CS 181 - Final Project

Bao Luu, Vy Nguyen, Sean Safi

Central Question

What are the major factors contributing to the huge number of casualties of the Russia-Ukraine war?

Dataset 1: Wikipedia website - Casualties of the Russo-Ukraine War 2022 (Web-scraping 2 tables)

URL:
https://en.wikipedia.org/wiki/Casualties_of_the_Russo-Ukrainian_War#2022_Russian_invasion_of

Civilian Deaths by Area

Area	Fatalities	Time period	Source
Cherkasy Oblast	**2** killed[122][123]	24 February – 26 June 2022	Ukrainian government
Chernihiv Oblast	**700+** killed[124]	24 February – 29 March 2022	
Dnipropetrovsk Oblast	**53** killed[125]	24 April – 25 October 2022	
Donetsk Oblast (excluding Mariupol and Volnovakha)	**1,246** killed[126]	24 February – 8 December 2022	
Kharkiv Oblast	**1,600+** killed[127]	24 February – 7 December 2022	
Kherson Oblast	**467** killed[128]	24 February – 24 November 2022	
Kirovohrad Oblast	**7** killed[129]	24 February – 28 July 2022	
Kyiv Oblast	**1,596+** killed[130]	24 February – 11 October 2022	
Luhansk Oblast	**1,986+** killed[131]	24 February – 1 October 2022	
Lviv Oblast	**7** killed[132]	18 April 2022	
Mariupol	**25,000+** killed[133][c]	24 February – 25 May 2022	
Mykolaiv Oblast	**403** killed[135]	24 February –	

Civilian Deaths by Area

Area	Fatalities	Time period	Source
Western Russia	**15** killed[119]	24 February – 15 November 2022	Russian Government
Donetsk People's Republic	**1,063** killed[145]	26 February – 9 December 2022	Donetsk PR
Luhansk People's Republic	**130** killed[146]	17 February – 24 November 2022	Luhansk PR

Area	Fatalities	Time period	Source
Lublin Voivodeship	**2** killed[121]	15 November 2022	Polish government

Foreign fighters and volunteers

Excluding the Russian and Ukrainian military casualties, at least 192 combatants and volunteers, foreign citizens or foreign-born, were killed during the war. Below is a list of the nationalities of the foreign fighter casualties.

Dead foreign fighters of the 2022 Russian invasion of Ukraine

Country	Deaths	Allegiance	Reference(s)
Ukrainian Armed forces (137)			
Argentina	1	Ukrainian Foreign Legion	[174]
Australia	3	Ukrainian Foreign Legion Sich Battalion	[175]
Austria	1	Ukrainian Armed Forces	[176]
Azerbaijan	25	Armed Forces of Ukraine Georgian Legion	[177][178] [179][180] [181][182]
Belarus	16	Kastuś Kalinoŭski Battalion 24th Mechanized Brigade	[183][184][185][186]
Brazil	2	Ukrainian	[187]

Captured foreign fighters of the 2022 Russian invasion of Ukraine

Country	Captured	Allegiance	Status	Reference(s)
Ukrainian Armed forces (13)				
Belarus	2	Kastuś Kalinoŭski Battalion	Prisoners	[242]
Croatia	1	Ukrainian Foreign Legion	Released	[243][244]
Israel	1	Ukrainian Foreign Legion	Prisoner	[245]
Morocco	1	Ukrainian Foreign Legion	Released	[243][246][244]
Sweden	1	Ukrainian Foreign Legion	Released	[243][244]
		Armed		

Dataset 2: Russian total losses

URL: https://www.kaggle.com/datasets/299939cceb4d5b67df96794141a0e2ee74fd0f054ab61f72afbfbde27a0ae8e7?resource=download&select=russo-ukraine-war-casualties.json

Difficulties faced during data acquisition and our solutions

- Two tables in one segment containing two different addresses.

-> scrape using the address instead of calling out table[0] or table[1].

```
captured_country = node_table2[1].xpath('//*[@id="mw-content-text"]/div[1]/table[6]/tbody/tr/td[2]/table/tbody/tr[*]/td[1]/a/text()')
captured = node_table2[1].xpath('//*[@id="mw-content-text"]/div[1]/table[6]/tbody/tr/td[2]/table/tbody/tr[*]/td[2]/text()')
captured = [x.strip("\n") for x in captured]
captured_allegiance = node_table2[1].xpath('//*[@id="mw-content-text"]/div[1]/table[6]/tbody/tr/td[2]/table/tbody/tr[*]/td[3]/a/text()')
status = node_table2[1].xpath('//*[@id="mw-content-text"]/div[1]/table[6]/tbody/tr/td[2]/table/tbody/tr[*]/td[4]/text()')
status = [x.strip("\n") for x in status]
status[-1] = status[-2] +" "+ status.pop()
captured_forces = ['Ukrainian Armed Forces' for x in captured_country]
```

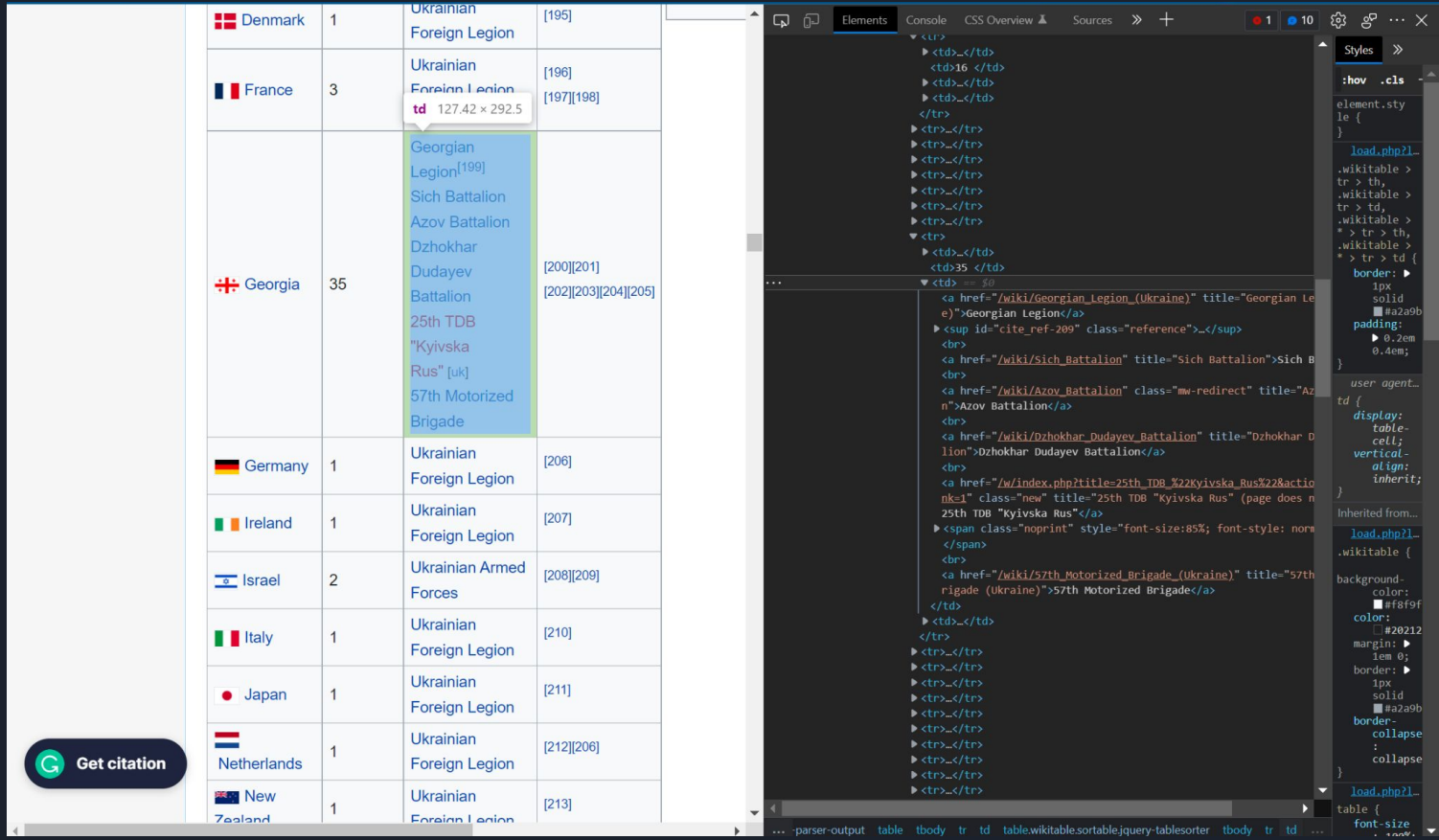
```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-17-aeb0081b4ff4> in <module>
----> 1 captured_country = node_table2[1].xpath('//*[@id="mw-content-text"]/div[1]/table[6]/tbody/tr/td[2]/table/tbody/tr[*]/td[1]/a/text()')
      2 captured = node_table2[1].xpath('//*[@id="mw-content-text"]/div[1]/table[6]/tbody/tr/td[2]/table/tbody/tr[*]/td[2]/text()')
      3 captured = [x.strip("\n") for x in captured]
      4 captured_allegiance = node_table2[1].xpath('//*[@id="mw-content-
text"]/div[1]/table[6]/tbody/tr/td[2]/table/tbody/tr[*]/td[3]/a/text()')
      5 status = node_table2[1].xpath('//*[@id="mw-content-text"]/div[1]/table[6]/tbody/tr/td[2]/table/tbody/tr[*]/td[4]/text()')

IndexError: list index out of range

SEARCH STACK OVERFLOW
```

- Not consistent amount of data in a row.

-> check for individual row and its corresponding data.



- JSON File includes Epoch/Unix times instead of actual dates and times. (Eg: 164566080)

-> Use Python datetime package for time conversion and dictionary comprehension to append the converted values to the date keys in the dictionaries of JSON Data.

First, we create a function `date_converter()` that takes in our list of epoch times and return the list of actual dates. The strftime() function converts the datetime objects into datetime strings, and its parameters are the format codes in which we wanted to customize the string of date

```
[ ] import datetime

    def date_converter(epoch_time):
      date_time = datetime.datetime.fromtimestamp(epoch_time) #datetime.fromtimestamp() converts the epoch time into DateTime
      dates = date_time.strftime( "%Y - %m - %d  %H : %M : %S")

      return dates
```

Loop through the LoD `total_casualties`, assigning the list of all values of the `date` keys in the dictionaries to variable `epoch_time`. Pass the list of epoch times in the `date_converter()` function and store all outputs in `converted` variable. Finally, assign all the `date` keys the converted values

```
[ ] for i in total_casualties:
      epoch_time = i['date']
      #print(epoch_time)
      epoch_time = epoch_time // 1000
      converted = date_converter(epoch_time)
      i['date'] = converted

    total_casualties
```

Difficulty faced during data analysis

When analyzing data using Python Pandas dataframe, the data would not properly sort because of the data type and '+' signs in the numbers

-> change the column's data type to string, delete the '+' and change back to numeric data

Area	Fatalities	Time	Source
Filter	Filter	Filter	Filter
Cherkasy Oblast	2	24 February – 26 June 2022	Ukrainian government
Chernihiv Oblast	700+	24 February – 29 March 2022	Ukrainian government
Dnipropetrovsk Oblast	53	24 April – 25 October 2022	Ukrainian government
Donetsk Oblast	1,246	24 February – 8 December 2022	Ukrainian government
Kharkiv Oblast	1,600+	24 February – 7 December 2022	Ukrainian government
Kherson Oblast	467	24 February – 24 November ...	Ukrainian government
Kirovohrad Oblast	7	24 February – 28 July 2022	Ukrainian government
Kyiv Oblast	1,596+	24 February – 11 October 2022	Ukrainian government
Luhansk Oblast	1,986+	24 February – 1 October 2022	Ukrainian government
Lviv Oblast	7	18 April 2022	Ukrainian government
Mariupol	25,000+	24 February – 25 May 2022	Ukrainian government
Mykolaiv Oblast	403	24 February – 2 August 2022	Ukrainian government
Odesa Oblast	33	24 February – 23 September ...	Ukrainian government
Poltava Oblast	22	27 June 2022	Ukrainian government
Rivne Oblast	25	24 February – 23 June 2022	Ukrainian government
Sumy Oblast	106+	24 February – 18 October 2022	Ukrainian government
Vinnytsia Oblast	23	14 July 2022	Ukrainian government
Volyn Oblast	5	24 February – 25 July 2022	Ukrainian government
Zaporizhzhia Oblast	113	24 February – 11 December ...	Ukrainian government

Table Civilian Deaths by Area - Bar chart

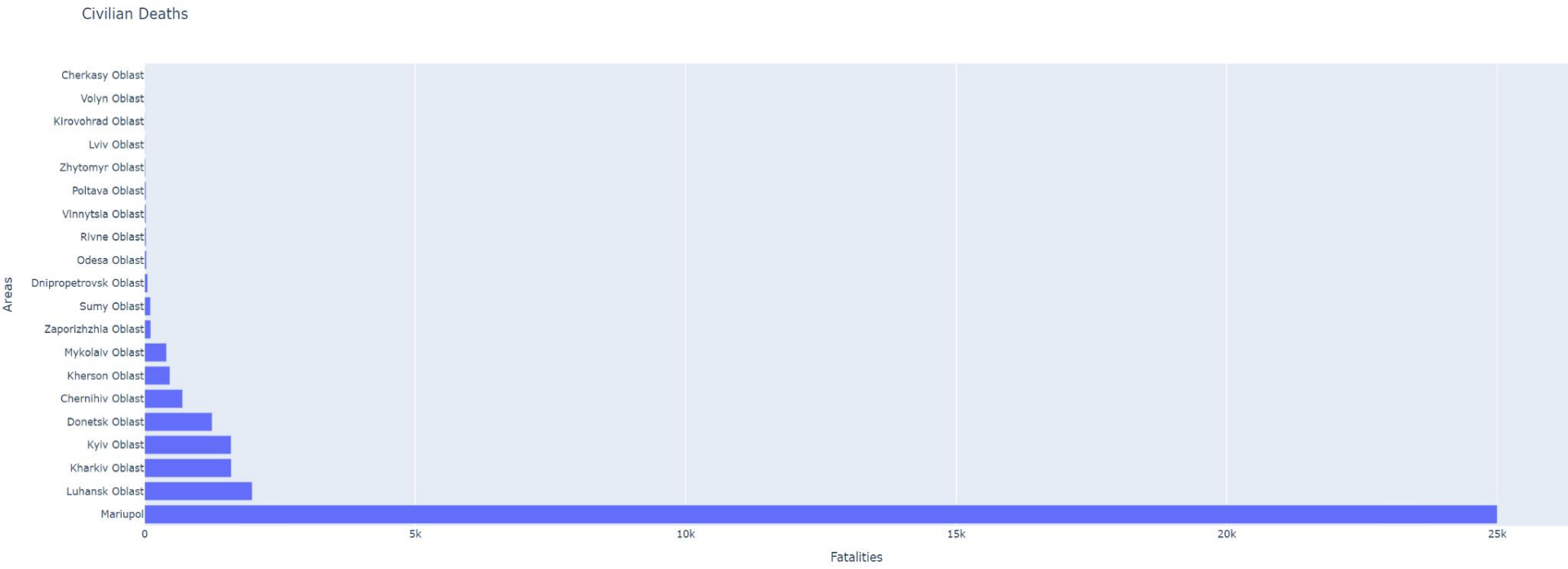


Table Dead foreign fighters - bar chart

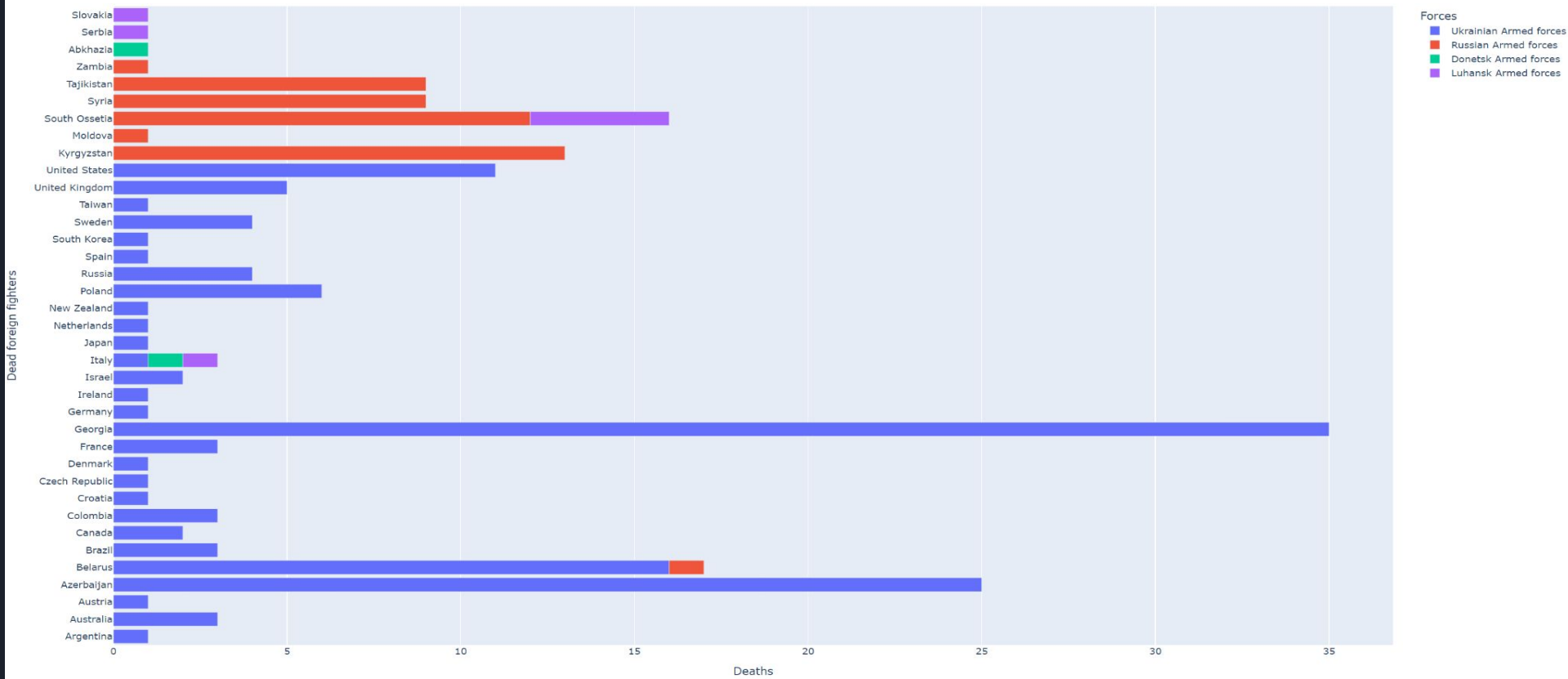


Table Russian losses - pandas DataFrame

	date	tanks	armored_vehicle	planes	helicopters	cannons	mlrs_buk	mlrs_grad	mlrs	anti_air	uav	cruise_missiles	ships	cars_cisterns	special_equipment	personnel
0	2022 - 02 - 24 00 : 00 : 00	30	130	7	6	0	0	0	0	0	0	0	0	0	0	800
1	2022 - 02 - 25 00 : 00 : 00	100	516	10	7	0	0	0	0	0	0	0	0	0	0	2800
2	2022 - 02 - 26 00 : 00 : 00	100	540	16	18	0	0	0	0	0	0	0	0	0	0	3000
3	2022 - 02 - 27 00 : 00 : 00	150	706	27	26	50	1	4	0	0	2	0	2	0	0	4500
4	2022 - 02 - 28 00 : 00 : 00	191	816	29	29	74	1	21	0	0	3	0	2	0	0	5300
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
219	2022 - 10 - 01 00 : 00 : 00	2354	4949	264	226	1397	0	0	336	176	1009	246	15	131	131	59610
220	2022 - 10 - 02 00 : 00 : 00	2377	4975	264	227	1405	0	0	337	176	1015	246	15	131	131	60110
221	2022 - 10 - 03 00 : 00 : 00	2380	4991	265	228	1405	0	0	0	176	1026	246	15	131	131	60430
222	2022 - 10 - 04 00 : 00 : 00	2424	5018	266	228	1407	0	0	340	177	1028	246	15	131	131	60800
223	2022 - 10 - 05 00 : 00 : 00	2435	5038	266	232	1414	0	0	341	177	1032	246	15	132	132	61000

224 rows × 16 columns

Answering the central question

The factors contributing to the huge number of casualties of the Russia-Ukraine war are:

1. Areas: the closer they are to Russian border, the more casualties there are
2. Majority of foreign fighters volunteering to fight alongside Ukrainians with the minority fighting for Russian army and Russian people's militas in Ukraine's Donbas region (Donetsk and Luhansk armed forces)
3. The constant increase in weapons and military vehicles being supplied to the Russian army during the war

What have we learned from the project?

- Delegation is important so we can get things done quicker
- Communication within a team is important so everyone is on the same page
- Web scraping, creating and querying SQL database
- Create graphs and charts from pandas DataFrames
- Writing clean code and using markdown cells to explain them thoroughly

Possible future work

- Analyzing the economic impacts of the war using financial data from both countries
- Analyzing the societal impacts rankings of safety, happiness, generosity
- Comparing this war to previous wars/invasions


That's all Folks!