Data Structuring

February 13, 2021

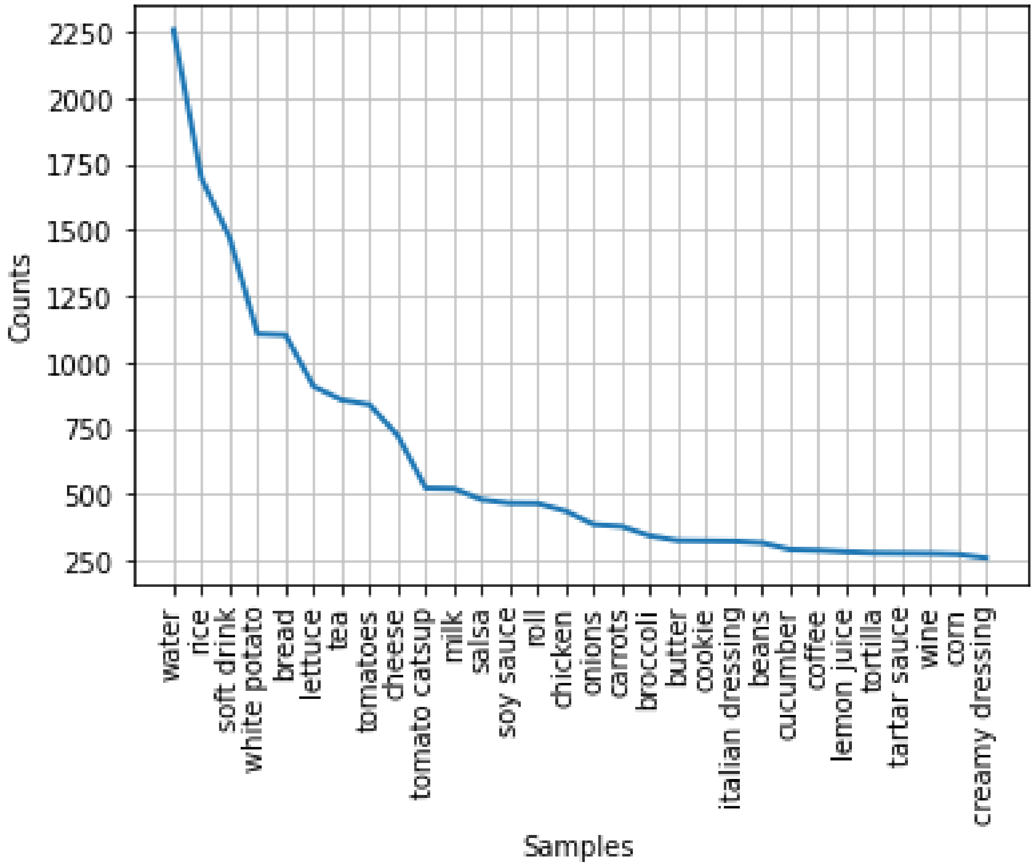
Objectives

1. Extract corpora from the food description text from the all the meals that contain seafood. Structure the corpora according the text patterns in the description. Questions: Is this an acceptable method for the analysis? The text after the comma seems descriptive of the food item, in the context of preparation method.
2. Obtain some descriptive statistics from the corpora. Identify potential issues that are relevant to the analysis objectives and address these issues.
   (1) Obtain most frequent words from corpora and seek potential issues. For example, should beverages be included? Maybe all caloric beverages (everything except water)?
   (2) Explore the item descriptions for each seafood type. The seafood types were extracted from the description category. However, seafood can be part of a dish that already includes sides. For example, there are many examples where the description contains wording like "seafood with vegetables", in which case vegetables could be categorized as a side.
   (3) Is there any interest in the descriptive food item text beyond the first comma?

[16]:

```
import pandas as pd
import re
import nltk



#Read filtered dataframe
nhanes = pd.read_pickle('../../Data/nhanes_post.pkl')



#Obtain dataframe with seafood items
seafood_df = nhanes[nhanes['DR1I_PF_SEAFD_TOT'] > 0]
#Obtain dataframe with side dishes
side_dish_df = nhanes[nhanes['DR1I_PF_SEAFD_TOT'] == 0]

"""
Obtain initial test corpus for the whole meal, seafood item only, and side␣
 ↪dishes only
Obtains the first word in the text description string before a comma, if comma␣
 ↪exists.
Obtains the whole string in the text description if comma is not present.
"""
```

```
food_type_cps = nhanes['DESCRIPTION'].apply(lambda x: re.search(r'^([^,])+', x).
 ↪group(0) if re.search((r','), x) else x)
seafood_cps = seafood_df['DESCRIPTION'].apply(lambda x: re.search(r'^([^,])+',␣
 ↪x).group(0) if re.search((r','), x) else x)
side_dish_cps = side_dish_df['DESCRIPTION'].apply(lambda x: re.
 ↪search(r'^([^,])+', x).group(0) if re.search((r','), x) else x)

#Obtain and plot frequency distribution of the side dish words
side_dish_fdist = nltk.FreqDist(side_dish_cps)
side_dish_fdist.plot(30)
```



```
[16]: <matplotlib.axes._subplots.AxesSubplot at 0x7fceaf30aa60>
```

Unconventional Side Dishes

The seafood side dish item list contains some unconventional side dishes for seafood. A sanity check can be performed on whether these items were pulled in by the filter, i.e. if there are any logic errors in the filter implementation. Some unconventional items are: milk, cookies, ice cream, sugar, banana, peanut butter.

After a check on the filtered csv output, it appears that these items are indeed associated with seafood meals. Although some of them are in a snack occasion, such as a snack that includes dried shrimp. Others are dessert items that are part of the meal.

Seafood Dish Types

This section performs some descriptive statistics on the seafood dishes, based on their type in the species classification. There are some seafood items that have multiple methods of consumptions (such as shrimp). Which items are these, and what types of seafood are more likely to be consumed as a single unique dish (i.e. not part of a lo mein, or fried rice)?

[17]:

```python
#Seafood type count based on species
seafood_species_count = seafood_df['species'].value_counts()
print(seafood_species_count.count())
print(seafood_species_count)
```

```
47
shrimp         2397
fish           1486
tuna-mixed     1174
salmon          796
seafood         749
tilapia         486
crab            398
catfish         312
cod             211
clam            201
flounder        137
tuna-fresh      125
oyster          107
shellfish       107
whiting         101
trout            95
perch            91
squid            88
sardines         79
lobster          65
scallops         56
haddock          52
porgy            46
pompano          46
croaker          45
ocean perch      44
sea bass         42
crayfish         40
mussels          33
herring          29
mackerel         28
carp             23
```

```
swordfish        19
anchovy          16
eel              13
octopus          10
sturgeon         10
frog              8
mullet            5
snails            5
halibut           5
shad              2
pike              2
shark             2
abalone           1
turtle            1
ray               1
Name: species, dtype: int64
```

We can see that there are 47 species of seafood in the survey, and their consumption count is given in the table above. What is the description distribution for each group, i.e. the variance in preparation?

[19]:

```
#Seafood type count based on species, convert to dataframe
seafood_species_count = seafood_df['species'].value_counts()
seafood_species_count = pd.DataFrame(seafood_species_count)



#Group by species, description
seafood_species_desc = seafood_df.groupby(['species', 'DESCRIPTION']).count()

#Obtain unique description count for each seafood species group
seafood_species_desc_count = seafood_df.groupby('species')['DESCRIPTION'].
 ↪nunique().sort_values(ascending=False)

#Join the frames to have unique species count and unique description in one␣
 ↪table
seafood_species_count = seafood_species_count.join(seafood_species_desc_count,␣
 ↪how='outer')
#Sort by species count, rename columns
seafood_species_count = seafood_species_count.sort_values(by = 'species',␣
 ↪ascending=False)
seafood_species_count = seafood_species_count.rename(columns={"Index":␣
 ↪"species", "species": "species_count", "DESCRIPTION":␣
 ↪"unique_description_count"})

'''
Find the number of instances of the words "and" and "with" in each description␣
 ↪item.
```

```
Convert to dataframe join with seafood dataframe and then join with the seafood␣
 ↪count
table to form a tally. Save the table as .csv
'''
seafood_description_contains_with = seafood_df['DESCRIPTION'].str.count(' with␣
 ↪')
seafood_description_contains_with = pd.
 ↪DataFrame(seafood_description_contains_with)
seafood_description_contains_with = seafood_description_contains_with.
 ↪rename(columns={"DESCRIPTION": "contains_with_count"})

seafood_description_contains_and = seafood_df['DESCRIPTION'].str.count(' and ')
seafood_description_contains_and = pd.
 ↪DataFrame(seafood_description_contains_and)
seafood_description_contains_and = seafood_description_contains_and.
 ↪rename(columns={"DESCRIPTION": "contains_and_count"})

seafood_description_contains_with = seafood_description_contains_with.
 ↪join(seafood_df, how='outer')
seafood_description_contains_with = seafood_description_contains_with.
 ↪groupby(['species'])['contains_with_count'].agg('sum')
seafood_description_contains_with = pd.
 ↪DataFrame(seafood_description_contains_with)

seafood_description_contains_and = seafood_description_contains_and.
 ↪join(seafood_df, how='outer')
seafood_description_contains_and = seafood_description_contains_and.
 ↪groupby(['species'])['contains_and_count'].agg('sum')
seafood_description_contains_and = pd.
 ↪DataFrame(seafood_description_contains_and)

seafood_species_count.reset_index(inplace=True)
seafood_species_count = pd.merge(seafood_species_count,␣
 ↪seafood_description_contains_with, how='left', left_on=['index'],␣
 ↪right_on=['species'])
seafood_species_count = pd.merge(seafood_species_count,␣
 ↪seafood_description_contains_and, how='left', left_on=['index'],␣
 ↪right_on=['species'])
```

[21]:

```
print(seafood_species_count)
```

```
        index  species_count  unique_description_count  contains_with_count  \
0      shrimp           2397                       109                  914
1        fish           1486                        81                  369
2  tuna-mixed           1174                        59                  585
3      salmon            796                        35                  197
```

```
4       seafood           749                       65                 453
5       tilapia           486                       26                 253
6          crab           398                       28                  30
7       catfish           312                       17                 138
8           cod           211                       31                  61
9          clam           201                       22                  98
10     flounder           137                       20                  39
11   tuna-fresh           125                       16                  18
12       oyster           107                       20                   1
13    shellfish           107                       14                 146
14      whiting           101                       14                  31
15        trout            95                       18                  27
16        perch            91                       20                  16
17        squid            88                       11                   0
18     sardines            79                        5                   4
19      lobster            65                       14                   5
20     scallops            56                       13                   2
21      haddock            52                       18                   1
22      pompano            46                       11                   0
23        porgy            46                       11                   0
24      croaker            45                       11                   0
25  ocean perch            44                       13                   0
26     sea bass            42                        9                   0
27     crayfish            40                        3                   0
28      mussels            33                        4                   1
29      herring            29                        9                   0
30     mackerel            28                        8                   0
31         carp            23                        8                   0
32    swordfish            19                        9                   0
33      anchovy            16                        2                   0
34          eel            13                        7                   5
35      octopus            10                        2                   0
36     sturgeon            10                        2                   0
37         frog             8                        1                   0
38      halibut             5                        3                   4
39       snails             5                        2                   0
40       mullet             5                        4                   0
41        shark             2                        2                   0
42         shad             2                        1                   0
43         pike             2                        2                   0
44          ray             1                        1                   0
45       turtle             1                        1                   0
46      abalone             1                        1                   0

    contains_and_count
0                  440
1                  254
2                  166
```

```
3                    0
4                  355
5                    0
6                   49
7                    0
8                    0
9                    0
10                   0
11                   0
12                   0
13                  21
14                   0
15                   0
16                   0
17                   0
18                   0
19                   0
20                   2
21                   0
22                   0
23                   0
24                   0
25                   0
26                   0
27                   0
28                   0
29                   0
30                   0
31                   0
32                   0
33                   0
34                   0
35                   0
36                   0
37                   0
38                   0
39                   0
40                   0
41                   0
42                   0
43                   0
44                   0
45                   0
46                   0
```

[20]:

```
seafood_df['DESCRIPTION'][(seafood_df.species == 'seafood')]
```

```
[20]: 17       seafood soup with vegetables (including carrot…
      30       seafood soup with potatoes and vegetables (exc…
      111      seafood stew with potatoes and vegetables (inc…
      340                    sushi, with vegetables and seafood
      372                                            sushi, nfs
                                     …
      44431    pasta with tomato-based sauce and seafood, hom…
      44439    seafood soup with vegetables including carrots…
      44576       pasta with cream sauce and seafood, restaurant
      44577    pasta with cream sauce, seafood, and added veg…
      44666    seafood soup with vegetables including carrots…
      Name: DESCRIPTION, Length: 749, dtype: object
```

Conclusions

From the table above, it appears that most seafood dishes are not individual seafood items. According to the item description, they are consumed in numerous ways, where the description already contains descriptive information about the seafood sides. This may require some NLP type techniques, to separate the side items from the description list, into a uniqe side item that is included in the side dish analysis.