M01_HW_KEY

January 16, 2023

1 Metadata

```
Course:  DS 5001
Moduele: 01 -- Homework KEY
```

2 Instructions

Using the notebook we reviewed in class as your guide (`M01_03_first_foray.ipynb`), extend this notebook (after the **Code** header below) to import the text contained the file `pg42324.txt` as a data frame of lines (not chunks). Once you have done this, answer the questions in **Questions**.

Submit this notebook to the Assignment in Gradescope as a PDF.

Be sure to fill out your full name and UVA ID at the top of this document.

3 Questions

3.1 What is the title of novel associated with the text file?

Answer: Frankenstein

3.2 How many tokens does the raw text have?

By raw text, we mean the text as-is, with all the Gutenberg boilerplate removed>

Answer: 80985

3.3 What are the top 10 most frequent term strings in the raw text?

Answer:

```
the     4575
and     3120
of      2918
i       2918
to      2257
my      1819
a       1497
in      1232
was     1064
that    1060
```

3.4 Compare this list with the top 10 term strings in the file we imported in class. Which subject pronoun is most frequest in each text?

Answer:

```
Persusion
-----------
the    3501
to     2862
and    2851
of     2684
a      1648
in     1439
was    1336
her    1202
had    1187
she    1143
```

- Persuasion = she
- Frankenstein = i

3.5 Provide a brief explanation for this difference, based on what you may know about the two novels.

Answer: One is written in the third first person, the other in the first (at least partly).

4 Code

```
[1]: import pandas as pd
```

```
[3]: text = pd.DataFrame(open('../../labs/data/gutenberg//pg42324.txt', 'r').
      ↪readlines(), columns=['line_str'])
     text.index.name = 'line_num'
```

4.1 Get title

```
[4]: text.head()
```

```
[4]:                                                     line_str
     line_num
     0         The Project Gutenberg EBook of Frankenstein, …
     1                                                       \n
     2         This eBook is for the use of anyone anywhere a…
     3         almost no restrictions whatsoever.  You may co…
     4         re-use it under the terms of the Project Guten…
```

```
[5]: K = text.line_str.str.split(expand=True).stack().to_frame()
     K.index.names = ['lie_num','token_num']
     K.columns = ['token_str']
```

```
[6]: K.head()
```

```
[6]:                     token_str
     lie_num token_num
     0       0                 The
             1             Project
             2           Gutenberg
             3               EBook
             4                  of
```

4.2 Find number of tokens

```
[7]: K.shape[0]
```

```
[7]: 80985
```

```
[8]: K['term_str'] = K.token_str.replace('\W+', '', regex=True).str.lower()
```

```
[9]: K.sample(10)
```

```
[9]:                     token_str   term_str
     lie_num token_num
     5756    9                 two        two
     5057    5                 but        but
     976     10           parents,    parents
     7913    8                 the        the
     5642    4                was,        was
     720     1           sweetness  sweetness
     353     4            welfare,    welfare
     7374    7                  to         to
     1560    0                   I          i
     1305    11               with       with
```

```
[10]: V = K.term_str.value_counts()
```

4.3 Get Most Frequent Words

```
[11]: V.head(10)
```

```
[11]: the     4575
      and     3120
      i       2918
      of      2918
      to      2257
      my      1819
      a       1497
      in      1232
      was     1064
```

```
that    1060
Name: term_str, dtype: int64
```

```
[12]: V
```

```
[12]: the           4575
      and           3120
      i             2918
      of            2918
      to            2257
                    …
      steal            1
      diffusing        1
      reflecting       1
      disgusting       1
      district         1
      Name: term_str, Length: 7858, dtype: int64
```

```
[ ]:
```