In [4]:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

Retriving dataset of FDI

In [5]:

```
df=pd.read_csv('/Users/shashankpatil/Desktop/pan/FDI_India.csv')
df1=pd.read_csv('/Users/shashankpatil/Desktop/pan/FDI.csv')
```

In [6]:

```
df.head(10)
```

Out[6]:

	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08
0	METALLURGICAL INDUSTRIES	22.69	14.14	36.61	8.11	200.38	149.13	169.94	1175.75
1	MINING	1.32	6.52	10.06	23.48	9.92	7.40	6.62	444.36
2	POWER	89.42	757.44	59.11	27.09	43.37	72.69	157.15	988.68
3	NON-CONVENTIONAL ENERGY	0.00	0.00	1.70	4.14	1.27	1.35	2.44	58.82
4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08
5	PETROLEUM & NATURAL GAS	9.35	211.07	56.78	80.64	102.78	12.09	87.71	1405.04
6	BOILERS AND STEAM GENERATING PLANTS	0.00	0.00	0.00	0.04	0.54	0.00	3.31	1.51
7	PRIME MOVER (OTHER THAN ELECTRICAL GENERATORS)	0.00	0.00	0.00	0.00	2.66	0.74	25.57	40.53
8	ELECTRICAL EQUIPMENTS	79.76	65.76	34.71	73.20	97.40	39.50	76.85	653.74
9	COMPUTER SOFTWARE & HARDWARE	228.39	419.39	314.24	368.32	527.90	1359.97	2613.33	1382.25

In [7]:

```
df1.head()
```

Out[7]:

	FY year	Sector	FDI
0	2000-01	METALLURGICAL INDUSTRIES	22.69
1	2001-02	METALLURGICAL INDUSTRIES	14.14
2	2002-03	METALLURGICAL INDUSTRIES	36.61
3	2003-04	METALLURGICAL INDUSTRIES	8.11
4	2004-05	METALLURGICAL INDUSTRIES	200.38

Descriptive statistics of the Dataset

In [8]:

```
df.describe()
```

Out[8]:

	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2
count	63.000000	63.000000	63.000000	63.000000	63.000000	63.000000	63.
mean	37.757302	63.931587	42.925714	34.727778	51.090317	87.932540	198.
std	112.227860	157.878737	86.606439	67.653735	101.934873	206.436967	686
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.
25%	0.000000	0.000000	0.200000	0.215000	0.715000	1.230000	4.
50%	4.030000	5.070000	11.010000	6.370000	9.090000	22.620000	25.
75%	23.510000	44.830000	36.555000	38.660000	43.205000	63.855000	108.
max	832.070000	873.230000	419.960000	368.320000	527.900000	1359.970000	4713.

In [9]:

```
df1.describe()
```

Out[9]:

	FDI
count	1071.000000
mean	309.982250
std	819.037233
min	0.000000
25%	3.140000
50%	37.940000
75%	213.740000
max	8684.070000

In [10]:

```
df.columns
```

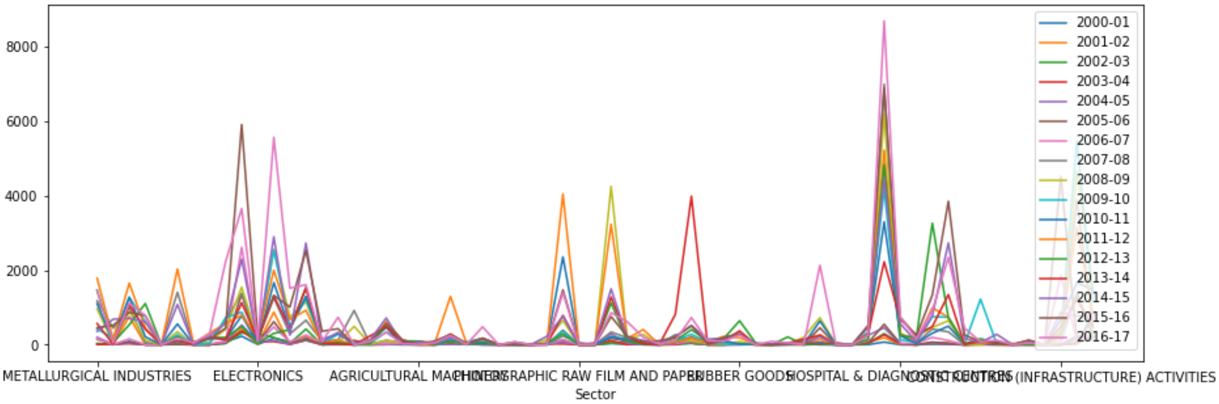
```
Out[10]: Index(['Sector', '2000-01', '2001-02', '2002-03', '2003-04', '2004-05',
                '2005-06', '2006-07', '2007-08', '2008-09', '2009-10', '2010-11',
                '2011-12', '2012-13', '2013-14', '2014-15', '2015-16', '2016-17'],
               dtype='object')
```

```
In [11]: df.set_index('Sector',inplace=True)
```

Visaulization of the FDI dataset

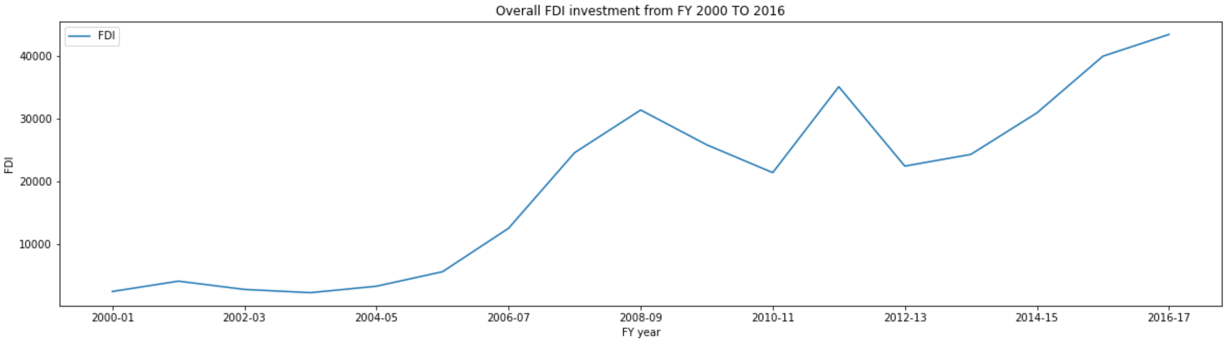
```
In [12]: df.plot(figsize=(15,5))
```

```
Out[12]: <AxesSubplot:xlabel='Sector'>
```



Overall FDI from Financial year 2000-2016

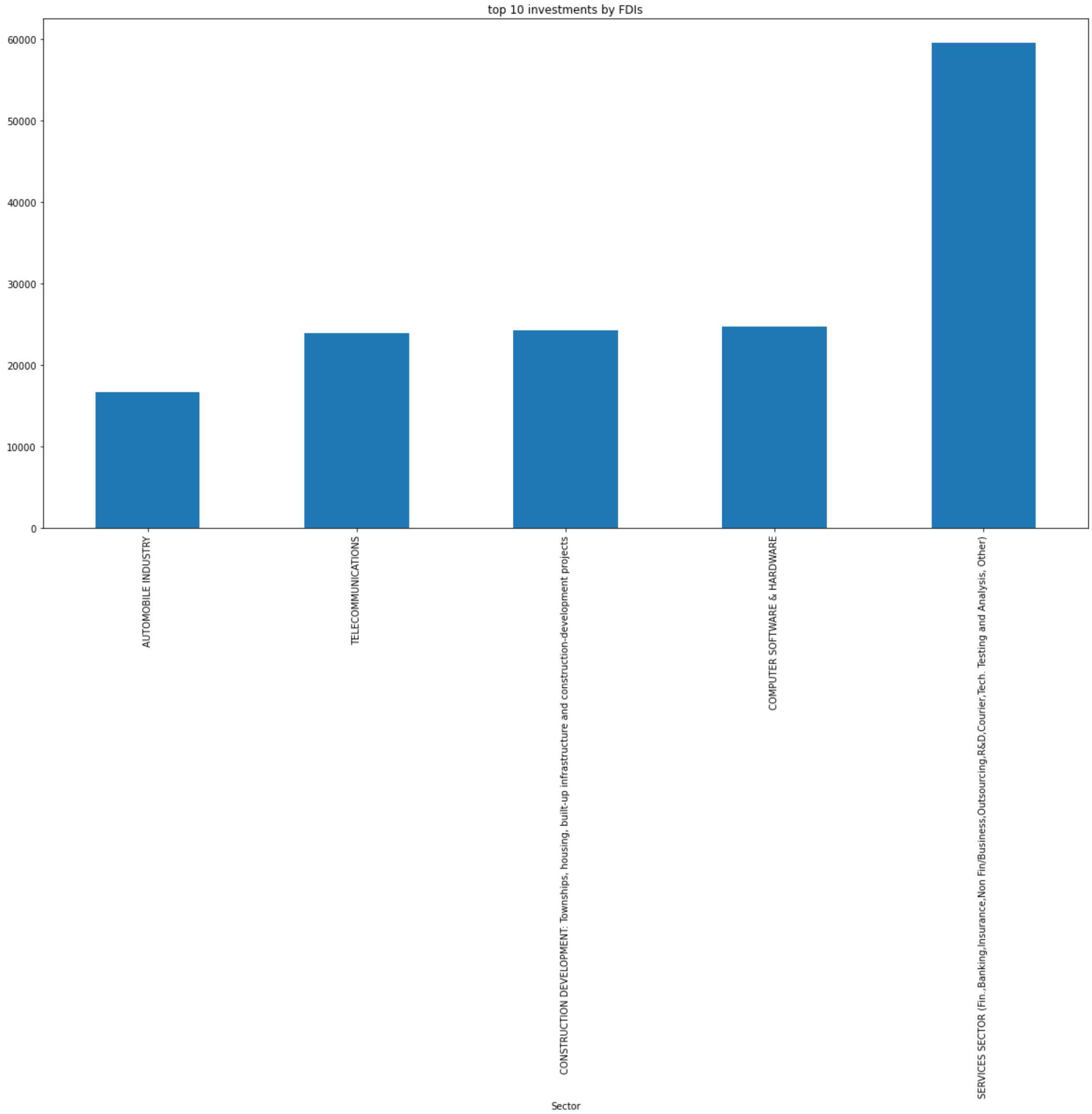
```
In [13]: df1.groupby('FY year').sum().plot(figsize=(20,5))
         plt.title('Overall FDI investment from FY 2000 TO 2016')
         plt.xlabel('FY year')
         plt.ylabel('FDI')
         plt.show()
```



Top 5 Investment Sectors by FDI

```
In [14]: df_trans = df.transpose()
         df_trans.sum().sort_values()[-5:].plot(figsize=(20,10),kind='bar', title
```

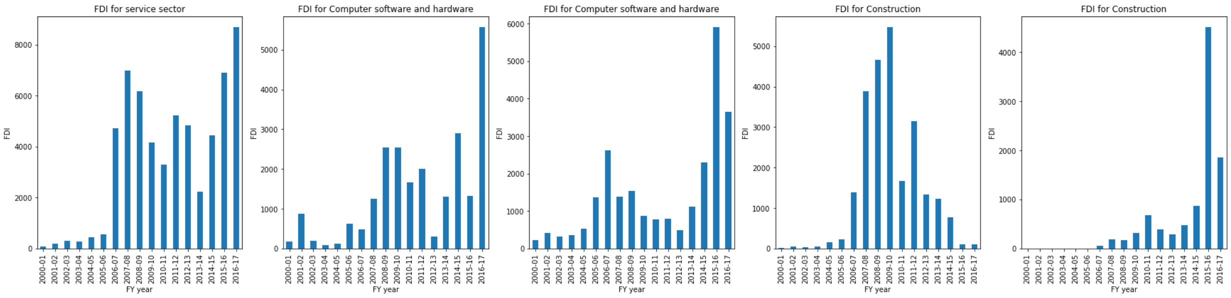
```
Out[14]: <AxesSubplot:title={'center':'top 10 investments by FDIs'}, xlabel='Secto
         r'>
```



Detailed FDI year on year for the top 5 best performing sector

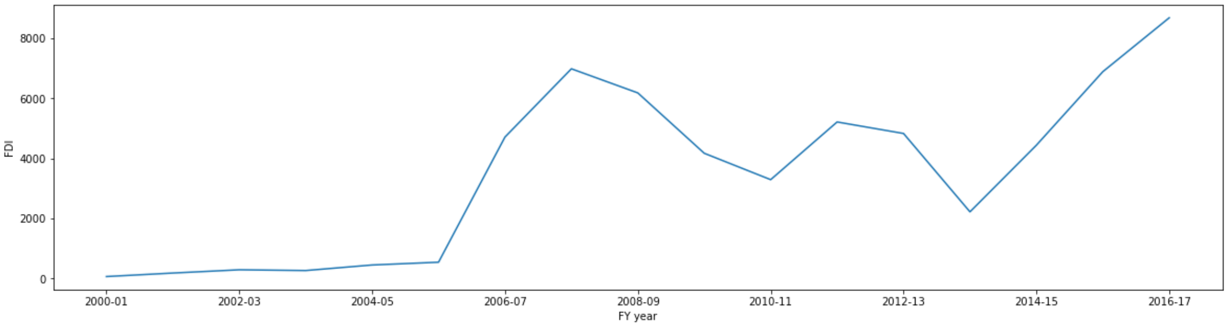
In [15]:
```
plt.subplot(1, 5, 1)
df.loc['SERVICES SECTOR (Fin.,Banking,Insurance,Non Fin/Business,Outsourc
plt.title('FDI for service sector')
plt.xlabel('FY year')
plt.ylabel('FDI')
plt.subplot(1, 5, 2)
df.loc['TELECOMMUNICATIONS'].plot(kind='bar',title='FDI for Computer soft
plt.xlabel('FY year')
plt.ylabel('FDI')
plt.subplot(1, 5, 3)
df.loc['COMPUTER SOFTWARE & HARDWARE'].plot(kind='bar',title='FDI for Com
plt.xlabel('FY year')
plt.ylabel('FDI')
plt.subplot(1, 5, 4)
df.loc['CONSTRUCTION DEVELOPMENT: Townships, housing, built-up infrastruc
plt.xlabel('FY year')
plt.ylabel('FDI')
plt.subplot(1, 5,5)
df.loc['CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES'].plot(kind='bar',title=
plt.xlabel('FY year')
plt.ylabel('FDI')
```

Out[15]:
```
Text(0, 0.5, 'FDI')
```



FDI overall growth of the best performing Sector from FY 2000-2016

In [16]:
```
df.loc['SERVICES SECTOR (Fin.,Banking,Insurance,Non Fin/Business,Outsourc
plt.xlabel('FY year')
plt.ylabel('FDI')
plt.show()
```

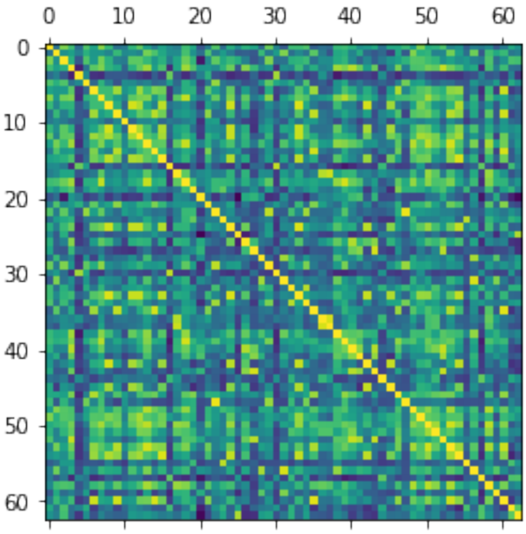


Visualizing the correlation among sectors

In [17]:
```
corr = df_trans.corr()
plt.matshow(corr)
```

Out[17]:

```
<matplotlib.image.AxesImage at 0x7fde4962a700>
```



Top 10 correlated sectors

In [18]:

```
def get_redundant_pairs(df):
    pairs_to_drop = set()
    cols = df.columns
    for i in range(0, df.shape[1]):
        for j in range(0, i+1):
            pairs_to_drop.add((cols[i], cols[j]))
    return pairs_to_drop

def get_top_abs_correlations(df, n=5):
    au_corr = df.corr().abs().unstack()
    labels_to_drop = get_redundant_pairs(df)
    au_corr = au_corr.drop(labels=labels_to_drop).sort_values(ascending=F
    return au_corr[0:n]

print("Top Absolute Correlations")
get_top_abs_correlations(df_trans,10)
```

```
Top Absolute Correlations
```

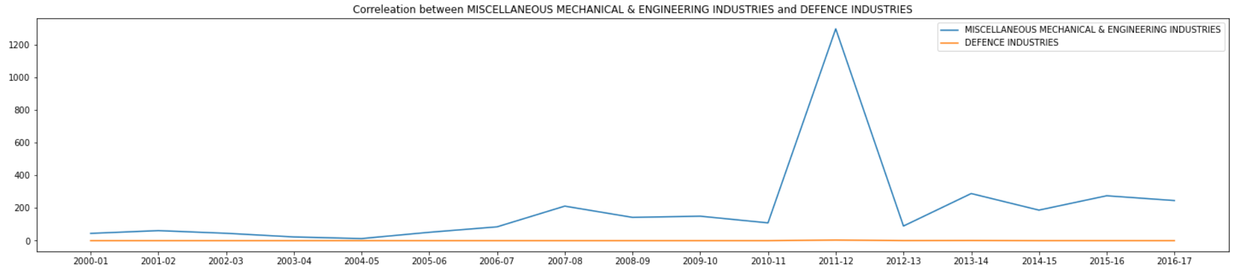
Out[18]:

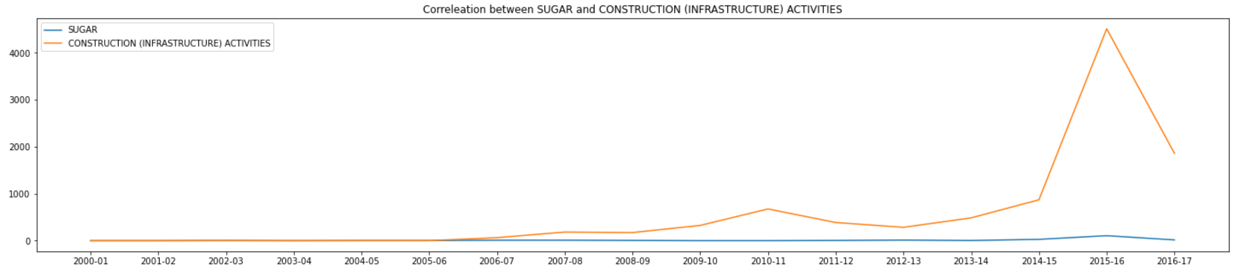
```
Sector                                             Sector
MISCELLANEOUS MECHANICAL & ENGINEERING INDUSTRIES  DEFENCE INDUSTRIES
0.958449
SUGAR                                              CONSTRUCTION (INFRASTR
UCTURE) ACTIVITIES    0.937258
ELECTRICAL EQUIPMENTS                              TEXTILES (INCLUDING DY
ED,PRINTED)           0.926705
MEDICAL AND SURGICAL APPLIANCES                    TEXTILES (INCLUDING DY
ED,PRINTED)           0.919642
SEA TRANSPORT                                      RETAIL TRADING
0.918936
DYE-STUFFS                                         DIAMOND,GOLD ORNAMENTS
0.916723
AIR TRANSPORT (INCLUDING AIR FREIGHT)              CONSTRUCTION (INFRASTR
UCTURE) ACTIVITIES    0.916622
FERMENTATION INDUSTRIES                            FOOD PROCESSING INDUST
RIES                  0.910990
ELECTRICAL EQUIPMENTS                              GLUE AND GELATIN
0.908833
MATHEMATICAL,SURVEYING AND DRAWING INSTRUMENTS     GLASS
0.908687
dtype: float64
```

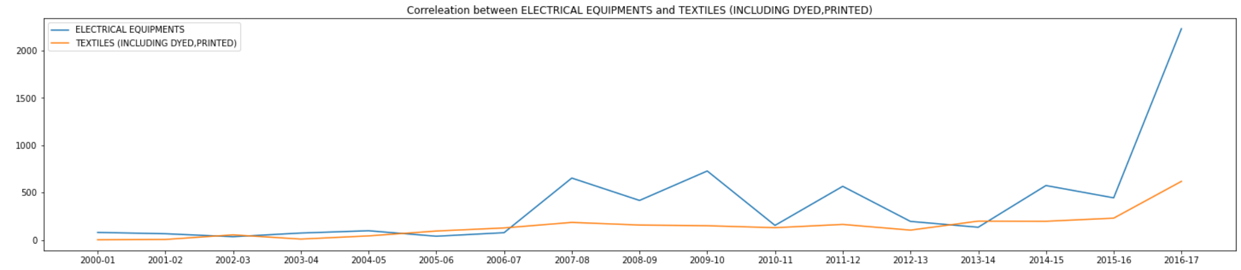
Plotting the correlated sector

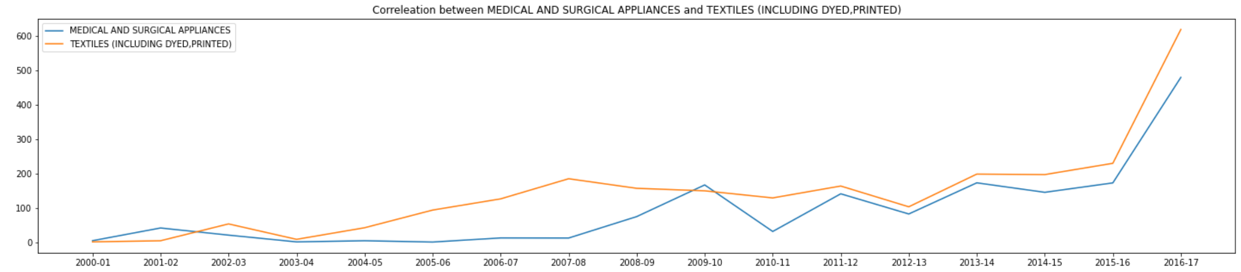
In [72]:

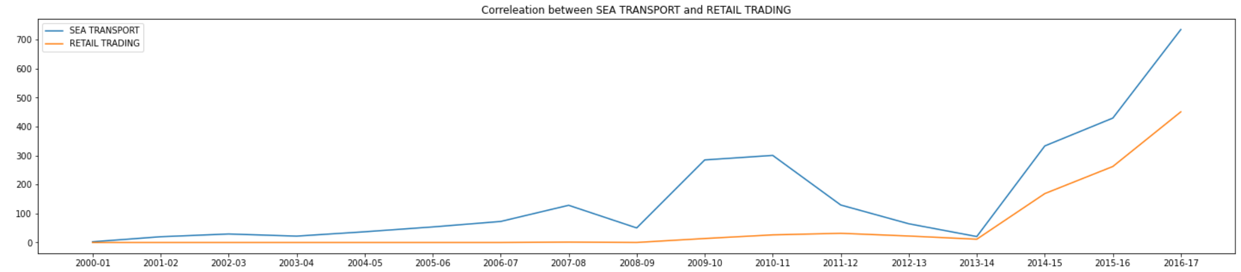
```python
abs_corr = get_top_abs_correlations(df_trans,10)
plt.rcParams['figure.figsize'] = [25, 5]
for i in range(10):
  plt.plot(df.loc[abs_corr.index[i][0]], label = abs_corr.index[i][0])
  plt.plot(df.loc[abs_corr.index[i][1]], label = abs_corr.index[i][1])
  plt.title(f'Correleation between {abs_corr.index[i][0]} and {abs_corr.i
  plt.legend()
  plt.show()
```

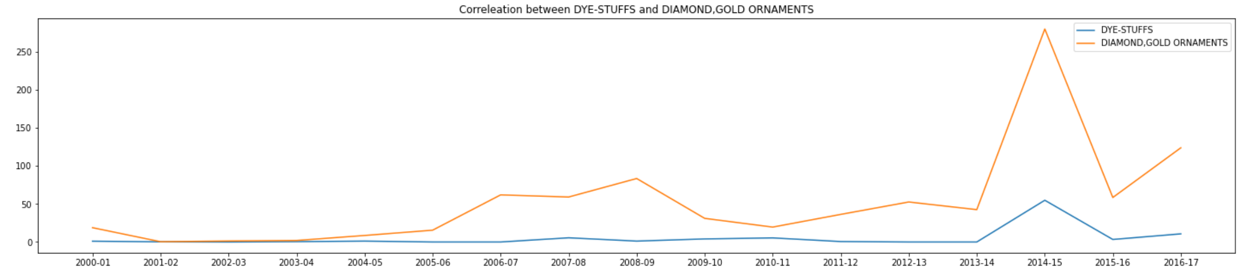

Correleation between MISCELLANEOUS MECHANICAL & ENGINEERING INDUSTRIES and DEFENCE INDUSTRIES

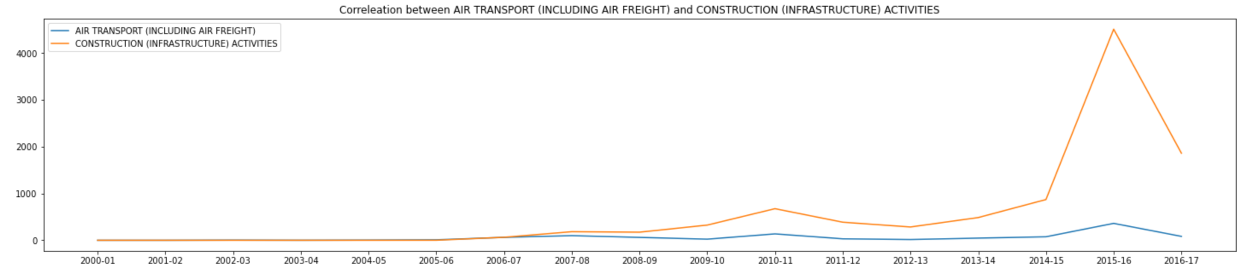

Correleation between SUGAR and CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES

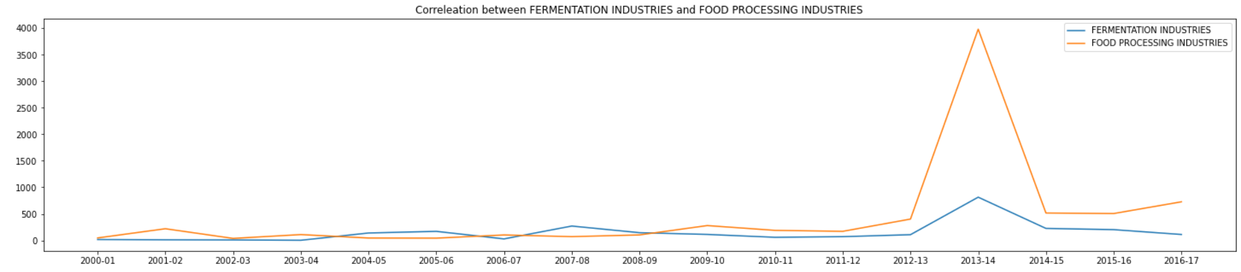

Correleation between ELECTRICAL EQUIPMENTS and TEXTILES (INCLUDING DYED,PRINTED)
ELECTRICAL EQUIPMENTS
TEXTILES (INCLUDING DYED,PRINTED)
2000
1500
1000
500
0
2000-01
2001-02
2002-03
2003-04
2004-05
2005-06
2006-07
2007-08
2008-09
2009-10
2010-11
2011-12
2012-13
2013-14
2014-15
2015-16
2016-17

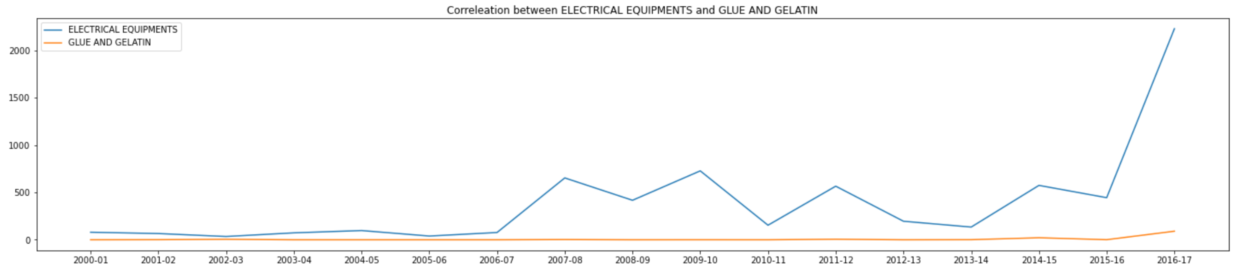

Correleation between MEDICAL AND SURGICAL APPLIANCES and TEXTILES (INCLUDING DYED,PRINTED)
MEDICAL AND SURGICAL APPLIANCES
TEXTILES (INCLUDING DYED,PRINTED)
600
500
400
300
200
100
0
2000-01
2001-02
2002-03
2003-04
2004-05
2005-06
2006-07
2007-08
2008-09
2009-10
2010-11
2011-12
2012-13
2013-14
2014-15
2015-16
2016-17

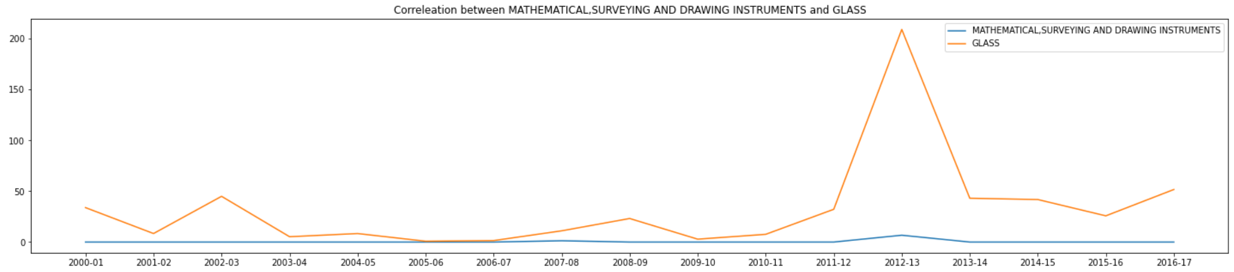

Correleation between SEA TRANSPORT and RETAIL TRADING
SEA TRANSPORT
RETAIL TRADING
700
600
500
400
300
200
100
0
2000-01
2001-02
2002-03
2003-04
2004-05
2005-06
2006-07
2007-08
2008-09
2009-10
2010-11
2011-12
2012-13
2013-14
2014-15
2015-16
2016-17


Correleation between DYE-STUFFS and DIAMOND,GOLD ORNAMENTS
DYE-STUFFS
DIAMOND,GOLD ORNAMENTS
250
200
150
100
50
0
2000-01
2001-02
2002-03
2003-04
2004-05
2005-06
2006-07
2007-08
2008-09
2009-10
2010-11
2011-12
2012-13
2013-14
2014-15
2015-16
2016-17


Correleation between AIR TRANSPORT (INCLUDING AIR FREIGHT) and CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES
AIR TRANSPORT (INCLUDING AIR FREIGHT)
CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES
4000
3000
2000
1000
0
2000-01
2001-02
2002-03
2003-04
2004-05
2005-06
2006-07
2007-08
2008-09
2009-10
2010-11
2011-12
2012-13
2013-14
2014-15
2015-16
2016-17

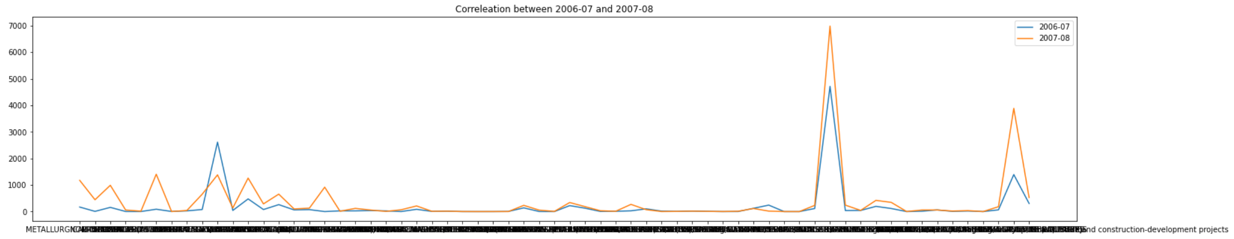

Correleation between FERMENTATION INDUSTRIES and FOOD PROCESSING INDUSTRIES
FERMENTATION INDUSTRIES
FOOD PROCESSING INDUSTRIES
4000
3500
3000
2500
2000
1500
1000
500
0
2000-01
2001-02
2002-03
2003-04
2004-05
2005-06
2006-07
2007-08
2008-09
2009-10
2010-11
2011-12
2012-13
2013-14
2014-15
2015-16
2016-17

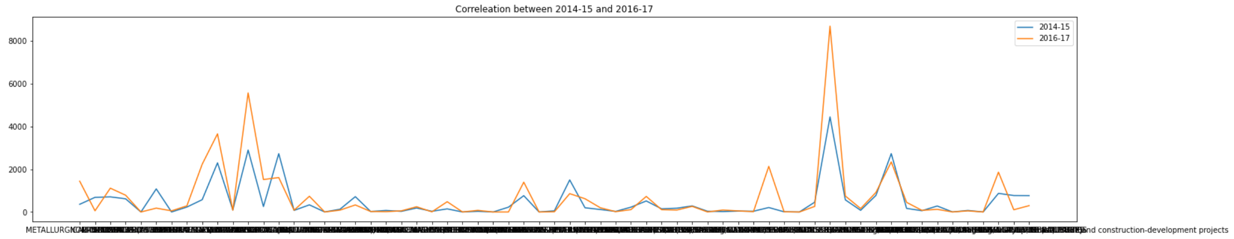


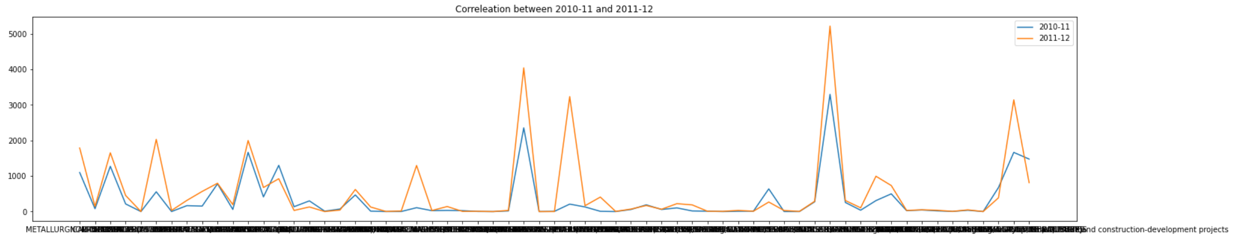


Lets look at the 3 most correlated years

```
In [56]: abs_corr = get_top_abs_correlations(df,3)
         abs_corr
         for i in range(3):
           plt.plot(df_trans.loc[abs_corr.index[i][0]], label = abs_corr.index[i][
           plt.plot(df_trans.loc[abs_corr.index[i][1]], label = abs_corr.index[i][
           plt.title(f'Correleation between {abs_corr.index[i][0]} and {abs_corr.i
           plt.legend()
           plt.show()
```







```
In [ ]:
```

```
In [ ]:
```

```
In [ ]:
```