 1
i: 126 , j: 2
i: 126 , j: 3
i: 126 , j: 4
i: 126 , j: 5
i: 127 , j: 1
i: 127 , j: 2
i: 127 , j: 3
i: 127 , j: 4
i: 127 , j: 5
i: 128 , j: 1
i: 128 , j: 2
i: 128 , j: 3
i: 128 , j: 4
i: 128 , j: 5
i: 129 , j: 1
i: 129 , j: 2
i: 129 , j: 3
i: 129 , j: 4
i: 129 , j: 5
i: 130 , j: 1
i: 130 , j: 2
i: 130 , j: 3
i: 130 , j: 4
i: 130 , j: 5
i: 131 , j: 1
i: 131 , j: 2
```

```
i: 131 , j: 3
i: 131 , j: 4
i: 131 , j: 5
i: 132 , j: 1
i: 132 , j: 2
i: 132 , j: 3
i: 132 , j: 4
i: 132 , j: 5
i: 133 , j: 1
i: 133 , j: 2
i: 133 , j: 3
i: 133 , j: 4
i: 133 , j: 5
i: 134 , j: 1
i: 134 , j: 2
i: 134 , j: 3
i: 134 , j: 4
i: 134 , j: 5
i: 135 , j: 1
i: 135 , j: 2
i: 135 , j: 3
i: 135 , j: 4
i: 135 , j: 5
k: 1
k: 2
k: 3
k: 4
k: 5
k: 6
k: 7
k: 8
k: 9
k: 10
k: 11
k: 12
k: 13
k: 14
k: 15
k: 16
k: 17
k: 18
k: 19
k: 20
k: 21
k: 22
k: 23
k: 24
k: 25
```

```
k: 26
k: 27
k: 28
k: 29
k: 30
k: 31
k: 32
k: 33
k: 34
k: 35
k: 36
k: 37
k: 38
k: 39
k: 40
k: 41
k: 42
k: 43
k: 44
k: 45
k: 46
k: 47
k: 48
k: 49
k: 50
k: 51
k: 52
k: 53
k: 54
k: 55
k: 56
k: 57
k: 58
k: 59
k: 60
k: 61
k: 62
k: 63
k: 64
k: 65
k: 66
k: 67
k: 68
k: 69
k: 70
k: 71
k: 72
k: 73
```

```
k: 74
k: 75
k: 76
k: 77
k: 78
k: 79
k: 80
k: 81
k: 82
k: 83
k: 84
k: 85
k: 86
k: 87
k: 88
k: 89
k: 90
k: 91
k: 92
k: 93
k: 94
k: 95
k: 96
k: 97
k: 98
k: 99
k: 100
k: 101
k: 102
k: 103
k: 104
k: 105
k: 106
k: 107
k: 108
k: 109
k: 110
k: 111
k: 112
k: 113
k: 114
k: 115
k: 116
k: 117
k: 118
k: 119
k: 120
k: 121
```

```
k: 122
k: 123
k: 124
k: 125
k: 126
k: 127
k: 128
k: 129
k: 130
k: 131
k: 132
k: 133
k: 134
k: 135
k: 136
k: 137
k: 138
k: 139
k: 140
k: 141
k: 142
k: 143
k: 144
k: 145
k: 146
k: 147
k: 148
k: 149
k: 150
k: 151
k: 152
k: 153
k: 154
k: 155
k: 156
k: 157
k: 158
k: 159
k: 160
k: 161
k: 162
k: 163
k: 164
k: 165
k: 166
k: 167
k: 168
k: 169
```

```
k: 170
k: 171
k: 172
k: 173
k: 174
k: 175
k: 176
k: 177
k: 178
k: 179
k: 180
k: 181
k: 182
k: 183
k: 184
k: 185
k: 186
k: 187
k: 188
k: 189
k: 190
k: 191
k: 192
k: 193
k: 194
k: 195
k: 196
k: 197
k: 198
k: 199
k: 200
k: 201
k: 202
k: 203
k: 204
k: 205
k: 206
k: 207
k: 208
k: 209
k: 210
k: 211
k: 212
k: 213
k: 214
k: 215
k: 216
k: 217
```

```
k: 218
k: 219
k: 220
k: 221
k: 222
k: 223
k: 224
k: 225
k: 226
k: 227
k: 228
k: 229
k: 230
k: 231
k: 232
k: 233
k: 234
k: 235
k: 236
k: 237
k: 238
k: 239
k: 240
k: 241
k: 242
k: 243
k: 244
k: 245
k: 246
k: 247
k: 248
k: 249
k: 250
k: 251
k: 252
k: 253
k: 254
k: 255
k: 256
k: 257
k: 258
k: 259
k: 260
k: 261
k: 262
k: 263
k: 264
k: 265
```

```
k: 266
k: 267
k: 268
k: 269
k: 270
k: 271
k: 272
k: 273
k: 274
k: 275
k: 276
k: 277
k: 278
k: 279
k: 280
k: 281
k: 282
k: 283
k: 284
k: 285
k: 286
k: 287
k: 288
k: 289
k: 290
k: 291
k: 292
k: 293
k: 294
k: 295
k: 296
k: 297
k: 298
k: 299
k: 300
k: 301
k: 302
k: 303
k: 304
k: 305
k: 306
k: 307
k: 308
k: 309
k: 310
k: 311
k: 312
k: 313
```

```
k: 314
k: 315
k: 316
k: 317
k: 318
k: 319
k: 320
k: 321
k: 322
k: 323
k: 324
k: 325
k: 326
k: 327
k: 328
k: 329
k: 330
k: 331
k: 332
k: 333
k: 334
k: 335
k: 336
k: 337
k: 338
k: 339
k: 340
k: 341
k: 342
k: 343
k: 344
k: 345
k: 346
k: 347
k: 348
k: 349
k: 350
k: 351
k: 352
k: 353
k: 354
k: 355
k: 356
k: 357
k: 358
k: 359
k: 360
k: 361
```

```
k: 362
k: 363
k: 364
k: 365
k: 366
k: 367
k: 368
k: 369
k: 370
k: 371
k: 372
k: 373
k: 374
k: 375
k: 376
k: 377
k: 378
k: 379
k: 380
k: 381
k: 382
k: 383
k: 384
k: 385
k: 386
k: 387
k: 388
k: 389
k: 390
k: 391
k: 392
k: 393
k: 394
k: 395
k: 396
k: 397
k: 398
k: 399
k: 400
k: 401
k: 402
k: 403
k: 404
k: 405
k: 406
k: 407
k: 408
k: 409
```

```
k: 410
k: 411
k: 412
k: 413
k: 414
k: 415
k: 416
k: 417
k: 418
k: 419
k: 420
k: 421
k: 422
k: 423
k: 424
k: 425
k: 426
k: 427
k: 428
k: 429
k: 430
k: 431
k: 432
k: 433
k: 434
k: 435
k: 436
k: 437
k: 438
k: 439
k: 440
k: 441
k: 442
k: 443
k: 444
k: 445
k: 446
k: 447
k: 448
k: 449
k: 450
k: 451
k: 452
k: 453
k: 454
k: 455
k: 456
k: 457
```

```
k: 458
k: 459
k: 460
k: 461
k: 462
k: 463
k: 464
k: 465
k: 466
k: 467
k: 468
k: 469
k: 470
k: 471
k: 472
k: 473
k: 474
k: 475
k: 476
k: 477
k: 478
k: 479
k: 480
k: 481
k: 482
k: 483
k: 484
k: 485
k: 486
k: 487
k: 488
k: 489
k: 490
k: 491
k: 492
k: 493
k: 494
k: 495
k: 496
k: 497
k: 498
k: 499
k: 500
k: 501
k: 502
k: 503
k: 504
k: 505
```

```
k: 506
k: 507
k: 508
k: 509
k: 510
k: 511
k: 512
k: 513
k: 514
k: 515
k: 516
k: 517
k: 518
k: 519
k: 520
k: 521
k: 522
k: 523
k: 524
k: 525
k: 526
k: 527
k: 528
k: 529
k: 530
k: 531
k: 532
k: 533
k: 534
k: 535
k: 536
k: 537
k: 538
k: 539
k: 540
k: 541
k: 542
k: 543
k: 544
k: 545
k: 546
k: 547
k: 548
k: 549
k: 550
k: 551
k: 552
k: 553
```

```
k: 554
k: 555
k: 556
k: 557
k: 558
k: 559
k: 560
k: 561
k: 562
k: 563
k: 564
k: 565
k: 566
k: 567
k: 568
k: 569
k: 570
k: 571
k: 572
k: 573
k: 574
k: 575
k: 576
k: 577
k: 578
k: 579
k: 580
k: 581
k: 582
k: 583
k: 584
k: 585
k: 586
k: 587
k: 588
k: 589
k: 590
k: 591
k: 592
k: 593
k: 594
k: 595
k: 596
k: 597
k: 598
k: 599
k: 600
k: 601
```

```
k: 602
k: 603
k: 604
k: 605
k: 606
k: 607
k: 608
k: 609
k: 610
k: 611
k: 612
k: 613
k: 614
k: 615
k: 616
k: 617
k: 618
k: 619
k: 620
k: 621
k: 622
k: 623
k: 624
k: 625
k: 626
k: 627
k: 628
k: 629
k: 630
k: 631
k: 632
k: 633
k: 634
k: 635
k: 636
k: 637
k: 638
k: 639
k: 640
k: 641
k: 642
k: 643
k: 644
k: 645
k: 646
k: 647
k: 648
k: 649
```

```
k: 650
k: 651
k: 652
k: 653
k: 654
k: 655
k: 656
k: 657
k: 658
k: 659
k: 660
k: 661
k: 662
k: 663
k: 664
k: 665
k: 666
k: 667
k: 668
k: 669
k: 670
k: 671
k: 672
k: 673
k: 674
k: 675
```

4.1.2 Preliminary Data Preprocessing

[11]:
```
# read in csv saved previously
data = pd.read_csv(r'C:/Users/Imran/Desktop/Assignment_3_Repo/
 ↪web_crawled_news_data.csv')
```

[12]:
```
data.head()
```

[12]:

	Body	Data_Date	Date	Tags
0	Melbourne dwelling values have surpassed their…	24-04-2021	29 Mar 2021	National VIC
1	This edition of the Pain and Gain report analy…	24-04-2021	25 Mar 2021	National NSW VIC QLD NT TAS ACT WA SA
2	New data suggests changes to mortgage lending …	24-04-2021	22 Mar 2021	National NSW SA VIC QLD TAS WA ACT NT
3	CoreLogic today announced Sydney property valu…	24-04-2021	15 Mar 2021	National NSW
4	Australian home values surged 2.1% higher in F…	24-04-2021	1 Mar 2021	

```
4  National NSW QLD VIC NT SA WA ACT TAS

                                                Title
0  Melbourne property values regain COVID loss an…
1  Profitability in Australian dwellings rose ove…
2  Housing lending may be cheap, but regulators a…
3       Sydney property values reach new record high
4  Momentum builds across Australian housing mark…
```

[14]:
```
# import packages
import ast # for converting string list representation to list
import numpy as np # for working with arrays

# translate missing values to nan
data = data.replace(('*/*', '', 'nan', '0', 'N/A'), np.nan)

# transform date
def convert_to_date(x):
    date_time_obj = datetime.strptime(x, '%d %b %Y')
    return date_time_obj
data['Date_Transformed'] = data['Date'].apply(convert_to_date)

# transform body tags
# replace unwanted strings
data['Body_Transformed'] = data['Body'].str.replace(r'\n',' ', regex=True)
data['Body_Transformed'] = data['Body_Transformed'].str.replace(r'\xa0',' ',␣
 ↪regex=True)
data['Body_Transformed'] = data['Body_Transformed'].str.replace(r'â€ ','',␣
 ↪regex=True)
data['Body_Transformed'] = data['Body_Transformed'].str.replace(r'â€˜','',␣
 ↪regex=True)
data['Body_Transformed'] = data['Body_Transformed'].str.replace(r'â€œ','',␣
 ↪regex=True)
data['Body_Transformed'] = data['Body_Transformed'].str.replace(r'â€"','',␣
 ↪regex=True)

# remove excess white space
data['Body_Transformed'] = data['Body_Transformed'].astype(str).apply(lambda x:␣
 ↪' '.join(x.split()))

# transform Tags
data['Tags_Transformed'] = data['Tags'].str.replace(r' ',"', '", regex=True)
data['Tags_Transformed'] = "'" + data['Tags_Transformed'].astype(str) + "'"
def literal_return(val):
    try:
        return ast.literal_eval(val)
    except (ValueError, SyntaxError) as e:
```

```
        return val
data['Tags_Transformed'] = data['Tags_Transformed'].apply(literal_return)
```

4.1.3 Exploratory Data Analysis

[15]:
```
# duplicate article for each place tag listed
data_overview = data
data_overview = data_overview.explode('Tags_Transformed').reset_index(drop =␣
 ↪True)

# check shape of data web crawled
nrow, ncol = data_overview.shape
nrow, ncol
```

[15]: (1287, 8)

[16]:
```
# check first couple of rows in dataframe
data_overview.head()
```

[16]:

	Body	Data_Date	Date	Tags	Title	Date_Transformed	Body_Transformed	Tags_Transformed
0	Melbourne dwelling values have surpassed their…	24-04-2021	29 Mar 2021	National VIC	Melbourne property values regain COVID loss an…	2021-03-29	Melbourne dwelling values have surpassed their…	National
1	Melbourne dwelling values have surpassed their…	24-04-2021	29 Mar 2021	National VIC	Melbourne property values regain COVID loss an…	2021-03-29	Melbourne dwelling values have surpassed their…	VIC
2	This edition of the Pain and Gain report analy…	24-04-2021	25 Mar 2021	National NSW VIC QLD NT TAS ACT WA SA	Profitability in Australian dwellings rose ove…	2021-03-25	This edition of the Pain and Gain report analy…	National
3	This edition of the Pain and Gain report analy…	24-04-2021	25 Mar 2021	National NSW VIC QLD NT TAS ACT WA SA	Profitability in Australian dwellings rose ove…	2021-03-25	This edition of the Pain and Gain report analy…	NSW
4	This edition of the Pain and Gain report analy…	24-04-2021	25 Mar 2021	National NSW VIC QLD NT TAS ACT WA SA	Profitability in Australian dwellings rose ove…	2021-03-25	This edition of the Pain and Gain report analy…	VIC

[17]:
```
# check first and last publication dates
print(min(data_overview['Date_Transformed']))
print(max(data_overview['Date_Transformed']))
```

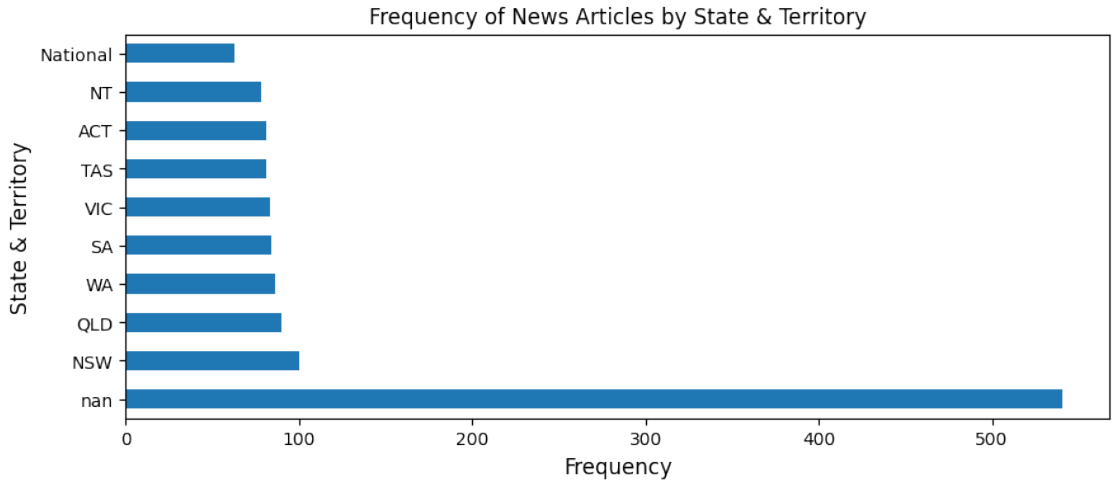
```
2017-05-02 00:00:00
2021-03-29 00:00:00
```

[18]:
```
# check for null values
print("Nulls in 'Date'", data_overview['Date'].isnull().sum())
print("Nulls in 'Body'", data_overview['Body'].isnull().sum())
print("Nulls in 'Title'", data_overview['Title'].isnull().sum())
print("Nulls in 'Tags'", data_overview['Tags'].isnull().sum())
```

```
Nulls in 'Date' 0
Nulls in 'Body' 0
Nulls in 'Title' 0
Nulls in 'Tags' 541
```

[19]:
```
#import packages
import matplotlib.pyplot as plt # for plotting
from matplotlib.pyplot import figure # for plotting

# specify fig size and dpi
figure(figsize=(10, 4), dpi=100)
data_overview['Tags_Transformed'].value_counts().plot(kind='barh')
plt.title('Frequency of News Articles by State & Territory', fontsize = 12)
plt.xlabel('Frequency', fontsize = 12)
plt.ylabel('State & Territory', fontsize = 12)
plt.show()
```



[20]:
```
# select relevant columns for further processing
data2 = data[['Data_Date', 'Title', 'Date_Transformed', 'Body_Transformed',␣
 ↪'Tags_Transformed']]

# save transformed data for use in the NLP pipeline
data.to_csv(r'C:/Users/Imran/Desktop/Assignment_3_Repo/data2.csv', index =␣
 ↪False)
```

[21]:
```
data2.head()
```

[21]:
```
    Data_Date                                              Title  \
0  24-04-2021  Melbourne property values regain COVID loss an…
1  24-04-2021  Profitability in Australian dwellings rose ove…
2  24-04-2021  Housing lending may be cheap, but regulators a…
3  24-04-2021       Sydney property values reach new record high
4  24-04-2021  Momentum builds across Australian housing mark…

  Date_Transformed                                   Body_Transformed  \
0       2021-03-29  Melbourne dwelling values have surpassed their…
1       2021-03-25  This edition of the Pain and Gain report analy…
2       2021-03-22  New data suggests changes to mortgage lending …
3       2021-03-15  CoreLogic today announced Sydney property valu…
4       2021-03-01  Australian home values surged 2.1% higher in F…

                                Tags_Transformed
0                                (National, VIC)
1  (National, NSW, VIC, QLD, NT, TAS, ACT, WA, SA)
2  (National, NSW, SA, VIC, QLD, TAS, WA, ACT, NT)
3                                (National, NSW)
4  (National, NSW, QLD, VIC, NT, SA, WA, ACT, TAS)
```

4.2 Demonstration of the Web Crawler Application

The following figure demonstrates the property news web crawler opening the *onthehouse* website on the property news page.

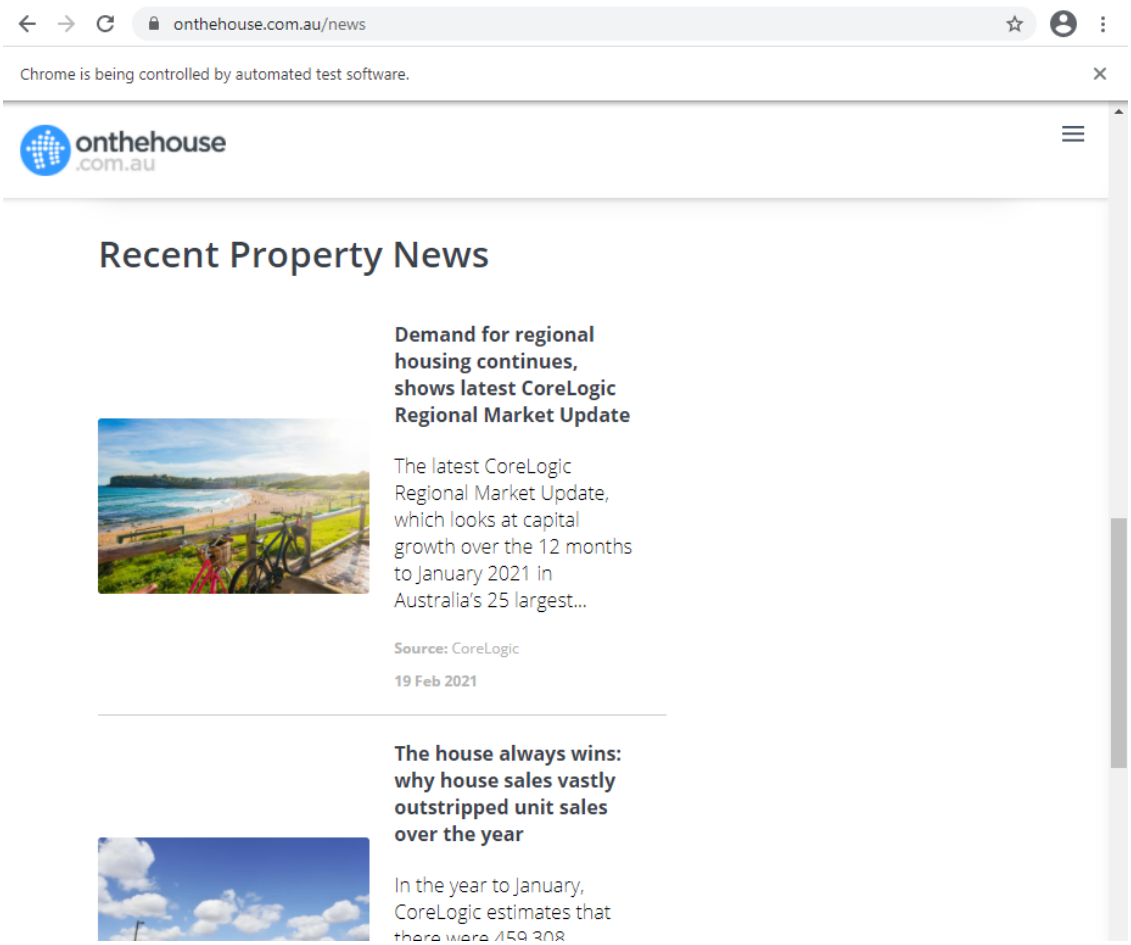
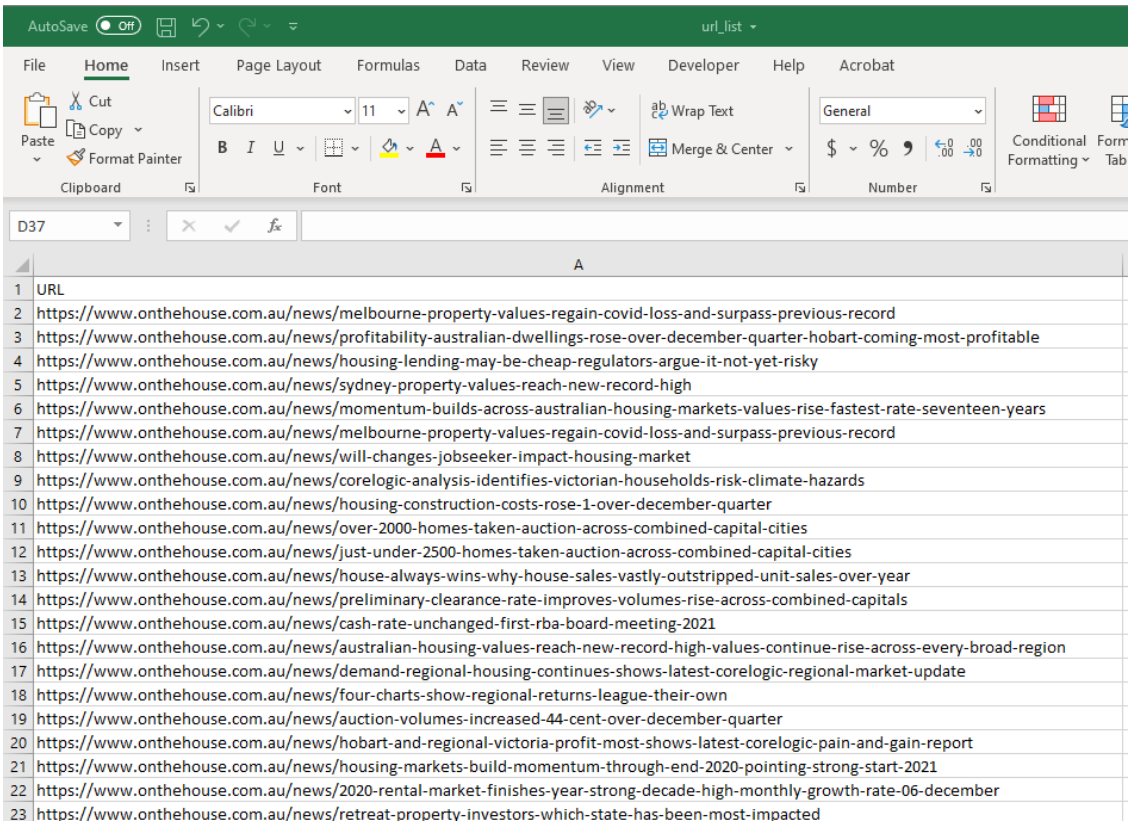


Figure 4: Web Crawler Starting Up

The following figure demonstrates the property news web crawler extracting the five URL addresses per page.

Figure 5: Web Crawler Getting URL Addresses

The following is a screenshot of the list of URL addresses stored as a CSV file.



	A
1	URL
2	https://www.onthehouse.com.au/news/melbourne-property-values-regain-covid-loss-and-surpass-previous-record
3	https://www.onthehouse.com.au/news/profitability-australian-dwellings-rose-over-december-quarter-hobart-coming-most-profitable
4	https://www.onthehouse.com.au/news/housing-lending-may-be-cheap-regulators-argue-it-not-yet-risky
5	https://www.onthehouse.com.au/news/sydney-property-values-reach-new-record-high
6	https://www.onthehouse.com.au/news/momentum-builds-across-australian-housing-markets-values-rise-fastest-rate-seventeen-years
7	https://www.onthehouse.com.au/news/melbourne-property-values-regain-covid-loss-and-surpass-previous-record
8	https://www.onthehouse.com.au/news/will-changes-jobseeker-impact-housing-market
9	https://www.onthehouse.com.au/news/corelogic-analysis-identifies-victorian-households-risk-climate-hazards
10	https://www.onthehouse.com.au/news/housing-construction-costs-rose-1-over-december-quarter
11	https://www.onthehouse.com.au/news/over-2000-homes-taken-auction-across-combined-capital-cities
12	https://www.onthehouse.com.au/news/just-under-2500-homes-taken-auction-across-combined-capital-cities
13	https://www.onthehouse.com.au/news/house-always-wins-why-house-sales-vastly-outstripped-unit-sales-over-year
14	https://www.onthehouse.com.au/news/preliminary-clearance-rate-improves-volumes-rise-across-combined-capitals
15	https://www.onthehouse.com.au/news/cash-rate-unchanged-first-rba-board-meeting-2021
16	https://www.onthehouse.com.au/news/australian-housing-values-reach-new-record-high-values-continue-rise-across-every-broad-region
17	https://www.onthehouse.com.au/news/demand-regional-housing-continues-shows-latest-corelogic-regional-market-update
18	https://www.onthehouse.com.au/news/four-charts-show-regional-returns-league-their-own
19	https://www.onthehouse.com.au/news/auction-volumes-increased-44-cent-over-december-quarter
20	https://www.onthehouse.com.au/news/hobart-and-regional-victoria-profit-most-shows-latest-corelogic-pain-and-gain-report
21	https://www.onthehouse.com.au/news/housing-markets-build-momentum-through-end-2020-pointing-strong-start-2021
22	https://www.onthehouse.com.au/news/2020-rental-market-finishes-year-strong-decade-high-monthly-growth-rate-06-december
23	https://www.onthehouse.com.au/news/retreat-property-investors-which-state-has-been-most-impacted

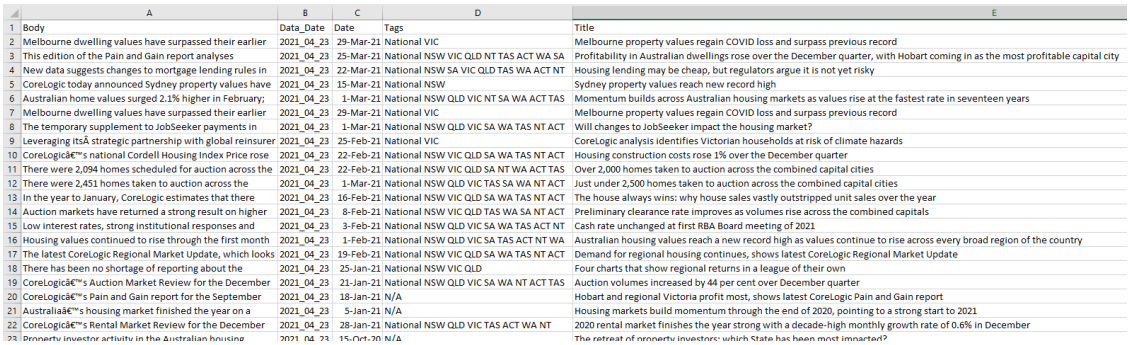
Figure 6: URL Addresses Saved to CSV

The following figure demonstrates the property news web crawler shifting towards extracting the title, date, body and tag data from each of the URL addresses collected.

```
In [56]: data = Property_News_Web_Crawler().run()
i: 134 , j: 4
i: 134 , j: 5
i: 135 , j: 1
i: 135 , j: 2
i: 135 , j: 3
i: 135 , j: 4
i: 135 , j: 5
k: 1
k: 2
k: 3
k: 4
k: 5
k: 6
k: 7
k: 8
k: 9
k: 10
k: 11
k: 12
k: 13
k: 14
```

Figure 7: Web Crawler Extracting News Article Data

The following is a screenshot of the extracted title, date, body and tag data stored as a CSV file. It also includes the date the data was extracted.



	A	B	C	D	E
1	Body	Data_Date	Date	Tags	Title
2	Melbourne dwelling values have surpassed their earlier	2021_04_23	29-Mar-21	National VIC	Melbourne property values regain COVID loss and surpass previous record
3	This edition of the Pain and Gain report analyses	2021_04_23	25-Mar-21	National NSW VIC QLD NT TAS ACT WA SA	Profitability in Australian dwellings rose over the December quarter, with Hobart coming in as the most profitable capital city
4	New data suggests changes to mortgage lending rules in	2021_04_23	22-Mar-21	National NSW SA VIC QLD TAS WA ACT NT	Housing lending may be cheap, but regulators argue it is not yet risky
5	CoreLogic today announced Sydney property values have	2021_04_23	15-Mar-21	National NSW	Sydney property values reach new record high
6	Australian home values surged 2.1% higher in February;	2021_04_23	1-Mar-21	National NSW QLD VIC NT SA WA ACT TAS	Momentum builds across Australian housing markets as values rise at the fastest rate in seventeen years
7	Melbourne dwelling values have surpassed their earlier	2021_04_23	29-Mar-21	National VIC	Melbourne property values regain COVID loss and surpass previous record
8	The temporary supplement to JobSeeker payments in	2021_04_23	1-Mar-21	National NSW QLD VIC SA WA TAS NT ACT	Will changes to JobSeeker impact the housing market?
9	Leveraging itsÃ strategic partnership with global reinsurer	2021_04_23	25-Feb-21	National VIC	CoreLogic analysis identifies Victorian households at risk of climate hazards
10	CoreLogicâ€™s national Cordell Housing Index Price rose	2021_04_23	22-Feb-21	National NSW VIC QLD SA WA TAS NT ACT	Housing construction costs rose 1% over the December quarter
11	There were 2,094 homes scheduled for auction across the	2021_04_23	22-Feb-21	National NSW VIC QLD SA NT WA ACT TAS	Over 2,000 homes taken to auction across the combined capital cities
12	There were 2,451 homes taken to auction across the	2021_04_23	1-Mar-21	National NSW QLD VIC TAS SA WA NT ACT	Just under 2,500 homes taken to auction across the combined capital cities
13	In the year to January, CoreLogic estimates that there	2021_04_23	16-Feb-21	National NSW VIC QLD SA WA TAS NT ACT	The house always wins: why house sales vastly outstripped unit sales over the year
14	Auction markets have returned a strong result on higher	2021_04_23	8-Feb-21	National NSW VIC QLD TAS WA SA NT ACT	Preliminary clearance rate improves as volumes rise across the combined capitals
15	Low interest rates, strong institutional responses and	2021_04_23	3-Feb-21	National NSW QLD VIC SA WA TAS ACT NT	Cash rate unchanged at first RBA Board meeting of 2021
16	Housing values continued to rise through the first month	2021_04_23	1-Feb-21	National NSW QLD VIC SA TAS ACT NT WA	Australian housing values reach a new record high as values continue to rise across every broad region of the country
17	The latest CoreLogic Regional Market Update, which looks	2021_04_23	19-Feb-21	National NSW VIC QLD SA WA TAS NT ACT	Demand for regional housing continues, shows latest CoreLogic Regional Market Update
18	There has been no shortage of reporting about the	2021_04_23	25-Jan-21	National NSW VIC QLD	Four charts that show regional returns in a league of their own
19	CoreLogicâ€™s Auction Market Review for the December	2021_04_23	21-Jan-21	National NSW QLD VIC SA WA NT ACT TAS	Auction volumes increased by 44 per cent over December quarter
20	CoreLogicâ€™s Pain and Gain report for the September	2021_04_23	18-Jan-21	N/A	Hobart and regional Victoria profit most, shows latest CoreLogic Pain and Gain report
21	Australiaâ€™s housing market finished the year on a	2021_04_23	5-Jan-21	N/A	Housing markets build momentum through the end of 2020, pointing to a strong start to 2021
22	CoreLogicâ€™s Rental Market Review for the December	2021_04_23	28-Jan-21	National NSW QLD VIC TAS ACT WA NT	2020 rental market finishes the year strong with a decade-high monthly growth rate of 0.6% in December
23	Property investor activity in the Australian housing	2021_04_23	15-Oct-20	N/A	The retreat of property investors: which State has been most impacted?

Figure 8: Data Extracted Saved to CSV

5 References

Caulfield, M. (2017). WHAT MAKES A TRUSTWORTHY NEWS SOURCE? Retrieved from: https://webliteracy.pressbooks.com/chapter/what-makes-a-trustworthy-news-source/

Datafiniti. (2014). Building a Web Scraper. Retrieved from: https://blog.datafiniti.co/building-a-web-scraper-f010a3d5f557

Heydt, M. (2018). Crawling with delays. Retrieved from: https://www.oreilly.com/library/view/python-web-scraping/9781787285217/9aca491e-81f1-4112-b1a8-1a0f00420b0c.xhtml

Lewallen, R. (2005). Advantages of an Object-Oriented Approach (for new programmers). Retrieved from: http://codebetter.com/raymondlewallen/2005/02/08/advantages-of-an-object-oriented-approach-for-new-programmers/

Onthehouse. (2021). Property News. Retrieved from: https://www.onthehouse.com.au

Python Software Foundation. (2021). Classes. Retrieved from: https://docs.python.org/3/tutorial/classes.html

Sahin, K. (2019). Practical XPath for Web Scraping. Retrieved from: https://www.scrapingbee.com/blog/practical-xpath-for-web-scraping/

Statcounter. (2021). Browser Market Share Worldwide. Retrieved from: https://gs.statcounter.com/browser-market-share