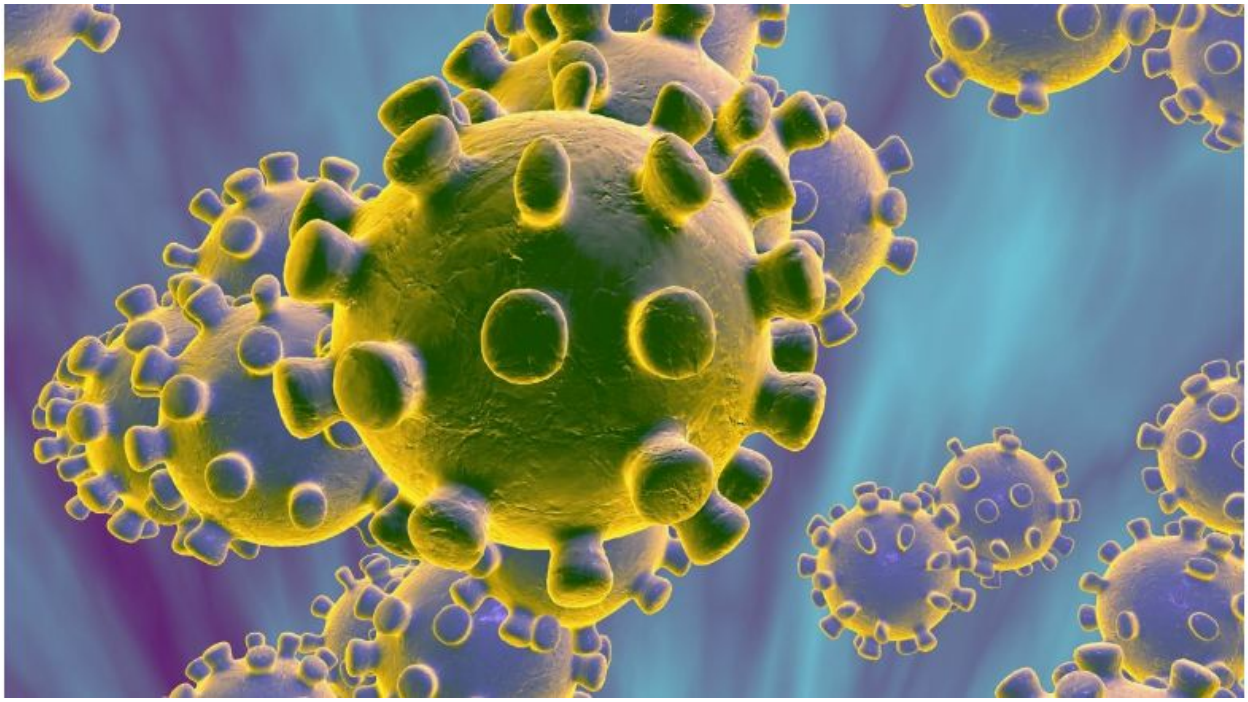


What do we know about COVID-19 risk factors?

04.03.2020

Daphney Carol Valiatingara	A20446937	dcarolvaliatingara@hawk.iit.edu
Neha Desai	A20402675	ndesai33@hawk.iit.edu
Ankit Patil	A20451742	apatil44@hawk.iit.edu

Part 1 Screenshots:

1. Data Preprocessing

- Eliminating Duplicates

```
# removing duplicate values by usung drop_duplicates
df_covid.drop_duplicates(['abstract', 'body_text'], inplace=True)
df_covid['abstract'].describe(include='all')
```

```
count     30184
unique    22480
top
freq       7677
Name: abstract, dtype: object
```

- Removing Null values

```
#Removing null values
df.dropna(inplace=True)
```

```
df.info()
```

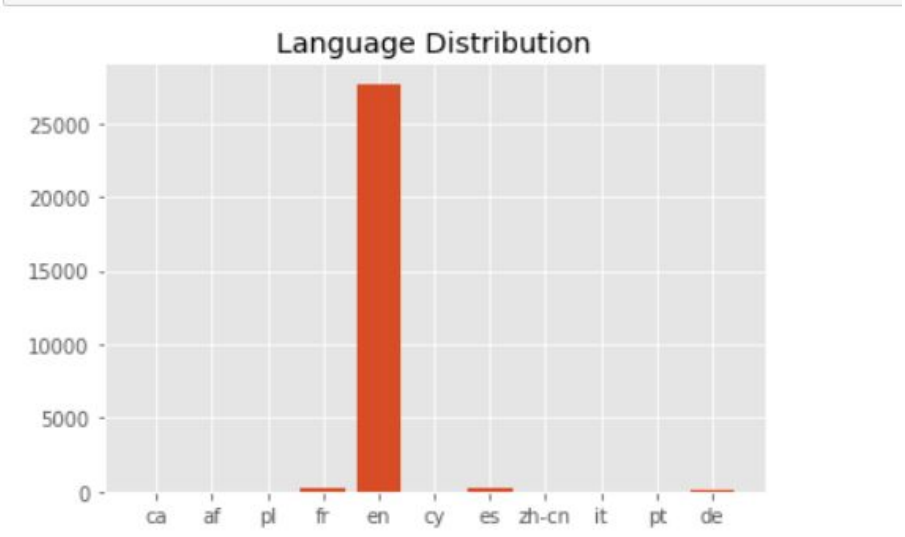
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 28314 entries, 1053 to 30196
Data columns (total 11 columns):
paper_id            28314 non-null object
doi                 28314 non-null object
abstract            28314 non-null object
body_text           28314 non-null object
authors             28314 non-null object
title               28314 non-null object
journal             28314 non-null object
abstract_summary    28314 non-null object
wcount_abstract     28314 non-null int64
wcount_body         28314 non-null int64
wcount_unique       28314 non-null int64
dtypes: int64(3), object(8)
memory usage: 2.6+ MB
```

2. Using the Langdetect package to detect different languages.

```
#Printing python data structures by pretty print module
from pprint import pprint

#Langage disctionary to store different language codes
lang_dictionary = {}
for lang in set(languages):
    lang_dictionary[lang] = languages.count(lang)

print("Total count: {}\n".format(len(languages)))
#Printing dictionary of words
pprint(lang_dictionary)
```

```
Total count: 28314

{'af': 1,
 'ca': 1,
 'cy': 1,
 'de': 48,
 'en': 27679,
 'es': 257,
 'fr': 297,
 'it': 13,
 'pl': 2,
 'pt': 14,
 'zh-cn': 1}
```

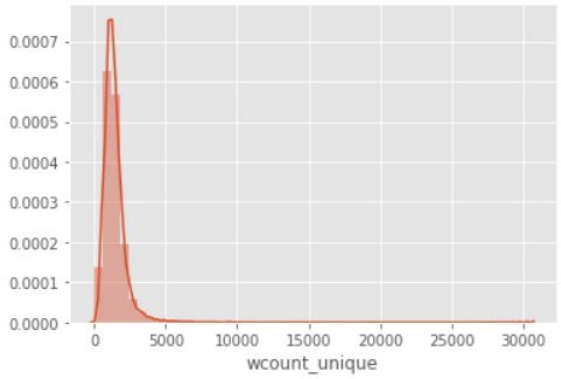
- Plotting different languages



3. Plotting words counts

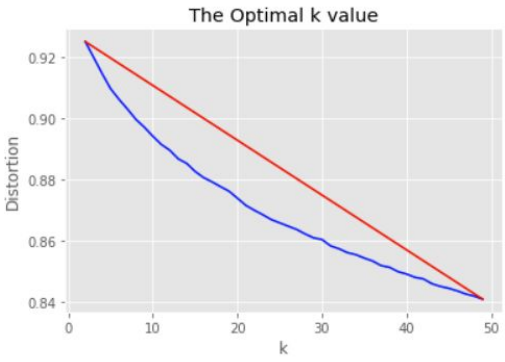
```
sns.distplot(df['wcount_unique']) #plotting unique word
df['wcount_unique'].describe()
#from the graphs we can see that most of the papers are
```

```
count    27679.000000
mean      1431.322194
std        929.678683
min         12.000000
25%        960.000000
50%       1277.000000
75%       1689.000000
max      30523.000000
Name: wcount_unique, dtype: float64
```



4. Plotting distortions and finding optimal K value

```
#Plotting the distortions
X_line = [K[0], K[-1]]
Y_line = [distortions[0], distortions[-1]]

# Plot the elbow
plt.plot(K, distortions, 'b-')
plt.plot(X_line, Y_line, 'r')
plt.xlabel('k')
plt.ylabel('Distortion')
plt.title('The Optimal k value')
plt.show()
```



5. K- means algorithm

```
#K means algorithm with optimal k value, here optimal k value is between 18 to 25
# we will use k=20
k = 20 # defining K
kmeans = KMeans(n_clusters=k, random_state=42, n_jobs=-1) #kmeans model
y_pred = kmeans.fit_predict(X_reduced_pca)
df['y'] = y_pred
```

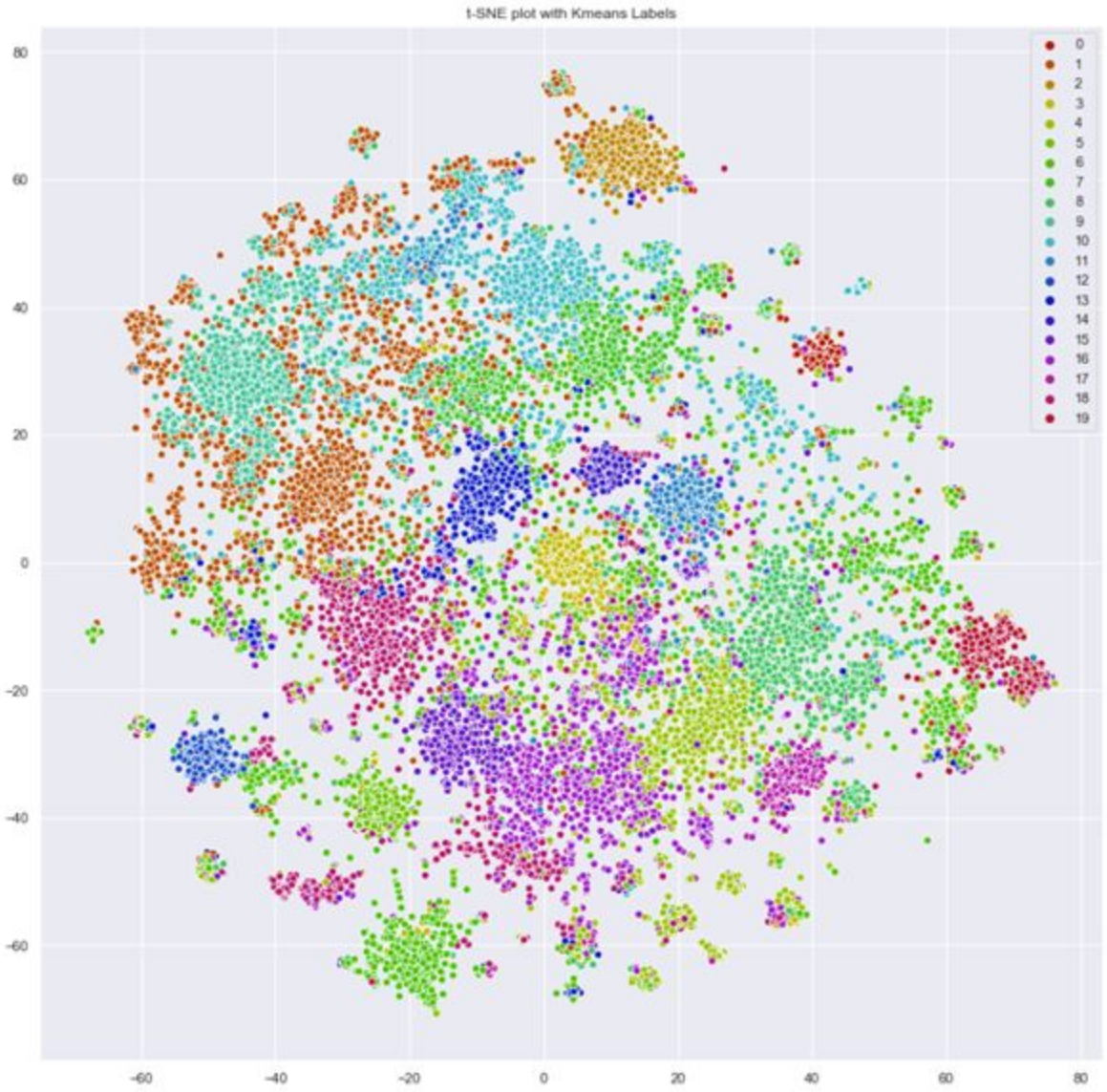
6. t-SNE algorithm

```
#t-distributed stochastic neighbouring entities
#Dimensionality reduction, visualization for high dimensional dataset
#We use TSNE to reduce the dimensions of the data, bring down higher dimensions to 2D, i.e. x-y plane
from sklearn.manifold import TSNE

tsne_data = TSNE(verbose=1, perplexity=100, random_state=42) # computing tsne
X_embedded_data = tsne_data.fit_transform(X.toarray()) #transforming the data in array
```

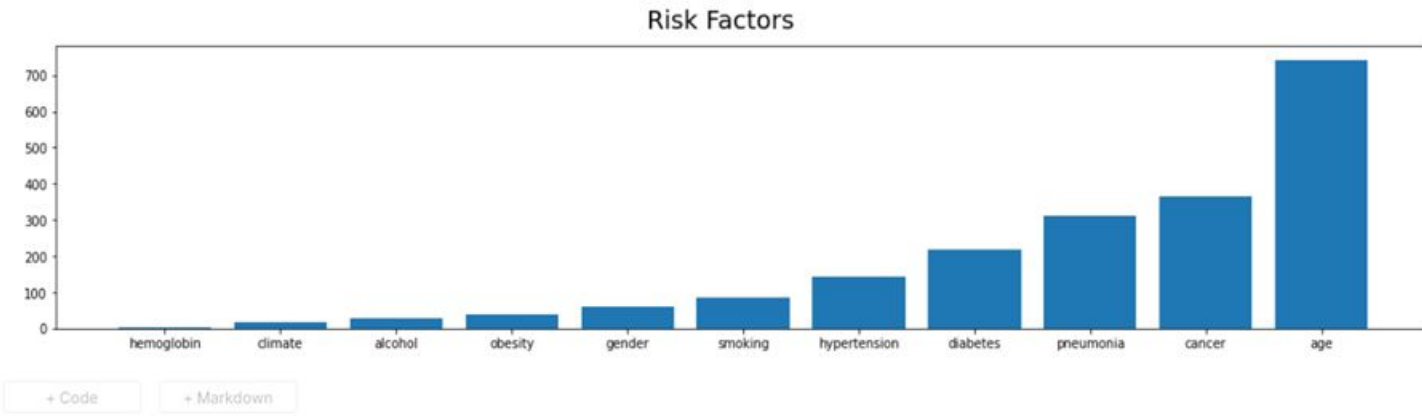
7. Plotting the data

```
#Plotting data using matplot and seaborn modules
%matplotlib inline
from matplotlib import pyplot as plt
import seaborn as sns

# sns settings for the plot
sns.set(rc={'figure.figsize':(15, 15)})

# different colors for the plot
palette = sns.hls_palette(20, l=.4, s=.9)

# plottin the plot with different colors and labels
sns.scatterplot(X_embedded_data[:,0], X_embedded_data[:,1], hue=y_pred, legend='full', palette=palette)
plt.title('t-SNE plot with Kmeans Labels')
#saving plot on the disk
plt.savefig("D:/ITM Sem 2/Data Mining/Project/Plots/cluster_tsne.png")
#displaying the plot
plt.show()
```

8. Clustered Data



- Results obtained from spaCy pattern matching are a list of risk factors and a bar graph visualizing it.



- Results of the top N papers based on query.