

```
In [2]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings("ignore")
```

```
In [3]: df=pd.read_csv("FDI data.csv")
df.head()
```

```
Out[3]:
```

	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08	2008-09	2009-10
0	METALLURGICAL INDUSTRIES	22.69	14.14	36.61	8.11	200.38	149.13	169.94	1175.75	959.94	419.88
1	MINING	1.32	6.52	10.06	23.48	9.92	7.40	6.62	444.36	34.16	174.40
2	POWER	89.42	757.44	59.11	27.09	43.37	72.69	157.15	988.68	907.66	1271.79
3	NON-CONVENTIONAL ENERGY	0.00	0.00	1.70	4.14	1.27	1.35	2.44	58.82	125.88	622.52
4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08	0.22	0.00

```
In [4]: sec=df['Sector'].unique()
for i in sec:
    print(i)
```

METALLURGICAL INDUSTRIES
MINING
POWER
NON-CONVENTIONAL ENERGY
COAL PRODUCTION
PETROLEUM & NATURAL GAS
BOILERS AND STEAM GENERATING PLANTS
PRIME MOVER (OTHER THAN ELECTRICAL GENERATORS)
ELECTRICAL EQUIPMENTS
COMPUTER SOFTWARE & HARDWARE
ELECTRONICS
TELECOMMUNICATIONS
INFORMATION & BROADCASTING (INCLUDING PRINT MEDIA)
AUTOMOBILE INDUSTRY
AIR TRANSPORT (INCLUDING AIR FREIGHT)
SEA TRANSPORT
PORTS
RAILWAY RELATED COMPONENTS
INDUSTRIAL MACHINERY
MACHINE TOOLS
AGRICULTURAL MACHINERY
EARTH-MOVING MACHINERY
MISCELLANEOUS MECHANICAL & ENGINEERING INDUSTRIES
COMMERCIAL, OFFICE & HOUSEHOLD EQUIPMENTS
MEDICAL AND SURGICAL APPLIANCES
INDUSTRIAL INSTRUMENTS
SCIENTIFIC INSTRUMENTS
MATHEMATICAL,SURVEYING AND DRAWING INSTRUMENTS
FERTILIZERS
CHEMICALS (OTHER THAN FERTILIZERS)
PHOTOGRAPHIC RAW FILM AND PAPER
DYE-STUFFS
DRUGS & PHARMACEUTICALS
TEXTILES (INCLUDING DYED,PRINTED)
PAPER AND PULP (INCLUDING PAPER PRODUCTS)
SUGAR
FERMENTATION INDUSTRIES
FOOD PROCESSING INDUSTRIES
VEGETABLE OILS AND VANASPATHI
SOAPS, COSMETICS & TOILET PREPARATIONS
RUBBER GOODS
LEATHER,LEATHER GOODS AND PICKERS
GLUE AND GELATIN
GLASS
CERAMICS
CEMENT AND GYPSUM PRODUCTS
TIMBER PRODUCTS
DEFENCE INDUSTRIES
CONSULTANCY SERVICES
SERVICES SECTOR (Fin.,Banking,Insurance,Non Fin/Business,Outsourcing,R&D,Courier,Tech. Testing and Analysis, Other)
HOSPITAL & DIAGNOSTIC CENTRES
EDUCATION
HOTEL & TOURISM
TRADING
RETAIL TRADING
AGRICULTURE SERVICES
DIAMOND,GOLD ORNAMENTS
TEA AND COFFEE (PROCESSING & WAREHOUSING COFFEE & RUBBER)
PRINTING OF BOOKS (INCLUDING LITHO PRINTING INDUSTRY)
COIR
CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES
CONSTRUCTION DEVELOPMENT: Townships, housing, built-up infrastructure and construc

tion-development projects
MISCELLANEOUS INDUSTRIES

In []:

In [4]: `df.columns`

Out[4]: 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 [5]: Year = ['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']
Sectors = ['Sector']

df_transformed = pd.melt(df, id_vars = Sectors, value_vars = Year, var_name='Year',
                        value_name='Total Investment', ignore_index=True)
df_transformed=round(df_transformed,2)
df_transformed
```

Out[5]:

	Sector	Year	Total Investment
0	METALLURGICAL INDUSTRIES	2000-01	22.69
1	MINING	2000-01	1.32
2	POWER	2000-01	89.42
3	NON-CONVENTIONAL ENERGY	2000-01	0.00
4	COAL PRODUCTION	2000-01	0.00
...
1066	PRINTING OF BOOKS (INCLUDING LITHO PRINTING IN...	2016-17	53.17
1067	COIR	2016-17	0.00
1068	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	2016-17	1860.73
1069	CONSTRUCTION DEVELOPMENT: Townships, housing, ...	2016-17	105.14
1070	MISCELLANEOUS INDUSTRIES	2016-17	296.40

1071 rows × 3 columns

In [90]: `df_transformed.shape`

Out[90]: (1071, 3)

In [8]: `df_transformed.to_csv("up.csv")`

```
In [85]: df_transformed['Sector']=df_transformed['Sector'].replace(["CONSTRUCTION DEVELOPMEN
                        , "SERVICES SECTOR (Fin.,Banking,Insurance,Non Fin/Bus
                        , 'TEA AND COFFEE (PROCESSING & WAREHOUSING COFFEE & R
                        , ["CONSTRUCTION DEVELOPMENT", "SERVICES SECTOR", 'TEA AN
```

```
In [7]: Metallurgical=df_transformed[df_transformed["Sector"]=="METALLURGICAL INDUSTRIES"]
Metallurgical
```

Out[7]:

	Sector	Year	Total Investment
0	METALLURGICAL INDUSTRIES	2000-01	22.69
63	METALLURGICAL INDUSTRIES	2001-02	14.14
126	METALLURGICAL INDUSTRIES	2002-03	36.61
189	METALLURGICAL INDUSTRIES	2003-04	8.11
252	METALLURGICAL INDUSTRIES	2004-05	200.38
315	METALLURGICAL INDUSTRIES	2005-06	149.13
378	METALLURGICAL INDUSTRIES	2006-07	169.94
441	METALLURGICAL INDUSTRIES	2007-08	1175.75
504	METALLURGICAL INDUSTRIES	2008-09	959.94
567	METALLURGICAL INDUSTRIES	2009-10	419.88
630	METALLURGICAL INDUSTRIES	2010-11	1098.14
693	METALLURGICAL INDUSTRIES	2011-12	1786.14
756	METALLURGICAL INDUSTRIES	2012-13	1466.23
819	METALLURGICAL INDUSTRIES	2013-14	567.63
882	METALLURGICAL INDUSTRIES	2014-15	359.34
945	METALLURGICAL INDUSTRIES	2015-16	456.31
1008	METALLURGICAL INDUSTRIES	2016-17	1440.18

In [73]: `sec=df['Sector'].unique()`

```

In [8]: fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[0]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title('METALLURGICAL INDUSTRIES',loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[1]],x='Year',y='Total Inve
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title('MINING',loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[2]],x='Year',y='Total Inve
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title('POWER',loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[4]],x='Year',y='Total Inve
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")

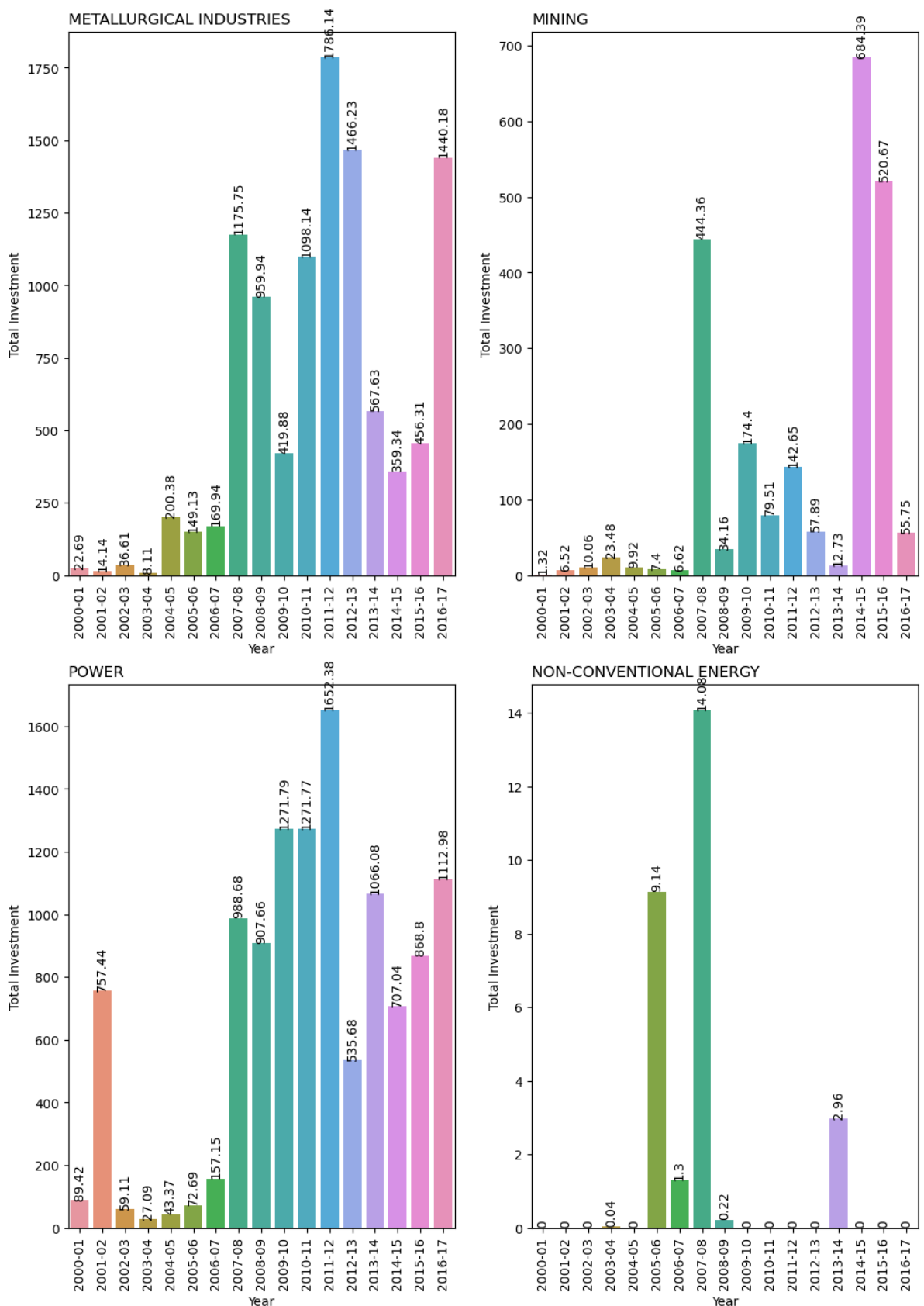
```

```

for label in labels:
    label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title('NON-CONVENTIONAL ENERGY',loc='left')

plt.show()

```



```

In [11]: fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]=="sec[4]",x='Year',y='Total Investment'])
for container in ax1.containers:

```

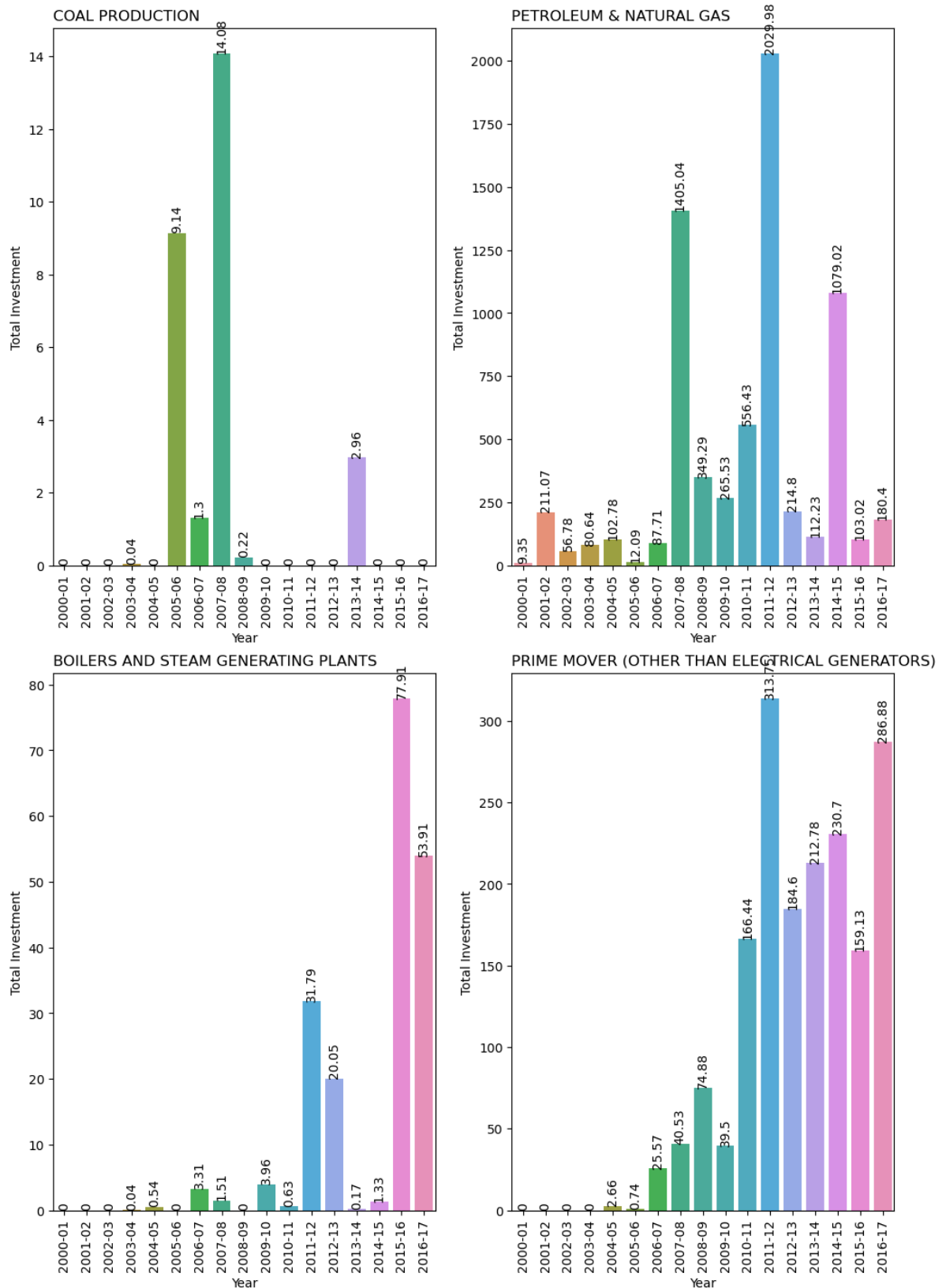
```
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[4],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[5]],x='Year',y='Total Inve
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[5],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[6]],x='Year',y='Total Inve
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[6],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[7]],x='Year',y='Total Inve
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[7],loc='left')

plt.show()
```



```
In [10]: fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[8]],x='Year',y='Total Investment')
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
for label in labels:
    label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[8],loc='left')

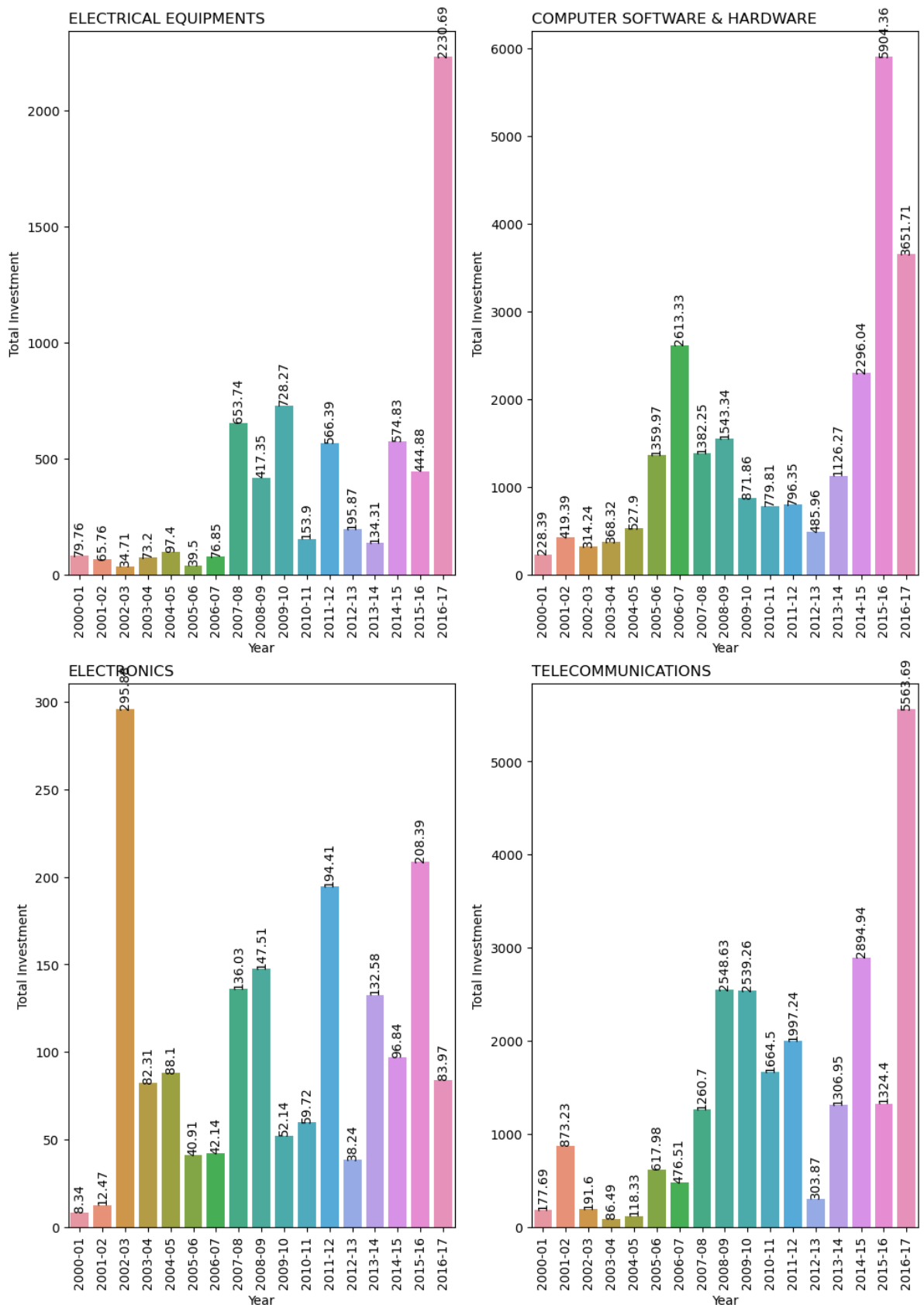
sns.barplot(df_transformed[df_transformed["Sector"]==sec[9]],x='Year',y='Total Investment')
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
```

```
        for label in labels:
            label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[9],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[10]],x='Year',y='Total Inv
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[10],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[11]],x='Year',y='Total Inv
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[11],loc='left')

plt.show()
```

```
In [13]: fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[12]],x='Year',y='Total Inv
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[12],loc='left')

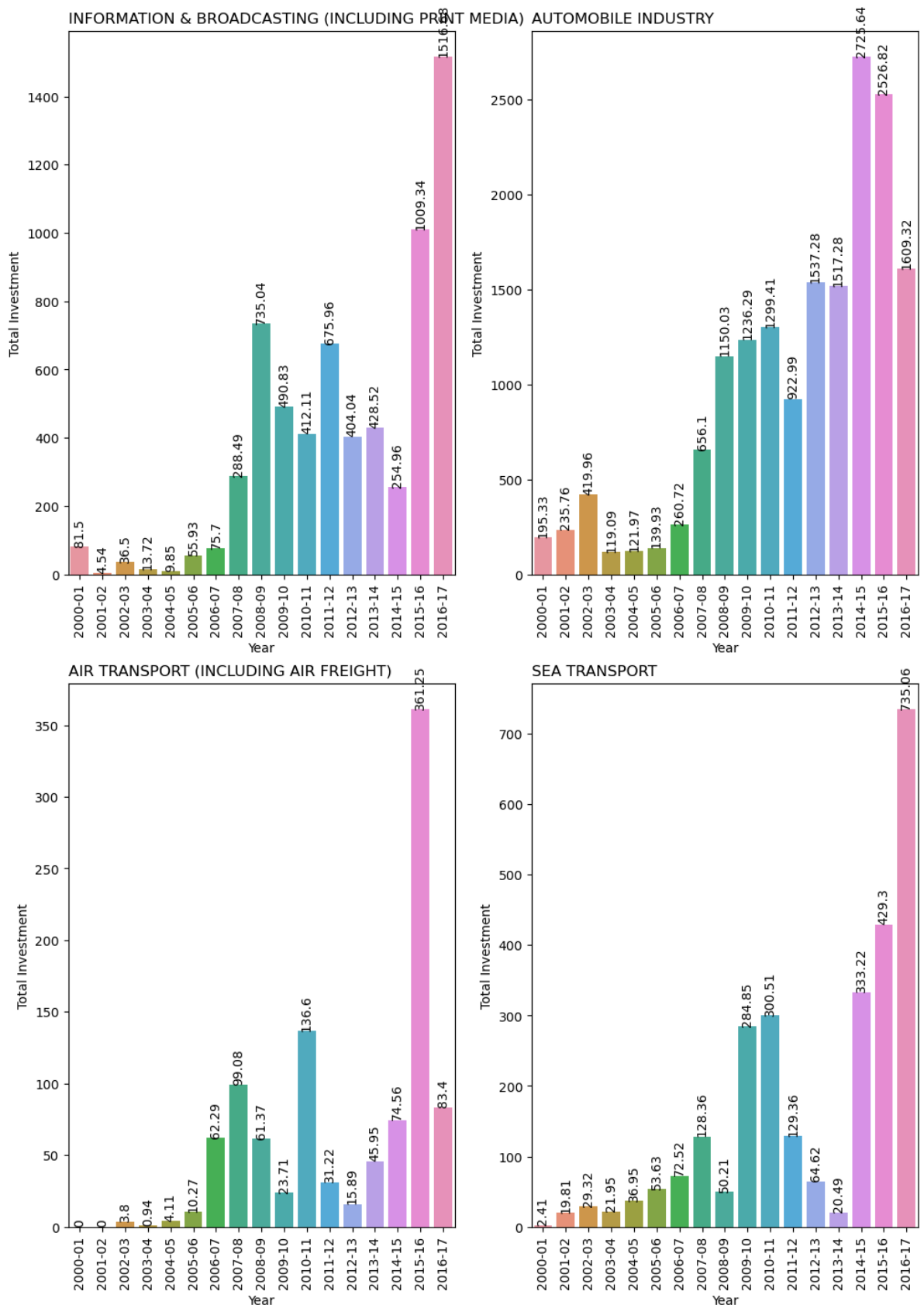
sns.barplot(df_transformed[df_transformed["Sector"]==sec[13]],x='Year',y='Total Inv
```

```
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[13],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[14]],x='Year',y='Total Inv
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[14],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[15]],x='Year',y='Total Inv
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[15],loc='left')

plt.show()
```



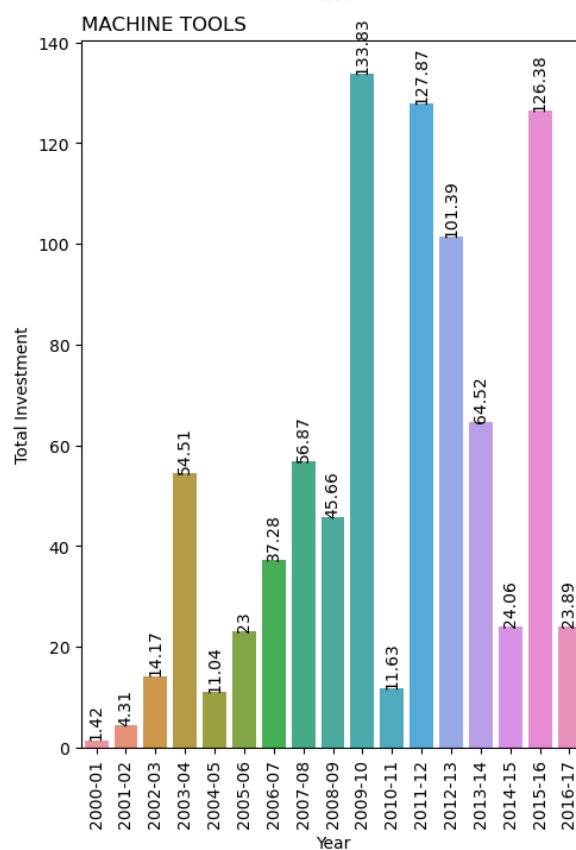
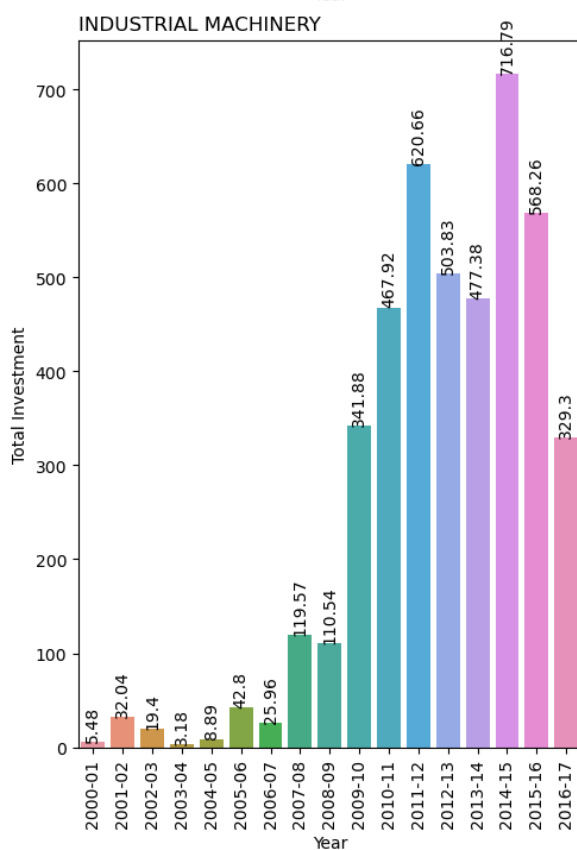
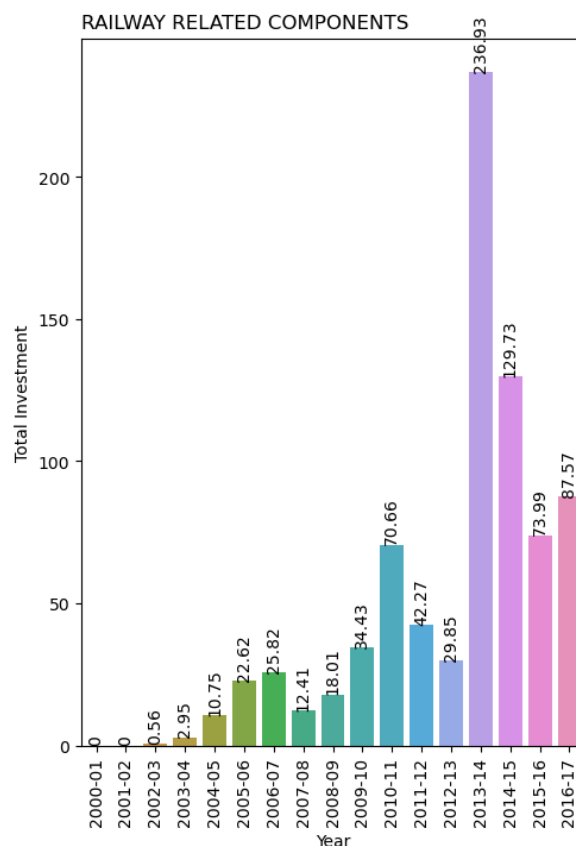
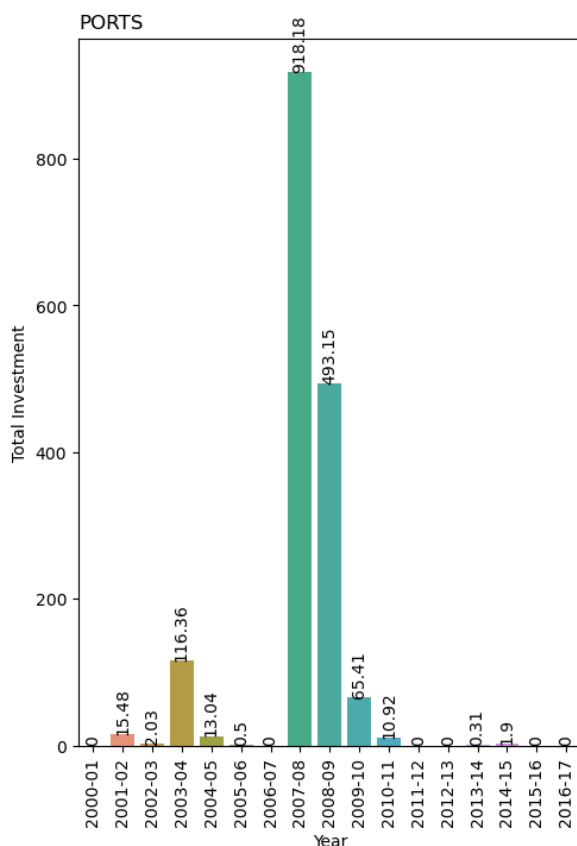
```
In [14]: i=16
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Investment')
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
for label in labels:
    label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```



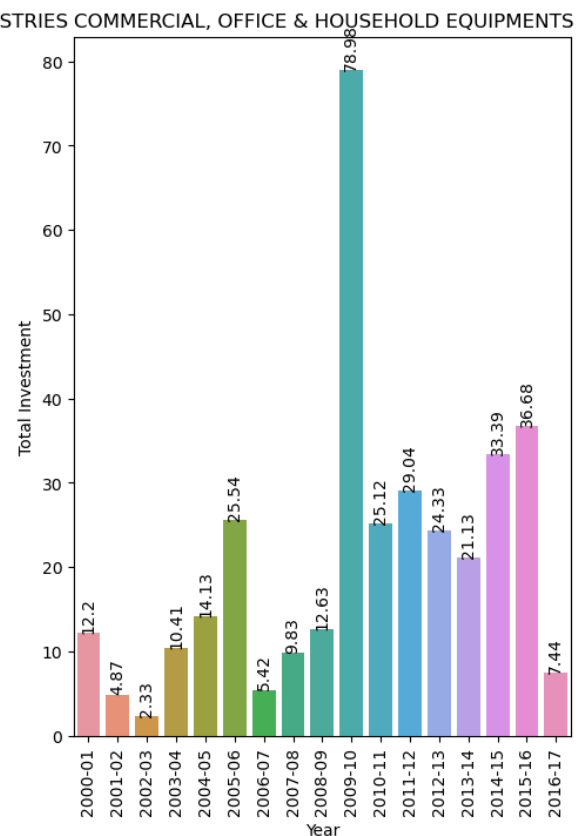
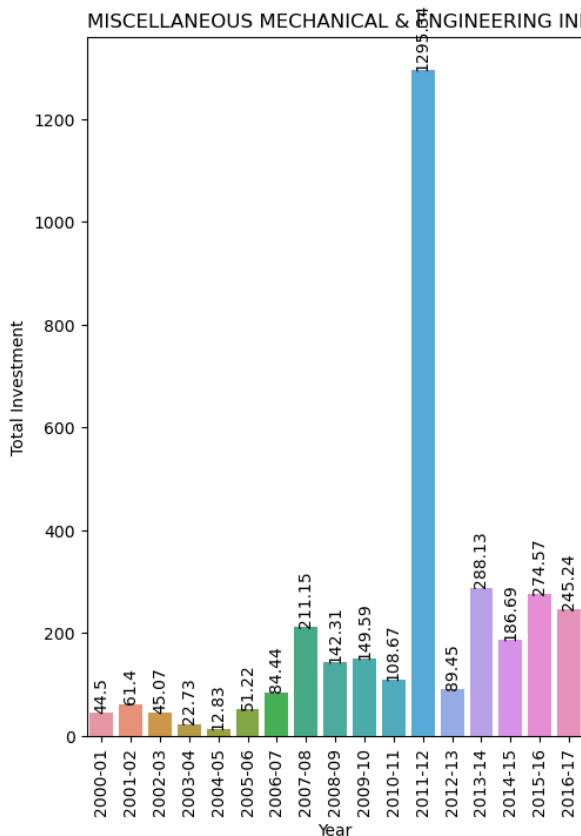
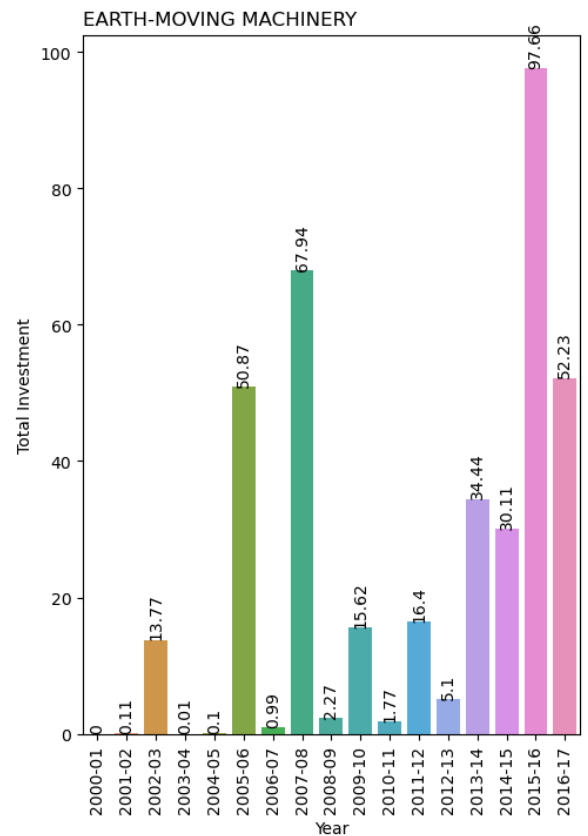
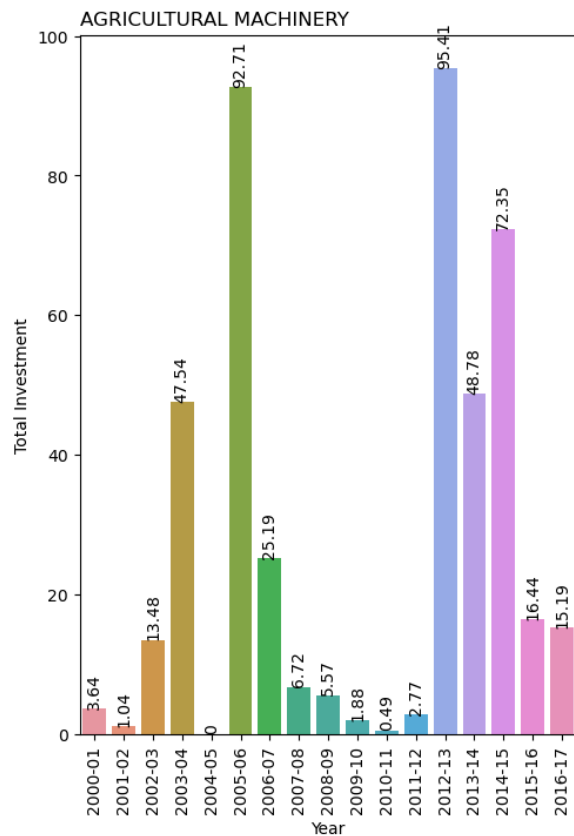
```
In [15]: i=20
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```



```
In [16]: i=24
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')

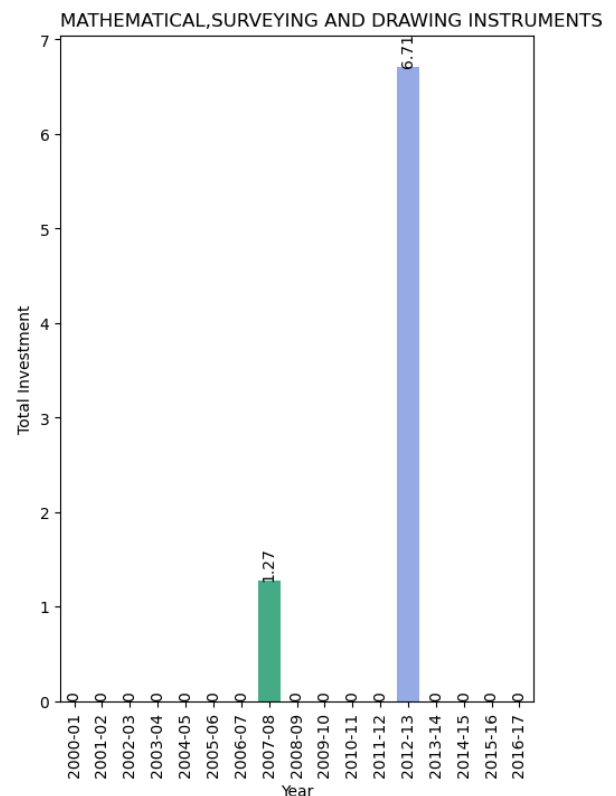
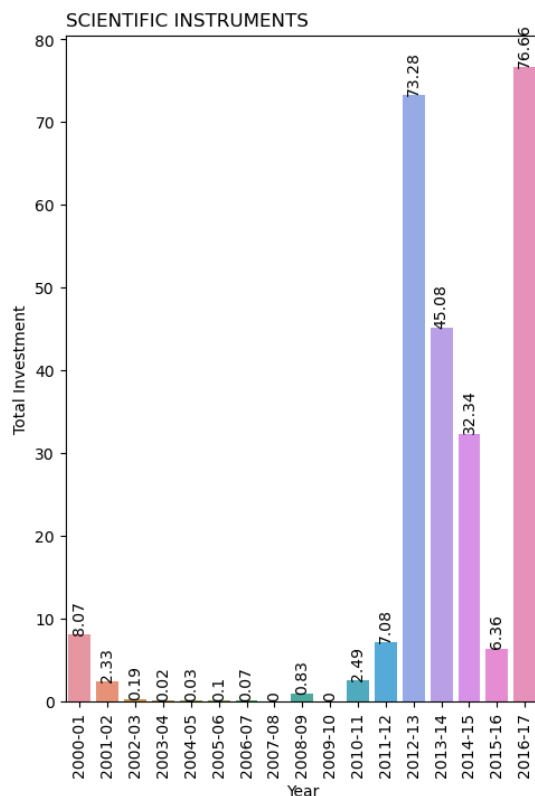
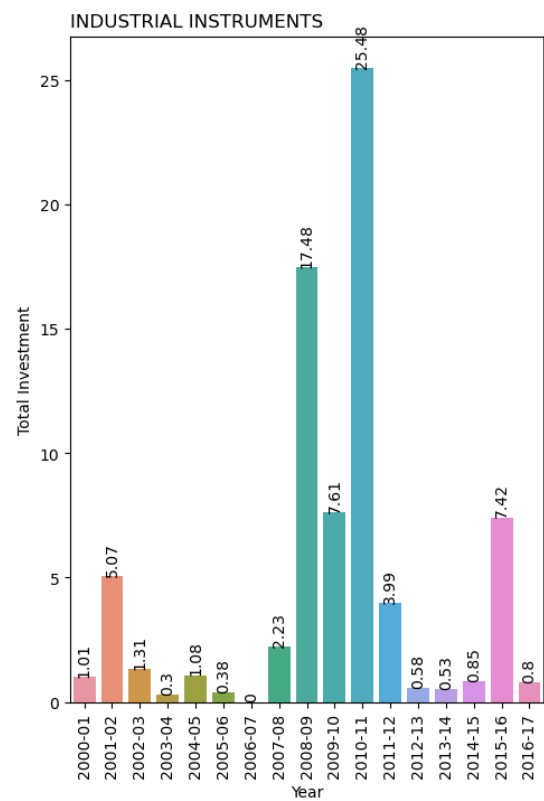
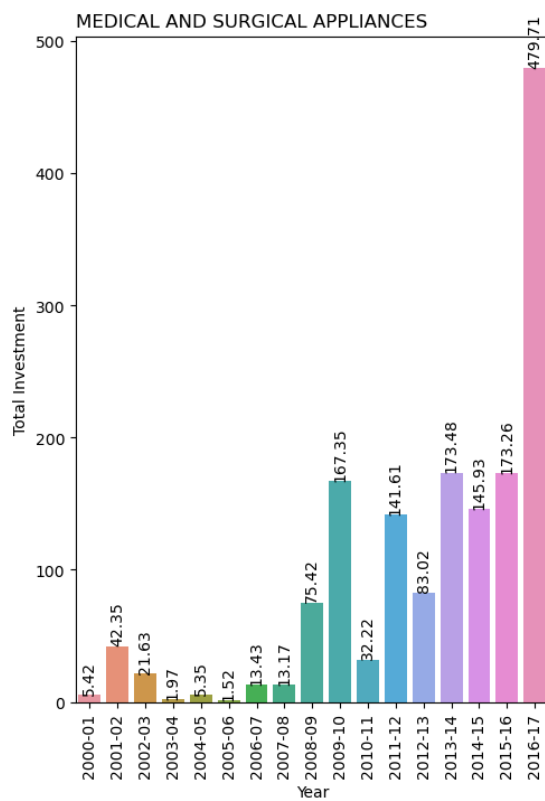
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
```

```
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```

```
In [17]: i=28
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')

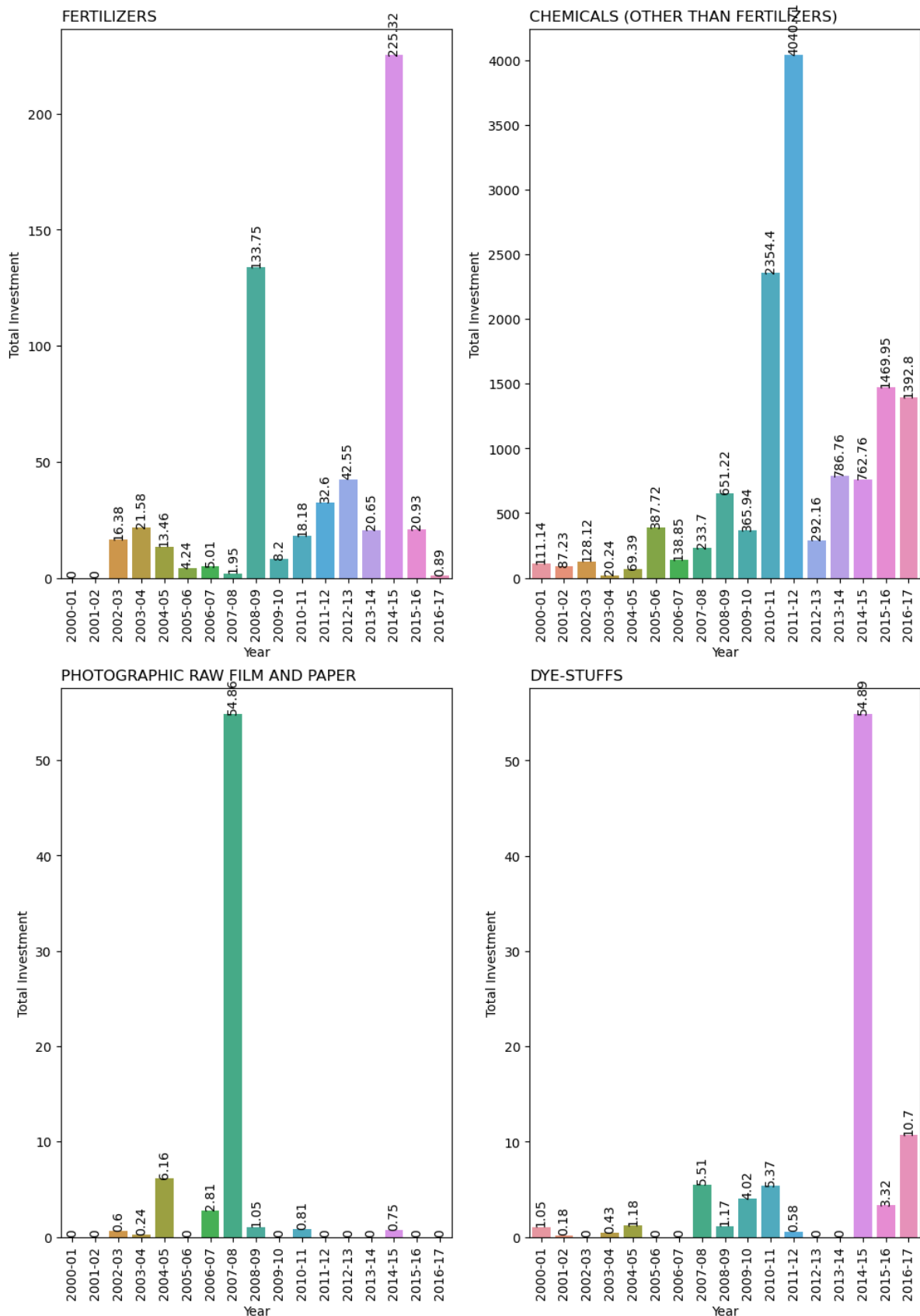
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
```

```
        for label in labels:
            label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```



```

In [18]: i=32
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')

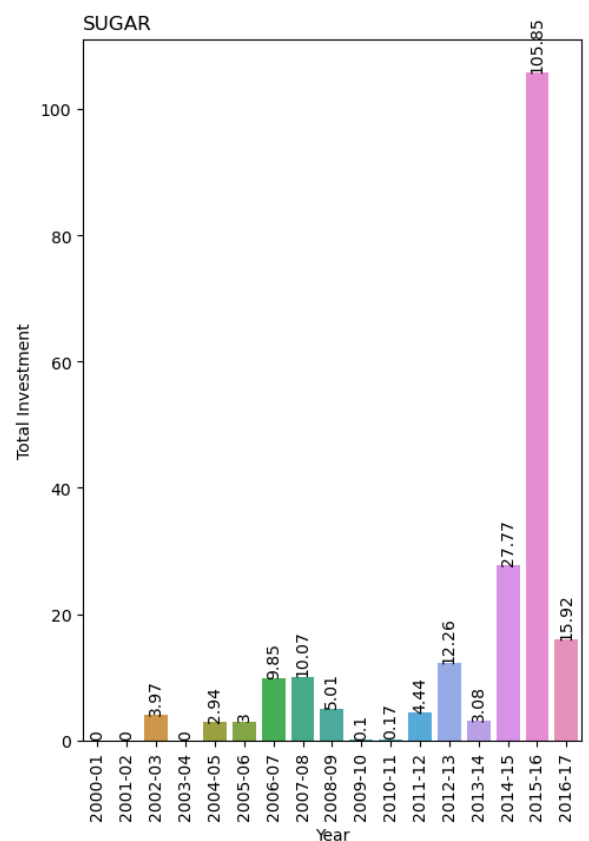
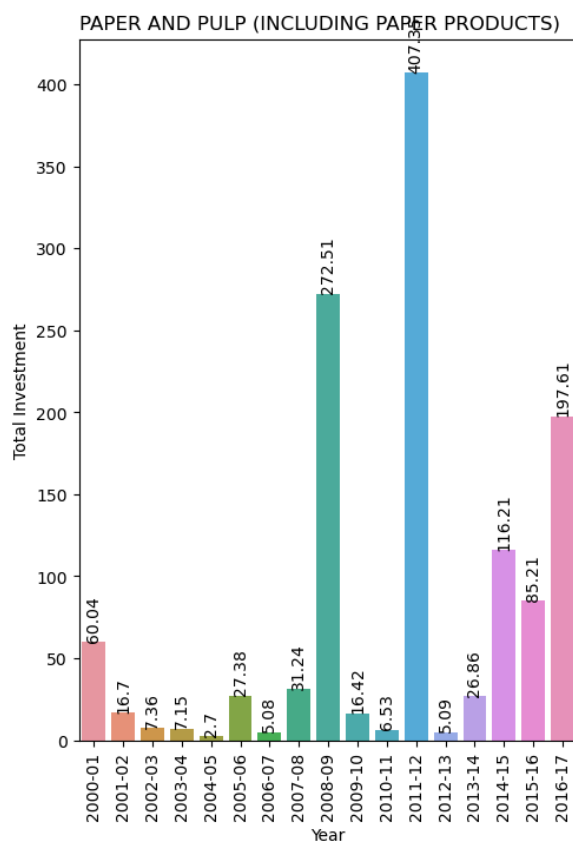
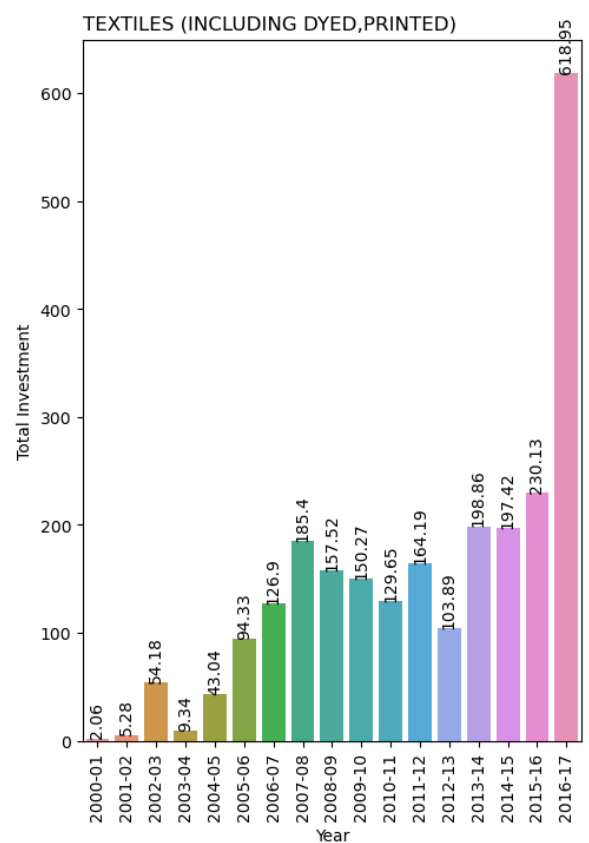
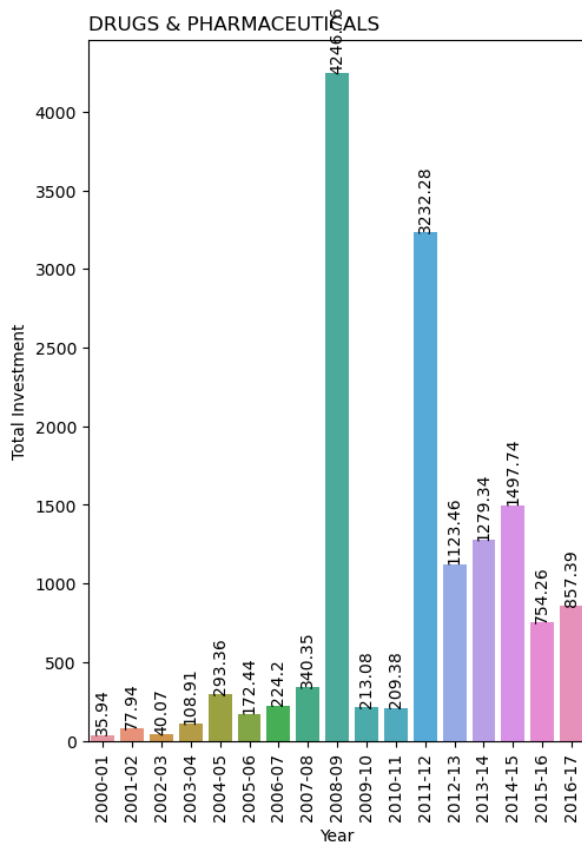
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```



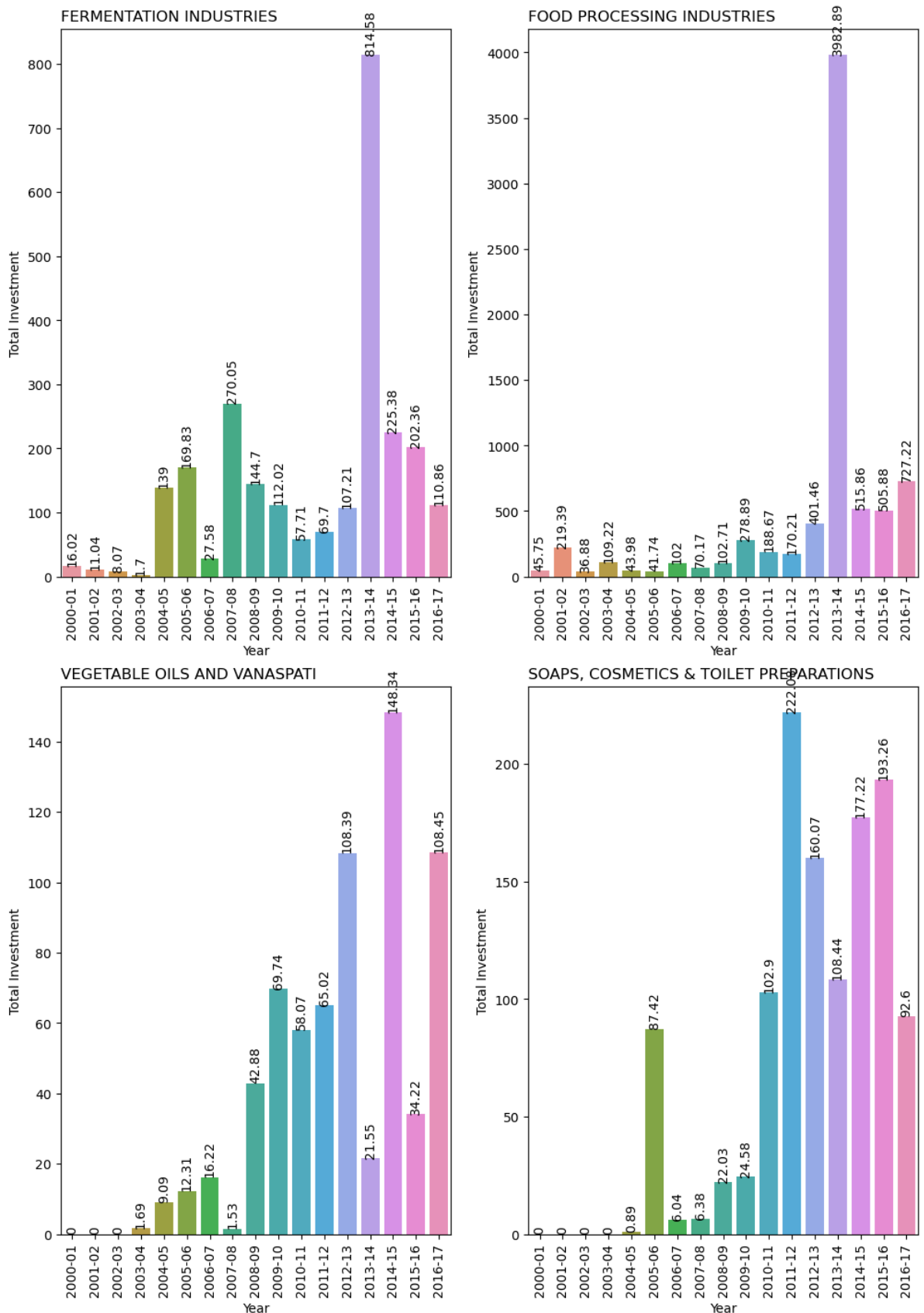
```
In [19]: i=36
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Investment')
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```



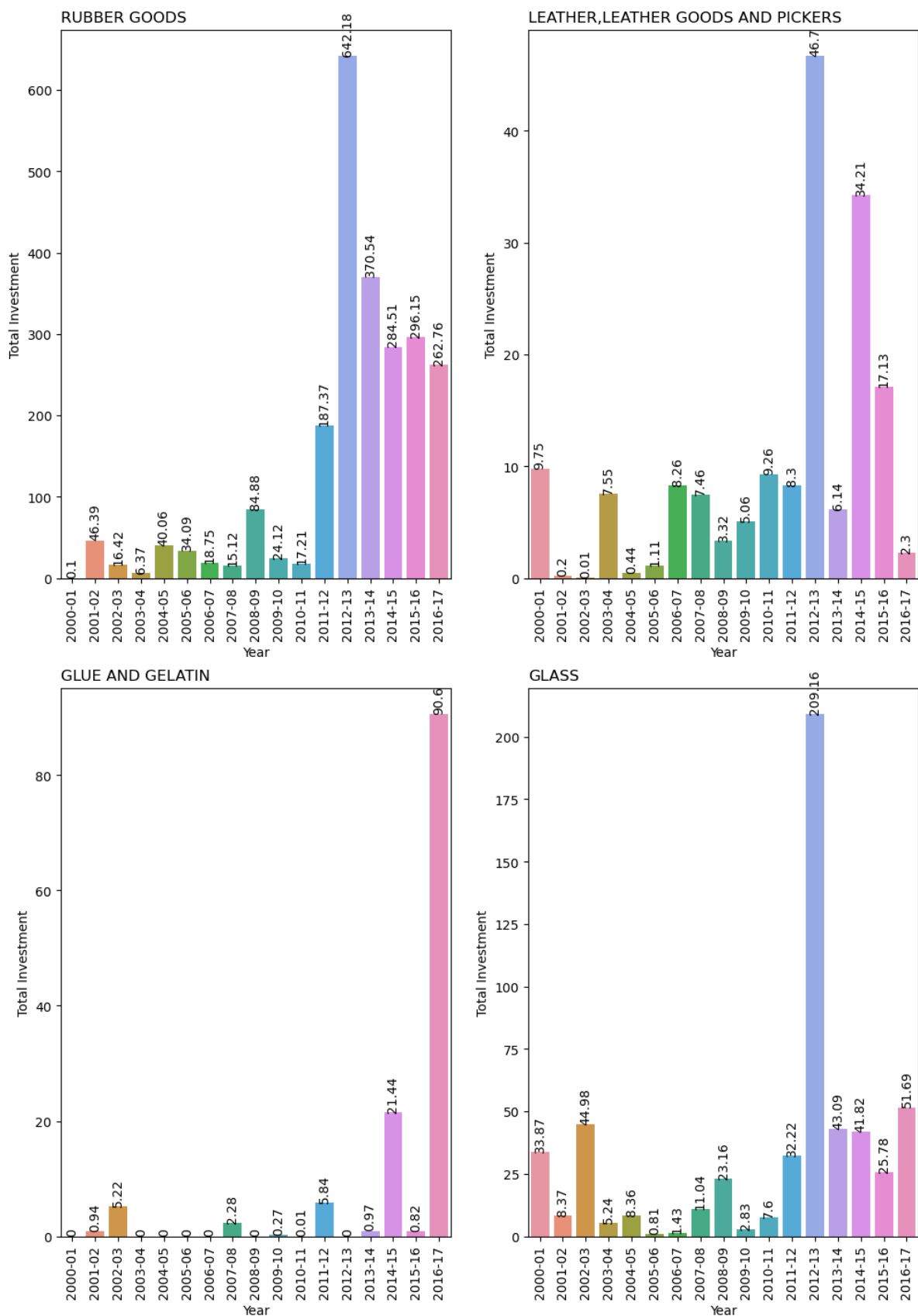
```
In [20]: i=40
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Investment')
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
for label in labels:
    label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```

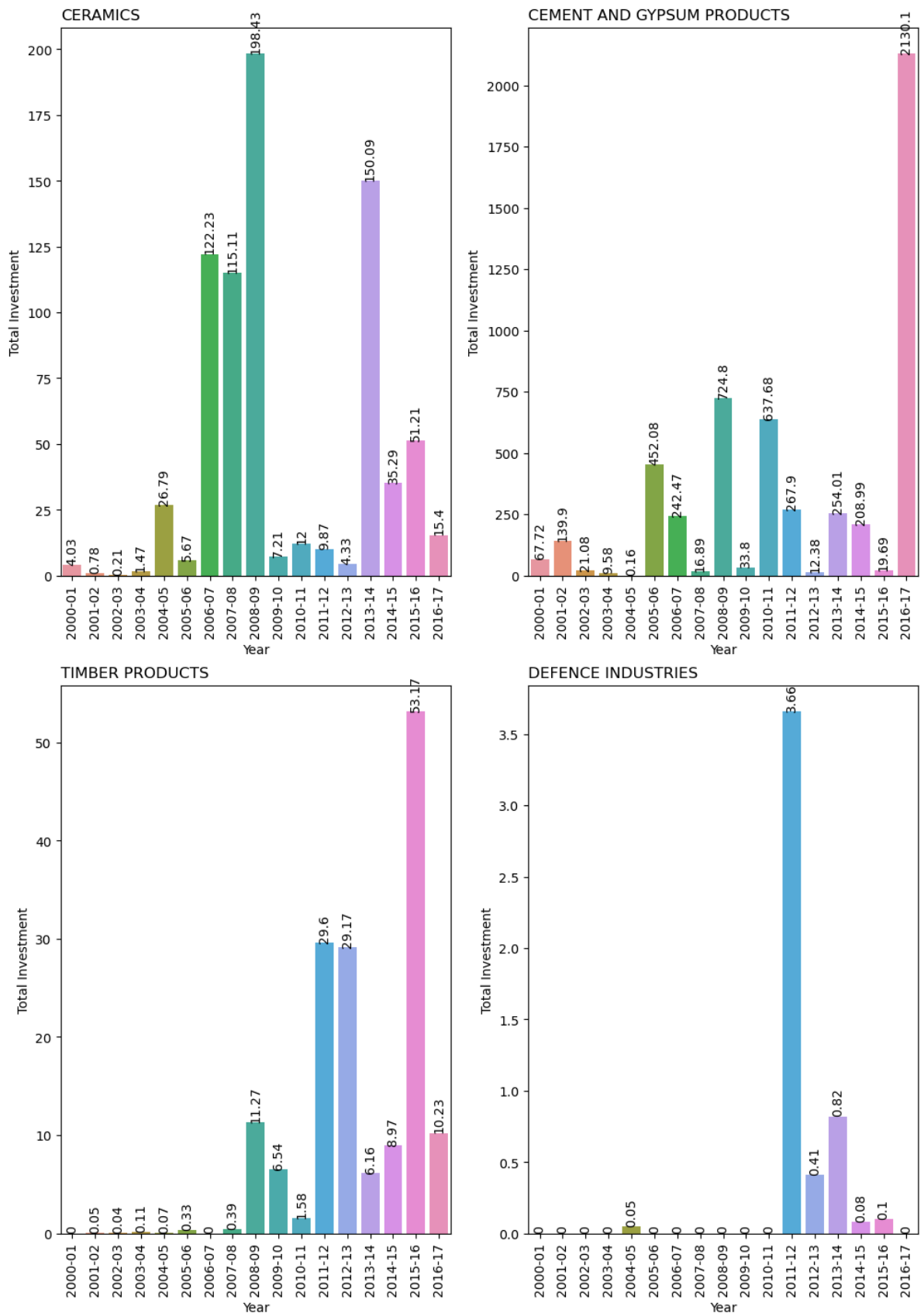
```
In [21]: i=44
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```



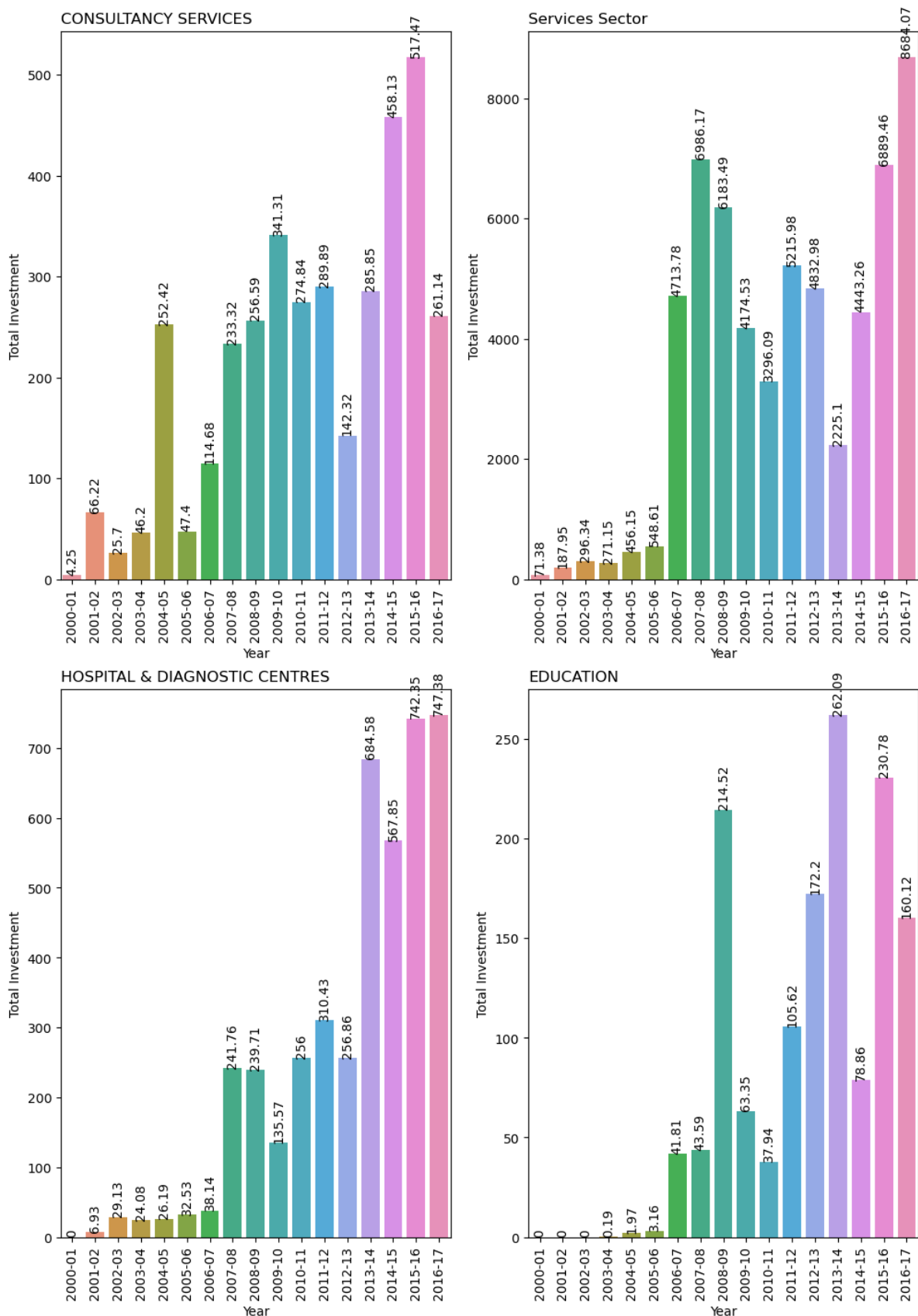
```
In [24]: i=48
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title('Services Sector',loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```



```

In [23]: i=52
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Investment')
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
for label in labels:
    label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')

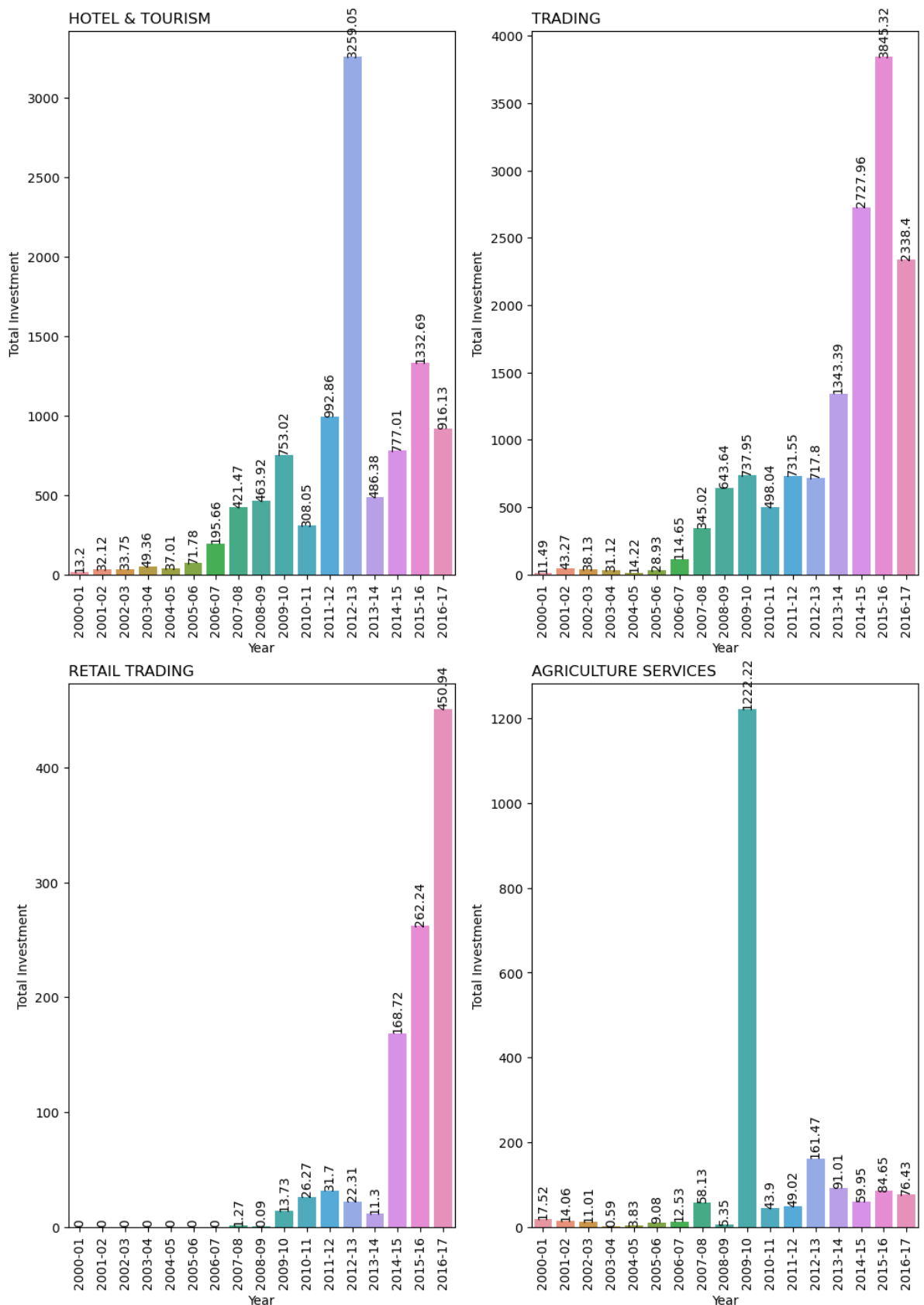
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title(sec[i+1],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```



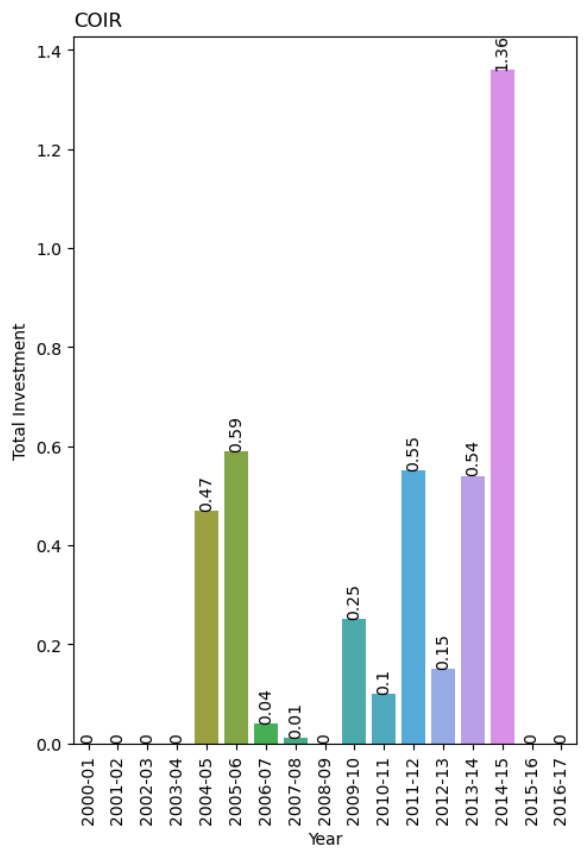
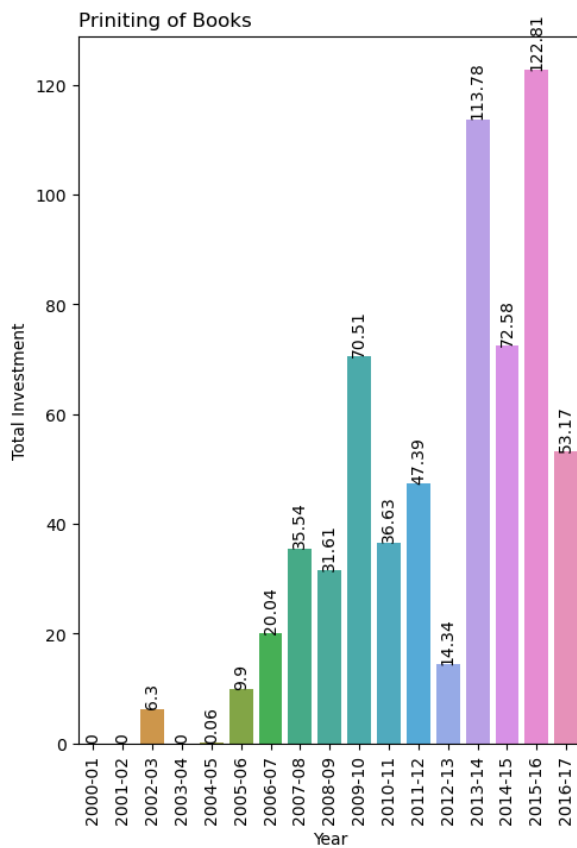
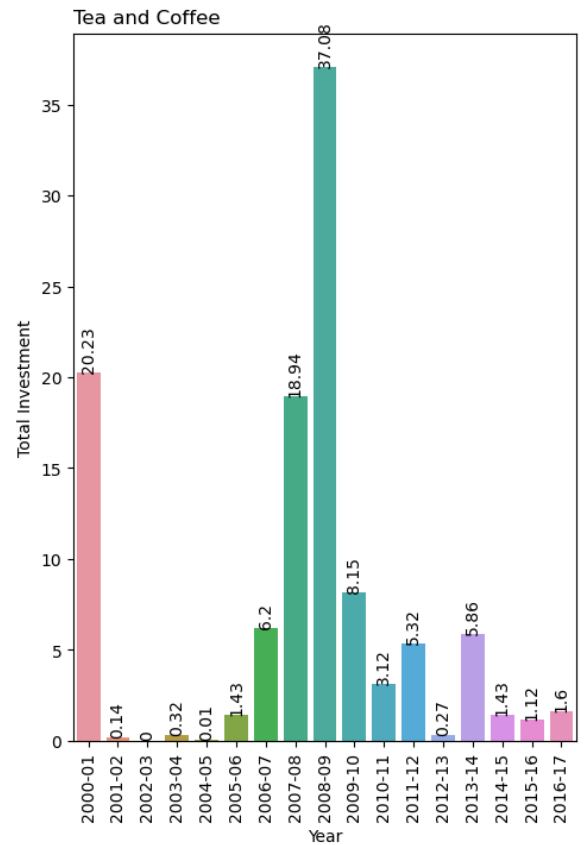
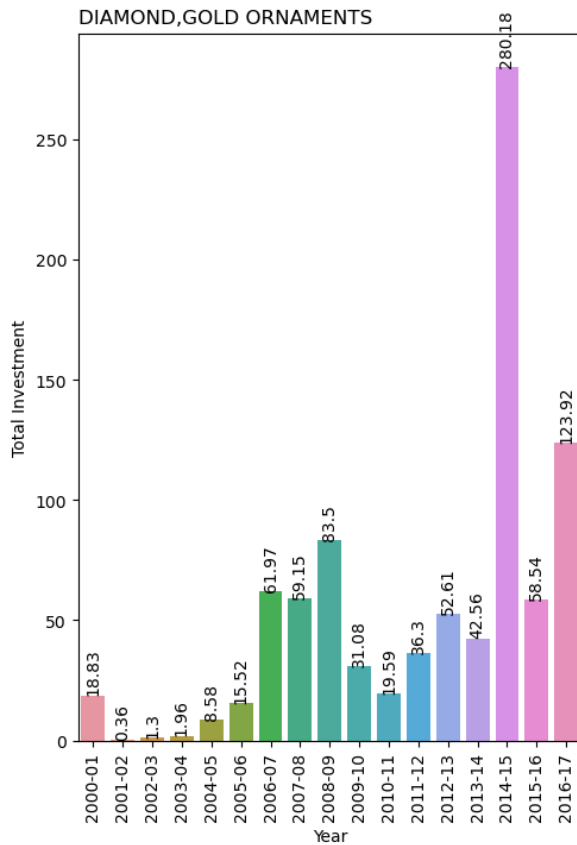
```
In [26]: i=56
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title('Tea and Coffee',loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title('Priniting of Books',loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+3]],x='Year',y='Total Ir
for container in ax4.containers:
    labels=ax4.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax4.tick_params(axis="x",rotation=90)
ax4.set_title(sec[i+3],loc='left')

plt.show()
```

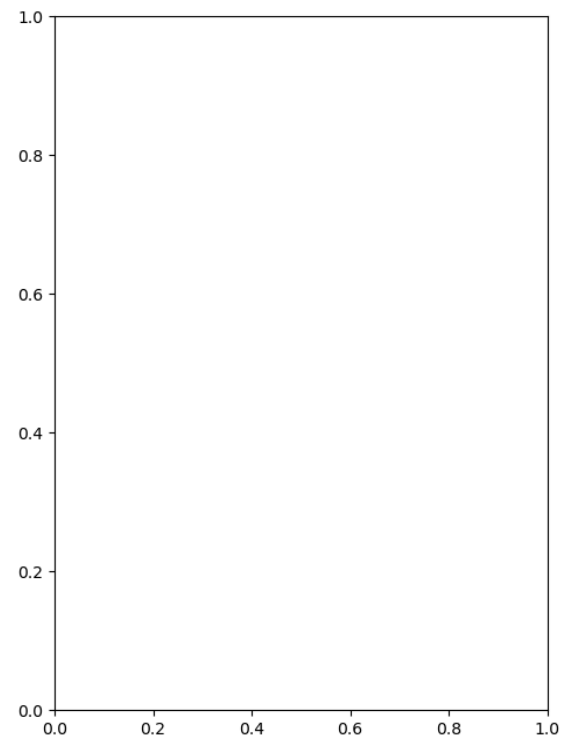
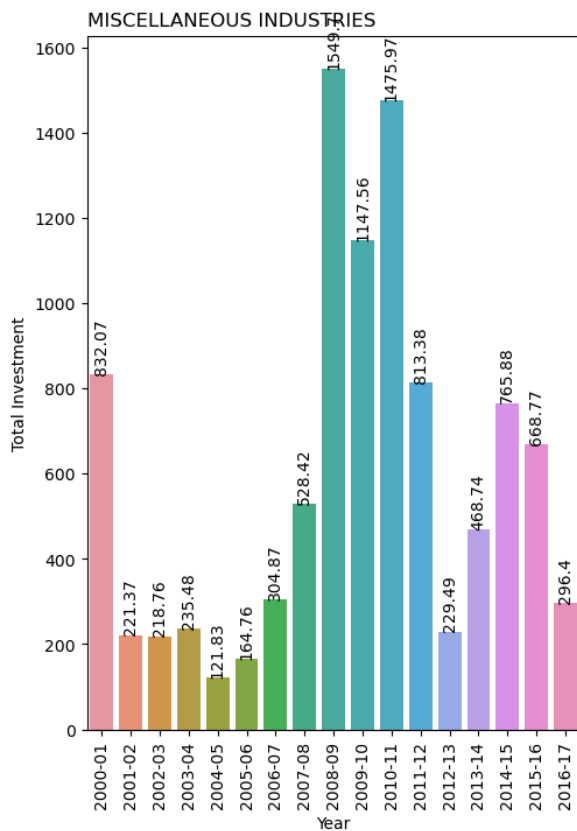
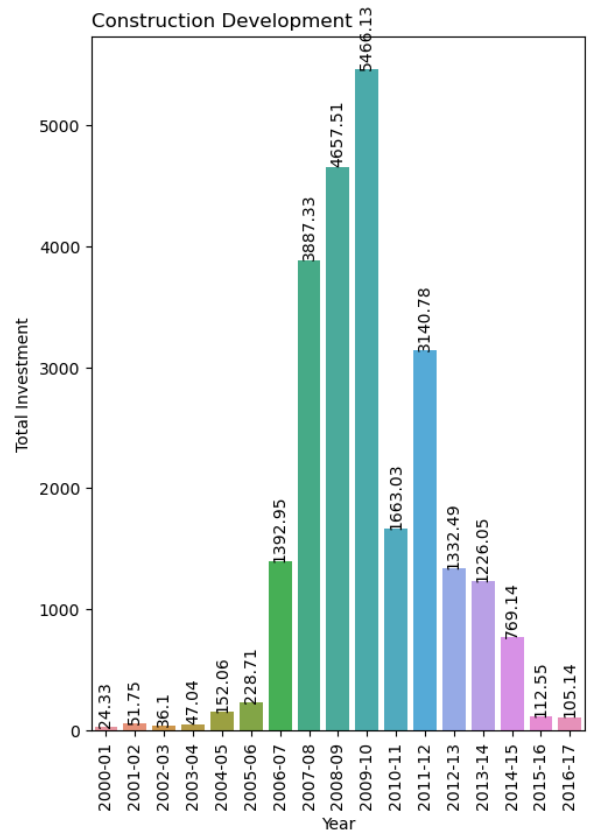
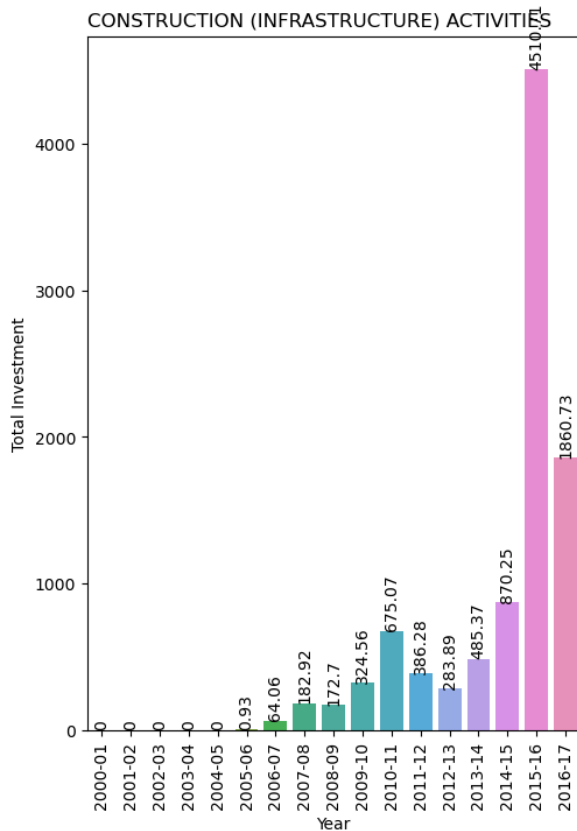



```
In [29]: i=60
fig,((ax1,ax2),(ax3,ax4))=plt.subplots(2,2,figsize=(12,17))
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i]],x='Year',y='Total Inve
for container in ax1.containers:
    labels=ax1.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax1.tick_params(axis="x",rotation=90)
ax1.set_title(sec[i],loc='left')
```

```
sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+1]],x='Year',y='Total Ir
for container in ax2.containers:
    labels=ax2.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax2.tick_params(axis="x",rotation=90)
ax2.set_title('Construction Development',loc='left')

sns.barplot(df_transformed[df_transformed["Sector"]==sec[i+2]],x='Year',y='Total Ir
for container in ax3.containers:
    labels=ax3.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
ax3.tick_params(axis="x",rotation=90)
ax3.set_title(sec[i+2],loc='left')

plt.show()
```



```
In [30]: df.head()
```

Out[30]:

	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08	2008-09	2009-10
0	METALLURGICAL INDUSTRIES	22.69	14.14	36.61	8.11	200.38	149.13	169.94	1175.75	959.94	419.88
1	MINING	1.32	6.52	10.06	23.48	9.92	7.40	6.62	444.36	34.16	174.40
2	POWER	89.42	757.44	59.11	27.09	43.37	72.69	157.15	988.68	907.66	1271.79
3	NON-CONVENTIONAL ENERGY	0.00	0.00	1.70	4.14	1.27	1.35	2.44	58.82	125.88	622.52
4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08	0.22	0.00

In [34]:

```
columns = [f"{year}-{str(year+1)[-2:]}" for year in range(2000, 2017)]
df['Total']=df[columns].sum(axis=1)
df.head()
```

Out[34]:

	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08	2008-09	2009-10
0	METALLURGICAL INDUSTRIES	22.69	14.14	36.61	8.11	200.38	149.13	169.94	1175.75	959.94	419.88
1	MINING	1.32	6.52	10.06	23.48	9.92	7.40	6.62	444.36	34.16	174.40
2	POWER	89.42	757.44	59.11	27.09	43.37	72.69	157.15	988.68	907.66	1271.79
3	NON-CONVENTIONAL ENERGY	0.00	0.00	1.70	4.14	1.27	1.35	2.44	58.82	125.88	622.52
4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08	0.22	0.00

In [63]:

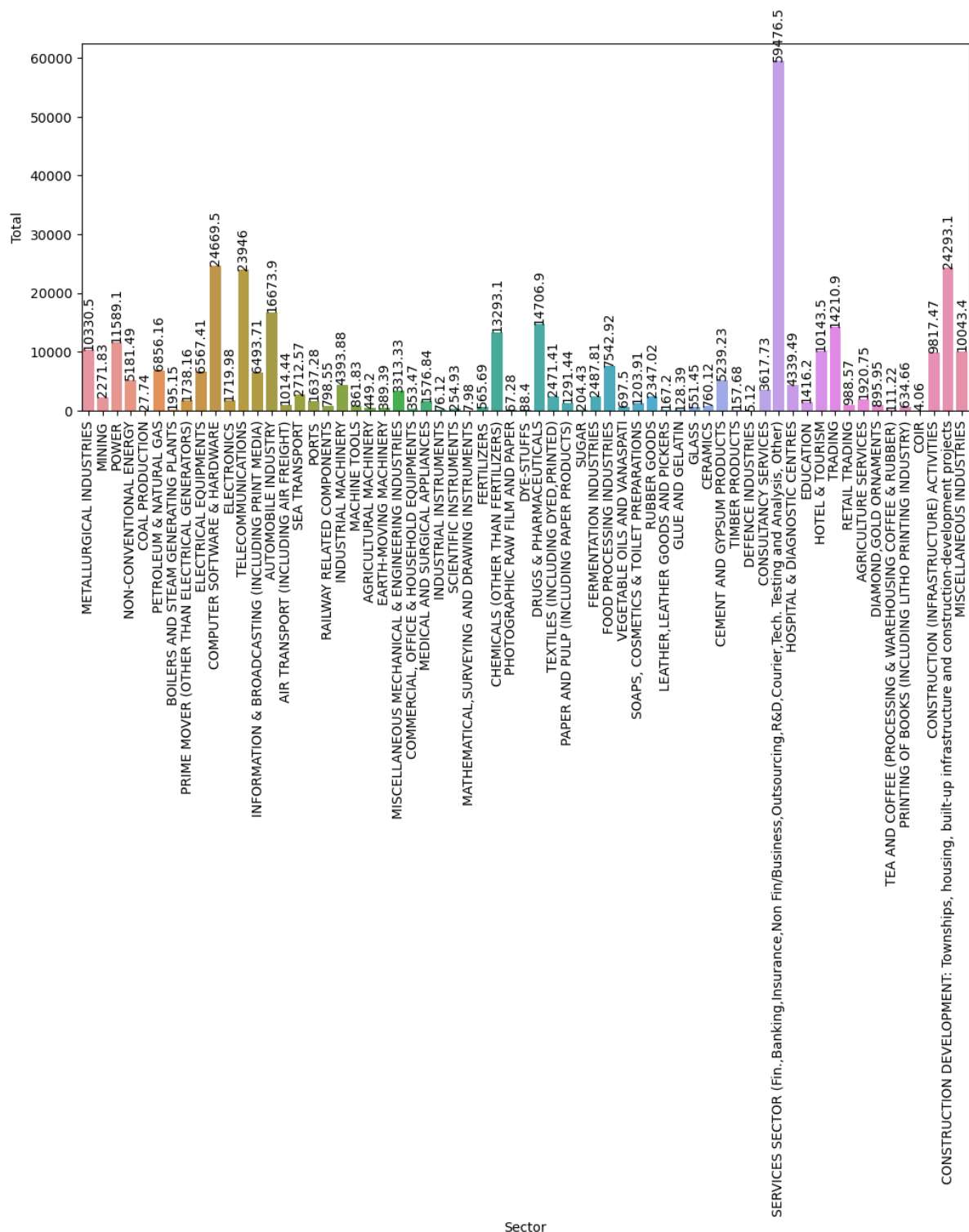
```
df[['Sector','Total']].sort_values('Total',ascending=False)
```

Out[63]:

	Sector	Total
49	SERVICES SECTOR (Fin.,Banking,Insurance,Non Fi...	59476.49
9	COMPUTER SOFTWARE & HARDWARE	24669.49
61	CONSTRUCTION DEVELOPMENT: Townships, housing, ...	24293.09
11	TELECOMMUNICATIONS	23946.01
13	AUTOMOBILE INDUSTRY	16673.92
...
30	PHOTOGRAPHIC RAW FILM AND PAPER	67.28
4	COAL PRODUCTION	27.74
27	MATHEMATICAL,SURVEYING AND DRAWING INSTRUMENTS	7.98
47	DEFENCE INDUSTRIES	5.12
59	COIR	4.06

63 rows × 2 columns

```
In [39]: plt.figure(figsize=(12,5))
sns.barplot(df,x="Sector",y="Total")
for container in plt.gca().containers:
    labels=plt.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
plt.tick_params(axis="x",rotation=90)
```



1.The highest investment has been made in the Services sector, which includes Finance, Banking, Insurance, Non-Financial Business, Outsourcing, R&D, Courier, Tech Testing and Analysis, and Other, amounting to ₹59,476.5.

In [41]: df

Out[41]:

	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08	2008-09
0	METALLURGICAL INDUSTRIES	22.69	14.14	36.61	8.11	200.38	149.13	169.94	1175.75	959.94
1	MINING	1.32	6.52	10.06	23.48	9.92	7.40	6.62	444.36	34.16
2	POWER	89.42	757.44	59.11	27.09	43.37	72.69	157.15	988.68	907.66
3	NON-CONVENTIONAL ENERGY	0.00	0.00	1.70	4.14	1.27	1.35	2.44	58.82	125.88
4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08	0.22
...
58	PRINTING OF BOOKS (INCLUDING LITHO PRINTING IN...	0.00	0.00	6.30	0.00	0.06	9.90	20.04	35.54	31.61
59	COIR	0.00	0.00	0.00	0.00	0.47	0.59	0.04	0.01	0.00
60	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	0.00	0.00	0.00	0.00	0.00	0.93	64.06	182.92	172.70
61	CONSTRUCTION DEVELOPMENT: Townships, housing, ...	24.33	51.75	36.10	47.04	152.06	228.71	1392.95	3887.33	4657.51
62	MISCELLANEOUS INDUSTRIES	832.07	221.37	218.76	235.48	121.83	164.76	304.87	528.42	1549.70

63 rows × 19 columns

```

In [55]: years = df.columns[1:]
cagr_values = [] # List to store calculated CAGR values

for index, row in df.iterrows():
    sector = row['Sector']
    beginning_value = row[years[0]]
    ending_value = row[years[-3]]
    n = len(years)-2

    # Check if beginning_value is zero, insert NaN if so
    if beginning_value == 0:
        cagr = np.nan # NaN indicates CAGR is not calculable
    else:
        cagr = (ending_value / beginning_value) ** (1/n) - 1

    cagr_values.append(cagr*100)

# Add CAGR values to a new column 'CAGR' in df1
df['CAGR'] = cagr_values

# Print or display the updated DataFrame with CAGR values
df

```

Out[55]:

	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08	2008-09
0	METALLURGICAL INDUSTRIES	22.69	14.14	36.61	8.11	200.38	149.13	169.94	1175.75	959.94
1	MINING	1.32	6.52	10.06	23.48	9.92	7.40	6.62	444.36	34.16
2	POWER	89.42	757.44	59.11	27.09	43.37	72.69	157.15	988.68	907.66
3	NON-CONVENTIONAL ENERGY	0.00	0.00	1.70	4.14	1.27	1.35	2.44	58.82	125.88
4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08	0.22
...
58	PRINTING OF BOOKS (INCLUDING LITHO PRINTING IN...	0.00	0.00	6.30	0.00	0.06	9.90	20.04	35.54	31.61
59	COIR	0.00	0.00	0.00	0.00	0.47	0.59	0.04	0.01	0.00
60	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	0.00	0.00	0.00	0.00	0.00	0.93	64.06	182.92	172.70
61	CONSTRUCTION DEVELOPMENT: Townships, housing, ...	24.33	51.75	36.10	47.04	152.06	228.71	1392.95	3887.33	4657.51
62	MISCELLANEOUS INDUSTRIES	832.07	221.37	218.76	235.48	121.83	164.76	304.87	528.42	1549.70

63 rows × 20 columns

```

In [57]: df.loc[df['Sector'] == 'NON-CONVENTIONAL ENERGY', 'CAGR'] = (((783.57/1.70)**(1/14))-1)*100
df.loc[df['Sector'] == 'CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES', 'CAGR'] = (((186.0/0.00)**(1/14))-1)*100
df.loc[df['Sector'] == 'PRINTING OF BOOKS (INCLUDING LITHO PRINTING INDUSTRY)', 'CAGR'] = (((450.94/1.27)**(1/9))-1)*100
df.loc[df['Sector'] == 'RETAIL TRADING', 'CAGR'] = (((450.94/1.27)**(1/9))-1)*100
df.loc[df['Sector'] == 'EDUCATION', 'CAGR'] = (((160.12/0.19)**(1/13))-1)*100
df.loc[df['Sector'] == 'HOSPITAL & DIAGNOSTIC CENTRES', 'CAGR'] = (((747.38/6.93)**(1/14))-1)*100
df.loc[df['Sector'] == 'SOAPS, COSMETICS & TOILET PREPARATIONS', 'CAGR'] = (((90.66/0.00)**(1/14))-1)*100
df.loc[df['Sector'] == 'VEGETABLE OILS AND VANASPATHI', 'CAGR'] = (((108.45/1.69)**(1/14))-1)*100
df.loc[df['Sector'] == 'RAILWAY RELATED COMPONENTS', 'CAGR'] = (((87.57/0.56)**(1/14))-1)*100
df.loc[df['Sector'] == 'PRIME MOVER (OTHER THAN ELECTRICAL GENERATORS)', 'CAGR'] =

```

```

In [59]: df.to_csv("UPDATED.CSV")

```

```

In [64]: df[['Sector', 'CAGR']].sort_values('CAGR', ascending=False)

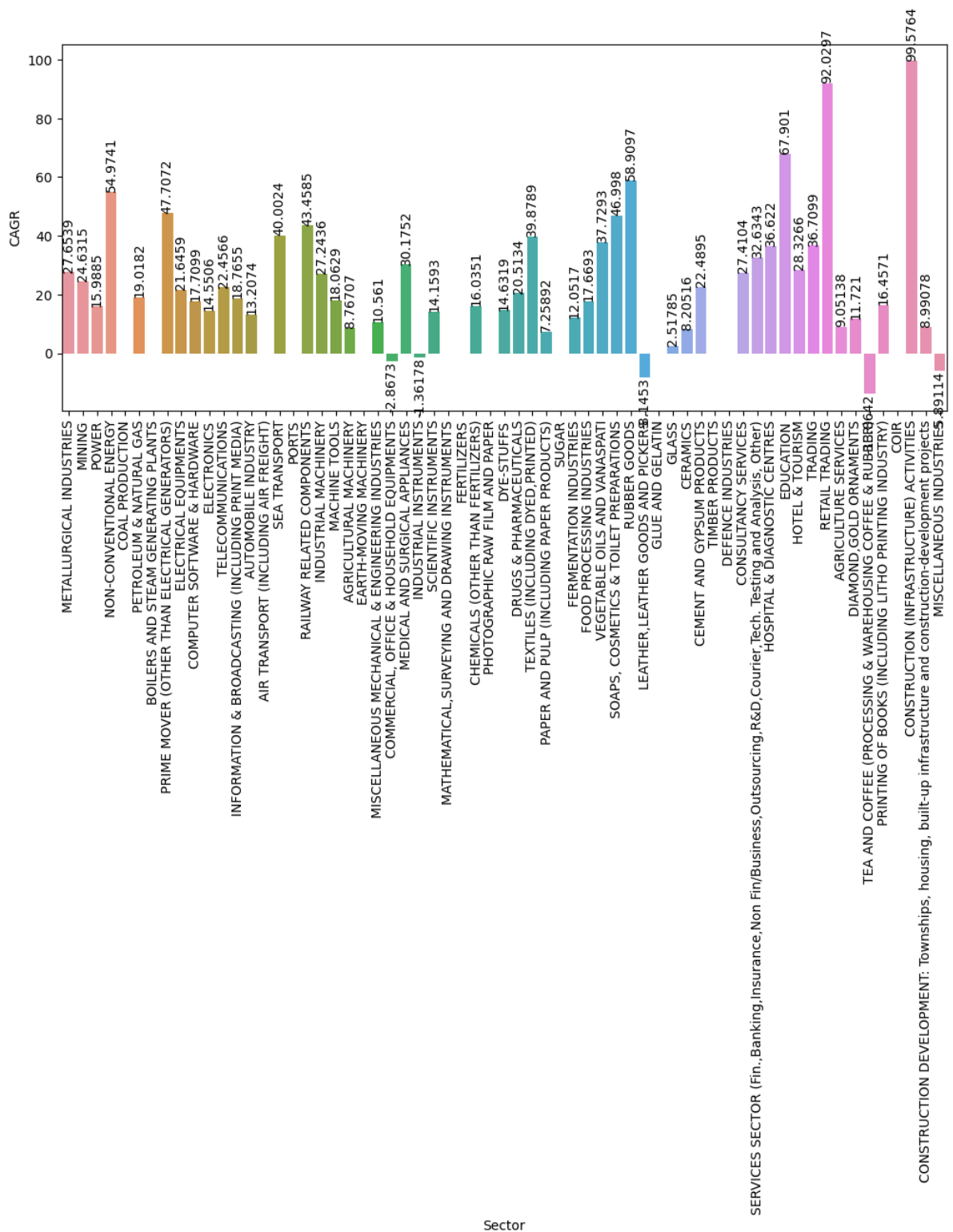
```


Out[64]:

	Sector	CAGR
60	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	99.576374
54	RETAIL TRADING	92.029671
51	EDUCATION	67.900979
40	RUBBER GOODS	58.909750
3	NON-CONVENTIONAL ENERGY	54.974130
...
35	SUGAR	NaN
42	GLUE AND GELATIN	NaN
46	TIMBER PRODUCTS	NaN
47	DEFENCE INDUSTRIES	NaN
59	COIR	NaN

63 rows × 2 columns

```
In [70]: plt.figure(figsize=(12,5))
sns.barplot(df,x="Sector",y="CAGR")
for container in plt.gca().containers:
    labels=plt.bar_label(container,color="black")
    for label in labels:
        label.set_rotation(90)
plt.tick_params(axis="x",rotation=90)
```



CONSTRUCTION ACTIVITIES HAS SHOWN MOST CAGR(Compound Annual Growth Rate) of 99.57%.

-COMMERCIAL

-OFFICE & HOUSEHOLD EQUIPMENTS

-INDUSTRIAL INSTRUMENTS

-LEATHER

-LEATHER GOODS AND PICKERS

-TEA AND COFFEE**-MISCELLANEOUS INDUSTRIES****-All thes sectors have shown a negative CAGR**

```
In [73]: list(df[df['CAGR']<0]['Sector'])
```

```
Out[73]: ['COMMERCIAL, OFFICE & HOUSEHOLD EQUIPMENTS',
          'INDUSTRIAL INSTRUMENTS',
          'LEATHER, LEATHER GOODS AND PICKERS',
          'TEA AND COFFEE (PROCESSING & WAREHOUSING COFFEE & RUBBER)',
          'MISCELLANEOUS INDUSTRIES']
```

```
In [2]: df2=pd.read_csv("UPDATED.CSV")
df2
```

```
Out[2]:
```

	Unnamed: 0	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08
0	0	METALLURGICAL INDUSTRIES	22.69	14.14	36.61	8.11	200.38	149.13	169.94	1175.75
1	1	MINING	1.32	6.52	10.06	23.48	9.92	7.40	6.62	444.36
2	2	POWER	89.42	757.44	59.11	27.09	43.37	72.69	157.15	988.68
3	3	NON-CONVENTIONAL ENERGY	0.00	0.00	1.70	4.14	1.27	1.35	2.44	58.82
4	4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08
...
58	58	PRINTING OF BOOKS (INCLUDING LITHO PRINTING IN...	0.00	0.00	6.30	0.00	0.06	9.90	20.04	35.54
59	59	COIR	0.00	0.00	0.00	0.00	0.47	0.59	0.04	0.01
60	60	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	0.00	0.00	0.00	0.00	0.00	0.93	64.06	182.92
61	61	CONSTRUCTION DEVELOPMENT: Townships, housing, ...	24.33	51.75	36.10	47.04	152.06	228.71	1392.95	3887.33
62	62	MISCELLANEOUS INDUSTRIES	832.07	221.37	218.76	235.48	121.83	164.76	304.87	528.42

63 rows × 21 columns

```
In [25]: percentage=df2[['Sector','Total']]
percentage['%']=(df2['Total']/df2['Total'].sum())*100
percentage
percentage=percentage.sort_values('%',ascending=False)

# pd.merge(df2['Sector'],percentage,left_index='index')
```

```
In [30]: percentage[:15]
```

```
Out[30]:
```

	Sector	Total	%
49	SERVICES SECTOR (Fin.,Banking,Insurance,Non Fi...	59476.49	17.915092
9	COMPUTER SOFTWARE & HARDWARE	24669.49	7.430771
61	CONSTRUCTION DEVELOPMENT: Townships, housing, ...	24293.09	7.317394
11	TELECOMMUNICATIONS	23946.01	7.212849
13	AUTOMOBILE INDUSTRY	16673.92	5.022401
32	DRUGS & PHARMACEUTICALS	14706.90	4.429909
53	TRADING	14210.88	4.280502
29	CHEMICALS (OTHER THAN FERTILIZERS)	13293.09	4.004051
2	POWER	11589.13	3.490797
0	METALLURGICAL INDUSTRIES	10330.54	3.111693
52	HOTEL & TOURISM	10143.46	3.055342
62	MISCELLANEOUS INDUSTRIES	10043.45	3.025218
60	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	9817.47	2.957150
37	FOOD PROCESSING INDUSTRIES	7542.92	2.272026
5	PETROLEUM & NATURAL GAS	6856.16	2.065164

OUT OF THE TOTAL INVESTMENT INFLOW NEARLY 17.9% HAS BEEN INVESTED IN SERVICES SCETOR, which includes Finance, Banking, Insurance, Non-Financial Business, Outsourcing, R&D, Courier, Tech Testing and Analysis

NEARLY 7.4% HAS BEEN INVESTED IN COMPUTER TECHNOLOGIES

```
In [29]: df2['Sector'].unique()
```

```
Out[29]: array(['METALLURGICAL INDUSTRIES', 'MINING', 'POWER',
'NON-CONVENTIONAL ENERGY', 'COAL PRODUCTION',
'PETROLEUM & NATURAL GAS', 'BOILERS AND STEAM GENERATING PLANTS',
'PRIME MOVER (OTHER THAN ELECTRICAL GENERATORS)',
'ELECTRICAL EQUIPMENTS', 'COMPUTER SOFTWARE & HARDWARE',
'ELECTRONICS', 'TELECOMMUNICATIONS',
'INFORMATION & BROADCASTING (INCLUDING PRINT MEDIA)',
'AUTOMOBILE INDUSTRY', 'AIR TRANSPORT (INCLUDING AIR FREIGHT)',
'SEA TRANSPORT', 'PORTS', 'RAILWAY RELATED COMPONENTS',
'INDUSTRIAL MACHINERY', 'MACHINE TOOLS', 'AGRICULTURAL MACHINERY',
'EARTH-MOVING MACHINERY',
'MISCELLANEOUS MECHANICAL & ENGINEERING INDUSTRIES',
'COMMERCIAL, OFFICE & HOUSEHOLD EQUIPMENTS',
'MEDICAL AND SURGICAL APPLIANCES', 'INDUSTRIAL INSTRUMENTS',
'SCIENTIFIC INSTRUMENTS',
'MATHEMATICAL,SURVEYING AND DRAWING INSTRUMENTS', 'FERTILIZERS',
'CHEMICALS (OTHER THAN FERTILIZERS)',
'PHOTOGRAPHIC RAW FILM AND PAPER', 'DYE-STUFFS',
'DRUGS & PHARMACEUTICALS', 'TEXTILES (INCLUDING DYED,PRINTED)',
'PAPER AND PULP (INCLUDING PAPER PRODUCTS)', 'SUGAR',
'FERMENTATION INDUSTRIES', 'FOOD PROCESSING INDUSTRIES',
'VEGETABLE OILS AND VANASPATI',
'SOAPS, COSMETICS & TOILET PREPARATIONS', 'RUBBER GOODS',
'LEATHER,LEATHER GOODS AND PICKERS', 'GLUE AND GELATIN', 'GLASS',
'CERAMICS', 'CEMENT AND GYPSUM PRODUCTS', 'TIMBER PRODUCTS',
'DEFENCE INDUSTRIES', 'CONSULTANCY SERVICES',
'SERVICES SECTOR (Fin.,Banking,Insurance,Non Fin/Business,Outsourcing,R&D,C
ourier,Tech. Testing and Analysis, Other)',
'HOSPITAL & DIAGNOSTIC CENTRES', 'EDUCATION', 'HOTEL & TOURISM',
'TRADING', 'RETAIL TRADING', 'AGRICULTURE SERVICES',
'DIAMOND,GOLD ORNAMENTS',
'TEA AND COFFEE (PROCESSING & WAREHOUSING COFFEE & RUBBER)',
'PRINTING OF BOOKS (INCLUDING LITHO PRINTING INDUSTRY)', 'COIR',
'CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES',
'CONSTRUCTION DEVELOPMENT: Townships, housing, built-up infrastructure and
construction-development projects',
'MISCELLANEOUS INDUSTRIES'], dtype=object)
```

```
In [4]: df3=pd.read_csv('UPDATED.csv')
df3.head()
```

```
Out[4]:
```

	Unnamed: 0	Sector	2000-01	2001-02	2002-03	2003-04	2004-05	2005-06	2006-07	2007-08	...
0	0	METALLURGICAL INDUSTRIES	22.69	14.14	36.61	8.11	200.38	149.13	169.94	1175.75	...
1	1	MINING	1.32	6.52	10.06	23.48	9.92	7.40	6.62	444.36	...
2	2	POWER	89.42	757.44	59.11	27.09	43.37	72.69	157.15	988.68	...
3	3	NON-CONVENTIONAL ENERGY	0.00	0.00	1.70	4.14	1.27	1.35	2.44	58.82	...
4	4	COAL PRODUCTION	0.00	0.00	0.00	0.04	0.00	9.14	1.30	14.08	...

5 rows × 21 columns

```
In [14]: df3[["Sector","CAGR"]].sort_values("CAGR",ascending=False)
```

Out[14]:

	Sector	CAGR
60	CONSTRUCTION (INFRASTRUCTURE) ACTIVITIES	99.576374
54	RETAIL TRADING	92.029671
51	EDUCATION	67.900979
40	RUBBER GOODS	58.909750
3	NON-CONVENTIONAL ENERGY	54.974130
...
35	SUGAR	NaN
42	GLUE AND GELATIN	NaN
46	TIMBER PRODUCTS	NaN
47	DEFENCE INDUSTRIES	NaN
59	COIR	NaN

63 rows × 2 columns

In []: