Analysis and interpretation of industry changes and the dynamics of different sectors in Singapore

Project proposal

1 Aims, Objectives and background

1.1 Introduction

In the constant ever changing landscape of Singapore's economic ecosystem, the relationship between industries and workforce dynamics play a pivotal role in ensuring the nation's prosperity. As Singapore continues to position itself as a global economic hub, the need for better insights into the employment system becomes crucial.There exists the potential to use a range of data gathered to bring a comprehensive understanding of how different industry changes within the job market.

I wanted to explore and gain better apprehension in the use of data science techniques to create metrics to measure data, grasp concepts of research into datasets for better analysis.

Having chanced upon datasets such as average monthly recruitment/resignation rates and quarterly job vacancy statistics, I was inspired to interpret patterns, perspectives and potential contributing factors on the state of the job market of Singapore.

1.2 Aims and Objectives

Within this project, I would like to explore the following:

- Examine the fluctuations in job vacancies in Singapore.
- Understand the transformations in various job industries in Singapore.
- Analyse the recuitment and resignation in job vacancies in Singapore.
- Determine the correlation between the changes in job resignations and shifts in career paths in Singapore.
- Evaluate whether the change in resignation in jobs is related to the change in career paths in Singapore.

For this project proposal, my aims are to:

1. Decide and determine if the data provided is sufficient to explore potential questions raised, all while managing the constraints of time and resources.

2. Decide which publications and articles should be used to allow for coherent analysis.

3. Clean and transform the data such that it is suitable for use in the techniques I would utilise later.

4. Carry out data analysis to identify the trends within the different data sets that ensures its viability for exploration.

1.3 Data

1.3.1 Data Requirements

I have taken three datasets,resignation and recruitment rates, Employment change, and job vacancies in Singapore as they complimen each other as can be grouped and monitored closer.

These data sets are 'Job Vacancy by Industry and Occupational Group', 'Employment Change By Industry' and 'Average Monthly Recruitment/Resignation Rates by Industry and Occupational Group' that have all been taken from the ministry of manpower Singapore. Leveraging these datasets are a strategic choice for research, offering a cohesive view of Singapore's labor market. Having sourced from a single, authoritative entity these datasets ensure methodological consistency and a reliable baseline for comparisons.

Maintaining such consistency is crucial to sidestepping potential pitfalls associated with using datasets from varied sources, such as divergent data collection methods or classification criteria. Grounding the research in Ministry of Manpower Singapore datasets mitigates these risks, allowing for a robust exploration of relationships between job vacancies, employment changes, and recruitment/resignation rates. The interconnected nature of these datasets facilitates a holistic understanding of labor market dynamics, promoting cross-referencing and validation of trends for a nuanced portrayal of Singapore's workforce landscape.

I intend to use exploratory analysis to examine the relationship between the three variables of interest, to understand potentially uncover hidden patterns within parts of the industry.

1.3.2 Choice of publications

Article sources and political editors chosen:

- The Straits Times Singapore is the most popular ans most read newspaper in Singapore, reaching around 1.3 million people everyday. It is generally right-center and publishes factual information that are trustworthy. Similarly, The buisness Times is a reputable financial newspaper in Singapore that is part of the SPH media group which also publishes The Straits Times.

- Gov.sg and Ministry of Trade and Industry Singapore are the official websites for the government of Singapore. Being an official government platform, it aims to provide accurate and unbiased information to the public. Although it does not scale it's popularity like the traditional news outlet, it serves as a direct channel for government news, and is generally right-center leaning.

- mycareersfuture is a government-supported job portal and career resource platform in Singapore. It is an official platform managed by Workforce Singapore (WSG) and the government. It is a reliable source for information related to workforce trends in Singapore.

- UNECE (United Nations Economic Commission for Europe) is an international organization, and its publications, including the one cited, focus on economic matters. While not a news outlet, it is a reputable source for research and data.

In order to examine different reasonings, it made more sense to have publications with different affiliations, reasonings and audience.

1.3.3 Choice of articles:

I have selected the most recent data that was given by the Ministry of Manpower Singapore on the basis that:

- It ensures a much seamless integration for analysis between multiple data sets.
- It ensures uniform data collection methods to minimise discrepancies.
- It mitigates the risk of inconsistencies or contradiction that may result in inaccurate conclusions.

1.3.4 Limitations and constraints of the data

Limited External Perspectives

- Reliance on a single source may limit exposure to different perspectives and alternative data collection methodologies.

- Utilising external organisations may provide a broader understanding of the labor market.

Bias and Subjectivity

- The data may be subjected to institutional bias which can potentially influence the presentation and interpretation of trends.

- The methods used to collect data can introduce bias.Survey methodologies, sampling techniques, or data collection instruments may favor certain groups or perspectives.

1.4 Ethical considerations

1.4.1: Use of open data

The data gathered is open government data provided by the Ministry of Manpower Singapore. It does not contain personally identifiable information and adheres to strict anonymity. Moreover, the provided information aligns with the principles outlined in the Privacy Policy and Terms of Use of Data.gov.sg, ensuring responsible and ethical utilization of open government data for analytical purposes.

1.4.2: Use of article texts for analysis

In conducting this analysis, I have selected and utilised reputable sources to ensure accuracy and reliability. The data from sources such as Business Times, Straits Times,and the UNECE were chosen for their established credibility and journalistic standards. Moreover, insights from government platforms like Gov.sg and and the Ministry of Trade and Industry were used to provide an official perspective on labor market statistics. The use of diverse sources, including academic reports like UNECE's analysis of employment quality during COVID-19, aimed to capture a more comprehensive view. Citations and references are also used to recognize the intellectual property of the authors and uphold ethical standards in referencing.

2 Webscraping articles

3 Data cleaning and processing

We first import the libraries and the datasets.

In [2]:
```
#Importing libraries and modules for this project

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

In [3]:
```
#loading of data sets
#sources of data set taken from data.gov.sg, an open data source.
resign = pd.read_csv("AverageMonthlyRecruitmentResignationRatebyIndustryandO
employChange = pd.read_csv("EmploymentChangeByIndustryQuarterly.csv")
jobVacancy = pd.read_csv("JobVacancybyIndustryOccupationalGroupLevel3.csv")
```

3.1.1 Cleaning data for Resign

In [4]:
```
resign.sample(3)
```

Out[4]:

	quarter	industry1	industry2	industry3	occupation1	recruitment_rate	resigr
10988	2018-Q4	services	community, social and personal services	other community, social and personal services	production and transport operators, cleaners a...	3.1	
6002	2009-Q3	manufacturing	transport equipment	transport equipment	professional, managers, executive and technicians	1.2	
104	1998-Q2	manufacturing	textile and wearing apparel	textile and wearing apparel	total	2.8	

From the values received in the columns, there are 3 types of industry depth. The first being the general industry, the second being a subset of industry1, and likewise third is a subset of industry2.

In [5]:
```
resignInd = resign['industry2']
resignWords = resignInd.tolist()
resignUniqueWords = set(resignWords)
print(len(resignUniqueWords))
print(resignUniqueWords)
```

```
36
{'machinery and equipment', 'business and real estate services', 'paper pro
ducts and publishing', 'electronic products', 'information and communicatio
ns', 'financial services', 'electrical products', 'others', 'administrative
and support services', 'professional services', 'other manufacturing indust
ries', 'textile and wearing apparel', 'fabricated metal products', 'financi
al intermediation', 'accommodation and food services', 'petroleum, chemical
and pharmaceutical products', 'wholesale and retail trade', 'financial and
insurance services', 'electronic, computer and optical products', 'paper,ru
bber,plastic products and printing', 'fabricated metal products, machinery
and equipment', 'real estate and leasing services', 'real estate services',
'community, social and personal services', 'transportation and storage', 'f
ood, beverages and tobacco', 'transport equipment', 'hotels and restaurant
s', 'paper products and printing', 'rubber and plastic products', 'petroleu
m and chemical products', 'transport and storage', 'medical and precision i
nstruments', 'transport, storage and communications', 'total', 'constructio
n'}
```

In [6]:
```
resignInd = resign['industry1']
resignWords = resignInd.tolist()
resignUniqueWords = set(resignWords)
print(len(resignUniqueWords))
print(resignUniqueWords)
```

```
5
{'others', 'services', 'manufacturing', 'total', 'construction'}
```

Industry 2 contains specific sectors that can be analysed.To better understand the general scope of the industries, we will use the broader umbrella, which is industry 1.

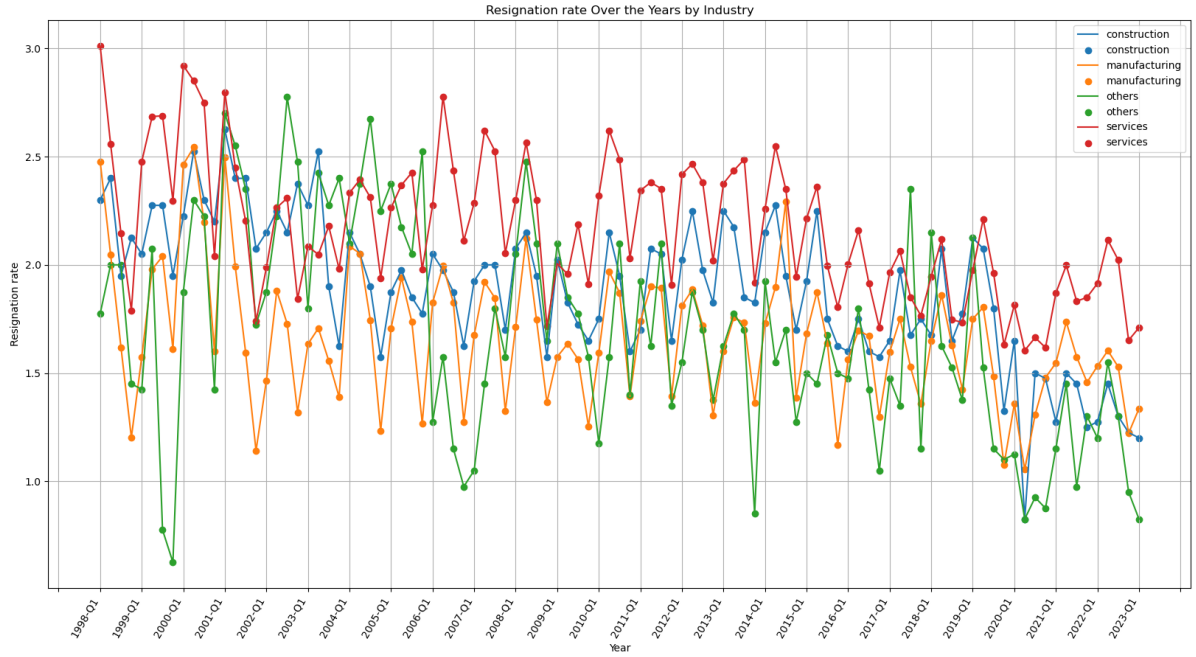
Now, we would like to know the types of industries within 'industry1'. Since there are 13235 different data values, manually filtering them out will not be efficient.

Since it contains both recruitment and resignation rates, we would need to filte both rates out so that we can use the datasets later.

In [7]:
```
# Read the CSV file into a DataFrame
resigndf = pd.read_csv('AverageMonthlyRecruitmentResignationRatebyIndustryar

resigndf['resignation_rate'] = pd.to_numeric(resigndf['resignation_rate'], e

# Group by 'industry1', 'quarter', and 'year'
grouped_data = resigndf.groupby(['industry1', 'quarter'])

# Calculate the average resignation rate for each group
average_resignation_rate = grouped_data['resignation_rate'].mean().reset_ind

# Print the resulting DataFrame
print(average_resignation_rate)

# Save the result to a new CSV file
average_resignation_rate.to_csv('average_resignation_rate.csv', index=False)
```

```
        industry1  quarter  resignation_rate
0    construction  1998-Q1             2.300
1    construction  1998-Q2             2.400
2    construction  1998-Q3             1.950
3    construction  1998-Q4             2.125
4    construction  1999-Q1             2.050
..            ...      ...               ...
500         total  2022-Q1             1.850
501         total  2022-Q2             1.975
502         total  2022-Q3             1.875
503         total  2022-Q4             1.550
504         total  2023-Q1             1.600

[505 rows x 3 columns]
```

In [36]:
```
#Plotting Resignation rate
resignC = pd.read_csv("average_resignation_rate.csv")

filtered_resign = resignC[resignC['industry1'] != 'total']

# Convert 'resignation_rate' column to numeric
filtered_resign.loc[:, 'resignation_rate'] = pd.to_numeric(filtered_resign[

# Plotting
fig, ax = plt.subplots(figsize=(20, 10))

# Iterate over unique values in 'industry1' and plot a line for each
for industry, group in filtered_resign.groupby('industry1'):
    ax.plot(group['quarter'], group['resignation_rate'], label=industry)
    ax.scatter(group['quarter'], group['resignation_rate'], label=industry)

# Customize the plot
ax.set_xlabel('Year')
ax.set_ylabel('Resignation rate')
ax.set_title('Resignation rate Over the Years by Industry')
ax.xaxis.set_major_locator(MultipleLocator(4))
ax.legend()

# Rotate x-axis labels by 60 degrees
plt.xticks(rotation=60, ha='right')

plt.grid(True)
plt.show()
```



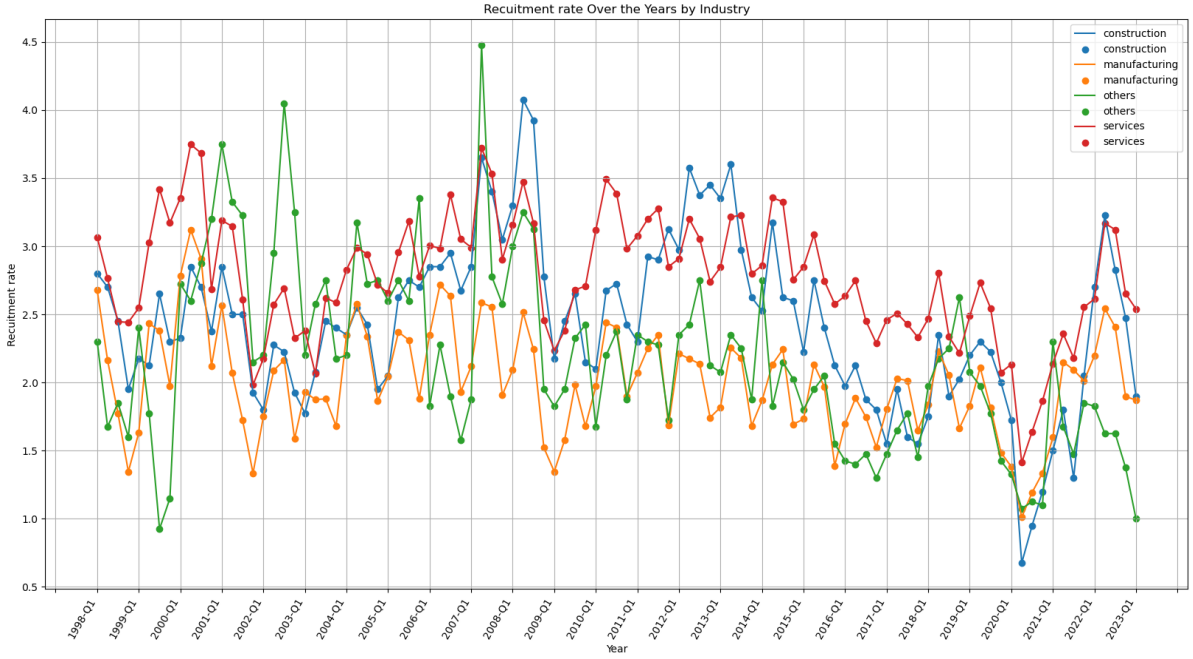
Likewise, we do the same for recuitment rates

In [9]:
```
# Read the CSV file into a DataFrame
recruitedf = pd.read_csv('AverageMonthlyRecruitmentResignationRatebyIndustry

recruitedf['recruitment_rate'] = pd.to_numeric(recruitedf['recruitment_rate

# Group by 'industry1', 'quarter', and 'year'
grouped_data = recruitedf.groupby(['industry1', 'quarter'])

# Calculate the average resignation rate for each group
average_recruitment_rate = grouped_data['recruitment_rate'].mean().reset_ind

# Print the resulting DataFrame
print(average_recruitment_rate)

# To save the result to a new CSV file
average_recruitment_rate.to_csv('average_recruitment_rate.csv', index=False)
```

```
        industry1  quarter  recruitment_rate
0    construction  1998-Q1             2.800
1    construction  1998-Q2             2.700
2    construction  1998-Q3             2.450
3    construction  1998-Q4             1.950
4    construction  1999-Q1             2.175
..            ...      ...               ...
500         total  2022-Q1             2.575
501         total  2022-Q2             3.075
502         total  2022-Q3             2.950
503         total  2022-Q4             2.550
504         total  2023-Q1             2.425

[505 rows x 3 columns]
```

In [39]:
```
#Plotting Resignation rate
recruiteC = pd.read_csv("average_recruitment_rate.csv")

filtered_recruite = recruiteC[recruiteC['industry1'] != 'total']

# Convert 'resignation_rate' column to numeric
filtered_recruite.loc[:, 'recruitment_rate'] = pd.to_numeric(filtered_recru
```

```
# Plotting
fig, ax = plt.subplots(figsize=(20, 10))

# Iterate over unique values in 'industry1' and plot a line for each
for industry, group in filtered_recruite.groupby('industry1'):
    ax.plot(group['quarter'], group['recruitment_rate'], label=industry)
    ax.scatter(group['quarter'], group['recruitment_rate'], label=industry)

# Customize the plot
ax.set_xlabel('Year')
ax.set_ylabel('Recuitment rate')
ax.set_title('Recuitment rate Over the Years by Industry')
ax.xaxis.set_major_locator(MultipleLocator(4))

ax.legend()

# Rotate x-axis labels by 90 degrees
plt.xticks(rotation=60, ha='right')

plt.grid(True)
plt.show()
```



3.1.2 Cleaning data for Employment change

In [11]:
```
employChange.sample(3)
```

Out[11]:

	quarter	industry1	industry2	industry3	employment_change
242	1993-Q3	others	others	others	0.2
3525	2021-Q4	manufacturing	paper / rubber / plastic products and printing	paper / rubber / plastic products and printing	0.1
368	1995-Q1	services	financial services	insurance	0.2

In [12]:
```
employInd = employChange['industry2']
employWords = employInd.tolist()
employUniqueWords = set(employWords)
```

```
print(len(employUniqueWords))
print(employUniqueWords)
```

```
32
{'machinery and equipment', 'business and real estate services', 'paper pro
ducts and publishing', 'electronic products', 'information and communicatio
ns', 'financial services', 'electrical products', 'paper / rubber / plastic
products and printing', 'others', 'administrative and support services', 'p
rofessional services', 'other manufacturing industries', 'textile and weari
ng apparel', 'fabricated metal products', 'accommodation and food service
s', 'petroleum, chemical and pharmaceutical products', 'wholesale and retai
l trade', 'financial and insurance services', 'electronic, computer and opt
ical products', 'fabricated metal products, machinery and equipment', 'real
estate and leasing services', 'real estate services', 'community, social an
d personal services', 'transportation and storage', 'food, beverages and to
bacco', 'transport equipment', 'hotels and restaurants', 'transport and sto
rage', 'medical and precision instruments', 'transport, storage and communi
cations', 'construction', 'petroleum, chemical, rubber and plastic product
s'}
```

In [13]:
```
employInd = employChange['industry1']
employWords = employInd.tolist()
employUniqueWords = set(employWords)
print(len(employUniqueWords))
print(employUniqueWords)
```

```
4
{'construction', 'services', 'others', 'manufacturing'}
```

Seen above, there are 4 subsectors within industry1. With the exclusion of 'total', we reduce the use of 'total' to prevent a lack of data.

Likewise, we would remove 'industry2' and 'industry3'.

In [14]:
```
employChangeC = employChange.copy()

employChange.drop(['industry2','industry3'], axis = 1)
```

Out[14]:

	quarter	industry1	employment_change
0	1991-Q1	manufacturing	0.1
1	1991-Q1	manufacturing	-1
2	1991-Q1	manufacturing	0.1
3	1991-Q1	manufacturing	0.3
4	1991-Q1	manufacturing	0.2
...	...	...	...
3679	2022-Q4	services	1.5
3680	2022-Q4	services	2.1
3681	2022-Q4	services	2.8
3682	2022-Q4	services	5.2
3683	2022-Q4	others	0.2

3684 rows × 3 columns

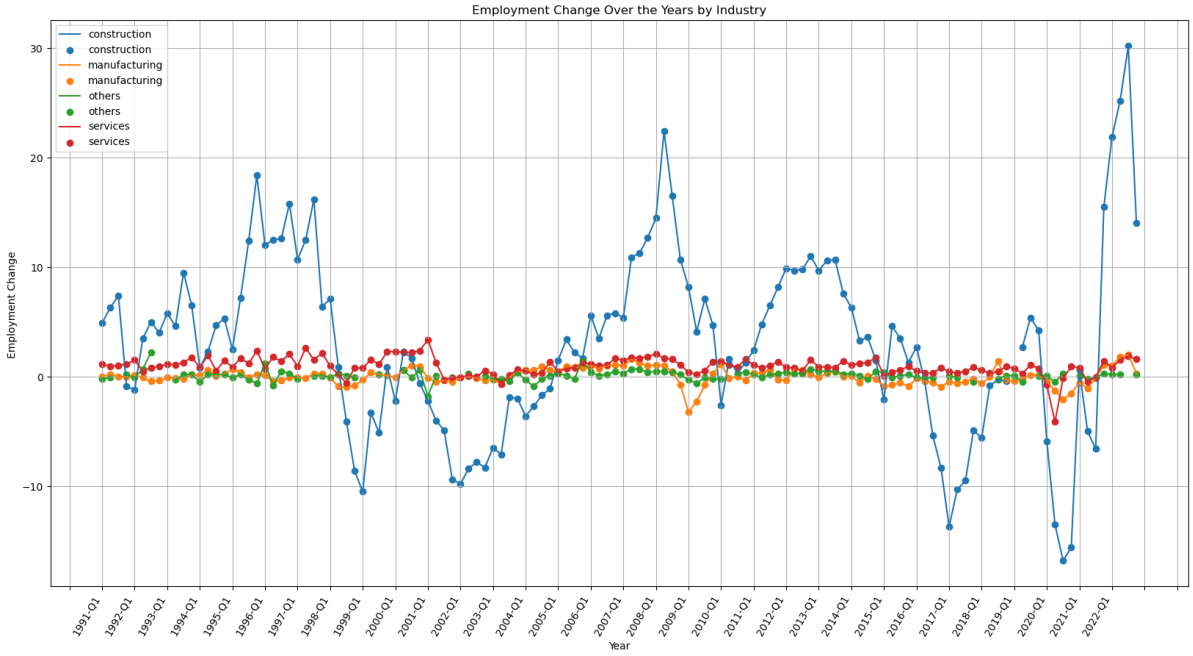
In [15]:
```
# Read the CSV file into a DataFrame
employdf = pd.read_csv('EmploymentChangeByIndustryQuarterly.csv')
```

```
employdf['employment_change'] = pd.to_numeric(employdf['employment_change'],

# Group by 'industry1', 'quarter', and 'year'
grouped_data = employdf.groupby(['industry1', 'quarter'])

# Calculate the average resignation rate for each group
employmentChange_rate = grouped_data['employment_change'].mean().reset_index

# Print the resulting DataFrame
print(employmentChange_rate)

# To save the result to a new CSV file
employmentChange_rate.to_csv('employmentChange_rate.csv', index=False)
```

```
        industry1  quarter  employment_change
0    construction  1991-Q1           4.900000
1    construction  1991-Q2           6.300000
2    construction  1991-Q3           7.400000
3    construction  1991-Q4          -0.900000
4    construction  1992-Q1          -1.200000
..            ...      ...                ...
507      services  2021-Q4           1.436364
508      services  2022-Q1           0.831818
509      services  2022-Q2           1.522727
510      services  2022-Q3           1.872727
511      services  2022-Q4           1.590000

[512 rows x 3 columns]
```

In [41]:

```
#Plotting employment change rate
employChangeC = pd.read_csv("employmentChange_rate.csv")

filtered_employ = employChangeC[employChangeC['industry1'] != 'total']

# Convert 'employment_change' column to numeric
filtered_employ.loc[:, 'employment_change'] = pd.to_numeric(filtered_employ

# Plotting
fig, ax = plt.subplots(figsize=(20, 10))

# Iterate over unique values in 'industry1' and plot a line for each
for industry, group in filtered_employ.groupby('industry1'):
    ax.plot(group['quarter'], group['employment_change'], label=industry)
    ax.scatter(group['quarter'], group['employment_change'], label=industry)

# Customize the plot
ax.set_xlabel('Year')
ax.set_ylabel('Employment Change ')
ax.set_title('Employment Change Over the Years by Industry')
ax.xaxis.set_major_locator(MultipleLocator(4))

ax.legend()

# Rotate x-axis labels
plt.xticks(rotation=60, ha='right')

plt.grid(True)
plt.show()
```



3.1.3 Cleaning data for Job vacancy by industry

We repeat the same data cleaning techniques for job Vacancy

In [17]:
```
jobVacancy.sample(3)
```

Out[17]:

	year	industry1	industry2	industry3	occupation1	job_vacancy
1266	2007	services	professional services	other professional services	professional, managers, executive and technicians	500
2356	2016	total	total	total	clerical, sales and services workers	13700
527	2002	services	wholesale and retail trade	wholesale trade	clerical, sales and services workers	300

In [18]:
```
vacancyInd = jobVacancy['industry2']
vacancyWords = vacancyInd.tolist()
vacancyUniqueWords = set(vacancyWords)
print(len(vacancyUniqueWords))
print(vacancyUniqueWords)
```

```
36
{'machinery and equipment', 'business and real estate services', 'paper pro
ducts and publishing', 'electronic products', 'information and communicatio
ns', 'financial services', 'electrical products', 'others', 'administrative
and support services', 'professional services', 'other manufacturing indust
ries', 'textile and wearing apparel', 'fabricated metal products', 'financi
al intermediation', 'accommodation and food services', 'petroleum, chemical
and pharmaceutical products', 'wholesale and retail trade', 'financial and
insurance services', 'electronic, computer and optical products', 'paper,ru
bber,plastic products and printing', 'fabricated metal products, machinery
and equipment', 'real estate and leasing services', 'real estate services',
'community, social and personal services', 'transportation and storage', 'f
ood, beverages and tobacco', 'transport equipment', 'hotels and restaurant
s', 'paper products and printing', 'rubber and plastic products', 'petroleu
m and chemical products', 'transport and storage', 'medical and precision i
nstruments', 'transport, storage and communications', 'total', 'constructio
n'}
```

In [19]:
```
vacancyInd1 = jobVacancy['industry1']
vacancyWords1 = vacancyInd1.tolist()
vacancyUniqueWords1 = set(vacancyWords1)
print(len(vacancyUniqueWords1))
print(vacancyUniqueWords1)
```

```
5
{'others', 'services', 'manufacturing', 'total', 'construction'}
```

In [20]:
```
jobVacancyC = jobVacancy.copy()

jobVacancyC.drop(['industry2','industry3', 'occupation1'], axis = 1)
```

Out[20]:

	year	industry1	job_vacancy
0	1998	total	18300
1	1998	total	4900
2	1998	total	6000
3	1998	total	7400
4	1998	manufacturing	200
...	...	...	...
3271	2022	services	300
3272	2022	others	700
3273	2022	others	400
3274	2022	others	-
3275	2022	others	200

3276 rows × 3 columns

In [21]:
```
#To compare the differences, vacancy and resign has 36 values while employ c
setVacancy = set(vacancyUniqueWords)
setResign = set(resignUniqueWords)
setChange = set(employUniqueWords)

#find missing values in Vacancy compared to Resign
setVtoR = setResign.difference(setVacancy)
print("Missing values in Vacancy compared to Resignation: ", setVtoR )

#find missing values in Resign compared to Vacancy
setRtoV = setVacancy.difference(setResign)
print("Missing values in Resignation compared to Vacancy: ", setRtoV )
print()

#no difference between Vacancy and Resign


#find missing values in Employment change compared to Vacancy
setCtoV = setVacancy.difference(setChange)
print("Missing values in employ Change compared to Vacancy: ", setCtoV)

print()
#find missing values in Vacancy compared to Employment Change
setVtoC = setChange.difference(setVacancy)
print("Missing values in Vacancy compared to employ Change: ", setVtoC)
print()
```

```
#Since there is no difference between  Vacancy and Resign, the difference be
```

```
Missing values in Vacancy compared to Resignation:  {'services', 'manufactu
ring'}
Missing values in Resignation compared to Vacancy:  {'machinery and equipme
nt', 'business and real estate services', 'paper products and publishing',
'electronic products', 'information and communications', 'financial service
s', 'electrical products', 'administrative and support services', 'professi
onal services', 'other manufacturing industries', 'textile and wearing appa
rel', 'fabricated metal products', 'financial intermediation', 'accommodati
on and food services', 'petroleum, chemical and pharmaceutical products',
'wholesale and retail trade', 'financial and insurance services', 'electron
ic, computer and optical products', 'paper,rubber,plastic products and prin
ting', 'fabricated metal products, machinery and equipment', 'real estate a
nd leasing services', 'real estate services', 'community, social and person
al services', 'transportation and storage', 'food, beverages and tobacco',
'transport equipment', 'hotels and restaurants', 'paper products and printi
ng', 'rubber and plastic products', 'petroleum and chemical products', 'tra
nsport and storage', 'medical and precision instruments', 'transport, stora
ge and communications'}

Missing values in employ Change compared to Vacancy:  {'machinery and equip
ment', 'business and real estate services', 'paper products and publishin
g', 'electronic products', 'information and communications', 'financial ser
vices', 'electrical products', 'administrative and support services', 'prof
essional services', 'other manufacturing industries', 'textile and wearing
apparel', 'fabricated metal products', 'financial intermediation', 'accommo
dation and food services', 'petroleum, chemical and pharmaceutical product
s', 'wholesale and retail trade', 'financial and insurance services', 'elec
tronic, computer and optical products', 'paper,rubber,plastic products and
printing', 'fabricated metal products, machinery and equipment', 'real esta
te and leasing services', 'real estate services', 'community, social and pe
rsonal services', 'transportation and storage', 'food, beverages and tobacc
o', 'transport equipment', 'hotels and restaurants', 'paper products and pr
inting', 'rubber and plastic products', 'petroleum and chemical products',
'transport and storage', 'medical and precision instruments', 'transport, s
torage and communications', 'total'}

Missing values in Vacancy compared to employ Change:  {'services', 'manufac
turing'}
```

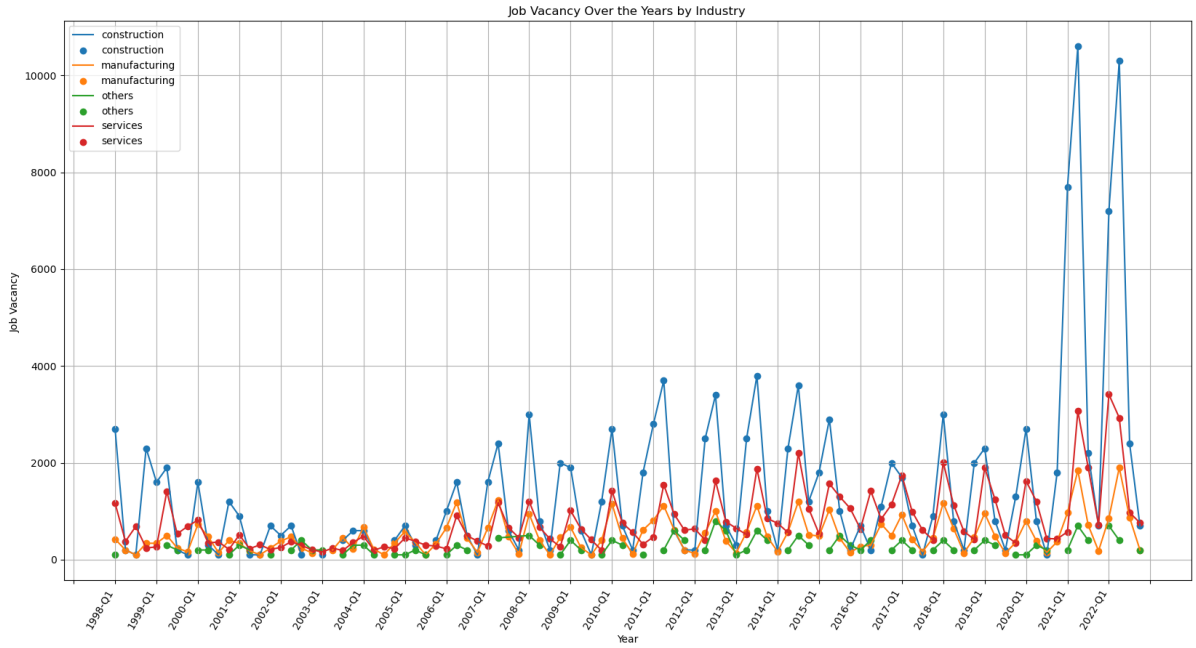
In [22]:
```
#adding a new column to the csv file so that there will be quarters involved

# Create a new column 'quarter'
jobVacancyC['quarter'] = ''

# Define a function to determine the quarter based on the row index
def determine_quarter(index):
    quarters = ['Q1', 'Q2', 'Q3', 'Q4']
    return quarters[index % len(quarters)]

def determine_yearQuarter(index):
    year_quarters = [index ]

# Iterate through rows and assign quarters
for i in range(len(jobVacancyC)):
    jobVacancyC.at[i, 'quarter'] = determine_quarter(i)

# Save the modified DataFrame to a new CSV file
jobVacancyC.to_csv('jobVacancyQuarter.csv', index=False)
```

In [23]:
```
# Read the CSV file into a DataFrame
vacancydf = pd.read_csv('jobVacancyQuarter.csv')

# Convert the 'job_vacancy' column to numeric, replacing non-numeric values
vacancydf['job_vacancy'] = pd.to_numeric(vacancydf['job_vacancy'], errors='

# Create a 'year_quarter' column
vacancydf['year_quarter'] = vacancydf['year'].astype(str) + '-Q' + vacancyd

# Group by 'industry1' and 'year_quarter'
grouped_data = vacancydf.groupby(['industry1', 'year_quarter'])

# Calculate the average job vacancy for each group
jobVacancy_rate = grouped_data['job_vacancy'].mean().reset_index()

# Print the resulting DataFrame
print(jobVacancy_rate)

# Save the result to a new CSV file
jobVacancy_rate.to_csv('jobVacancy_rate.csv', index=False)
```

```
        industry1 year_quarter  job_vacancy
0    construction      1998-Q1       2700.0
1    construction      1998-Q2        200.0
2    construction      1998-Q3        100.0
3    construction      1998-Q4       2300.0
4    construction      1999-Q1       1600.0
..            ...          ...          ...
492         total      2021-Q4      17100.0
493         total      2022-Q1      27000.0
494         total      2022-Q2     115000.0
495         total      2022-Q3      63700.0
496         total      2022-Q4      24300.0

[497 rows x 3 columns]
```

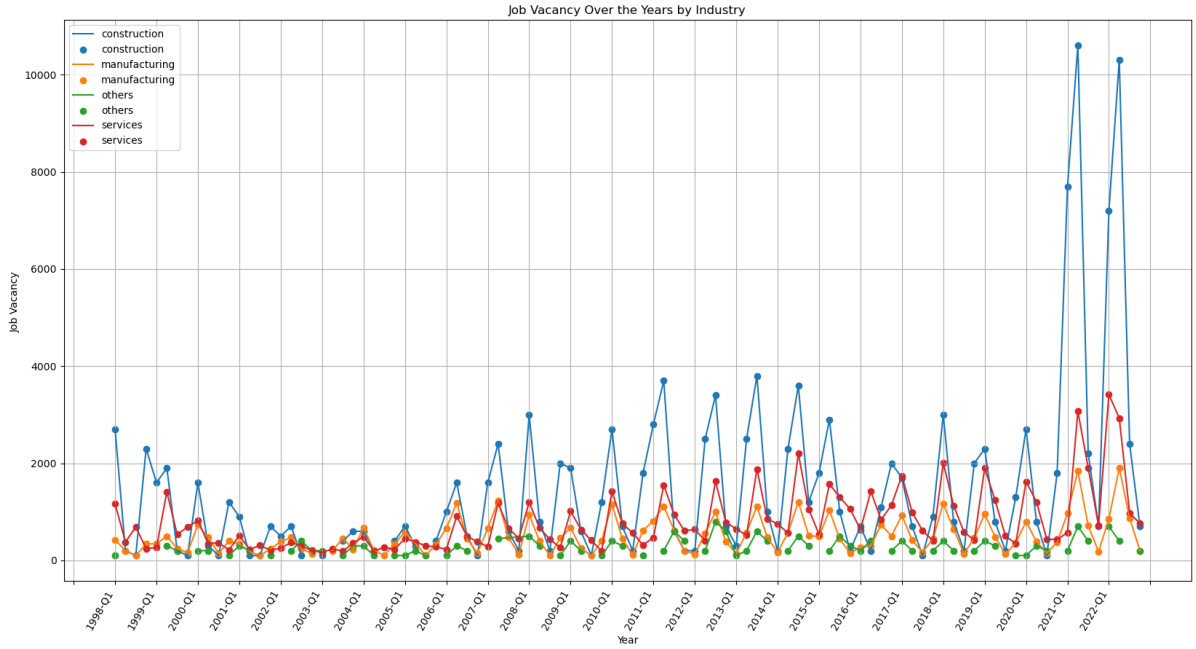
In [42]:
```
#Plotting employment change rate
jobVacancyC = pd.read_csv("jobVacancy_rate.csv")

filtered_job = jobVacancyC[jobVacancyC['industry1'] != 'total']

# Convert 'employment_change' column to numeric
filtered_job.loc[:, 'job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy

# Plotting
fig, ax = plt.subplots(figsize=(20, 10))

# Iterate over unique values in 'industry1' and plot a line for each
for industry, group in filtered_job.groupby('industry1'):
    ax.plot(group['year_quarter'], group['job_vacancy'], label=industry)
    ax.scatter(group['year_quarter'], group['job_vacancy'], label=industry)

# Customize the plot
ax.set_xlabel('Year')
ax.set_ylabel('Job Vacancy ')
ax.set_title('Job Vacancy Over the Years by Industry')
ax.xaxis.set_major_locator(MultipleLocator(4))

ax.legend()

# Rotate x-axis labels
plt.xticks(rotation=60, ha='right')
```

```
plt.grid(True)
plt.show()
```



In [43]:

```
jobVacancyC = pd.read_csv("jobVacancy_rate.csv")

filtered_job = jobVacancyC[jobVacancyC['industry1'] != 'total']

# Convert 'employment_change' column to numeric
filtered_job.loc[:, 'job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy

# Plotting
fig, ax = plt.subplots(figsize=(20, 10))

# Iterate over unique values in 'industry1' and plot a line for each
for industry, group in filtered_job.groupby('industry1'):
    ax.plot(group['year_quarter'], group['job_vacancy'], label=industry)
    ax.scatter(group['year_quarter'], group['job_vacancy'], label=industry)

# Customize the plot
ax.set_xlabel('Year')
ax.set_ylabel('Job Vacancy ')
ax.set_title('Job Vacancy Over the Years by Industry')
ax.xaxis.set_major_locator(MultipleLocator(4))

ax.legend()

# Rotate x-axis labels
plt.xticks(rotation=60, ha='right')

plt.grid(True)
plt.show()
```



4 Generating features

We generate features by analysing the changes in different industries. analysing different sectors one at a time.

We first compare the changes within the construction sector. since the y axis on both sides are different, we would need to plot the graph on a dual axis

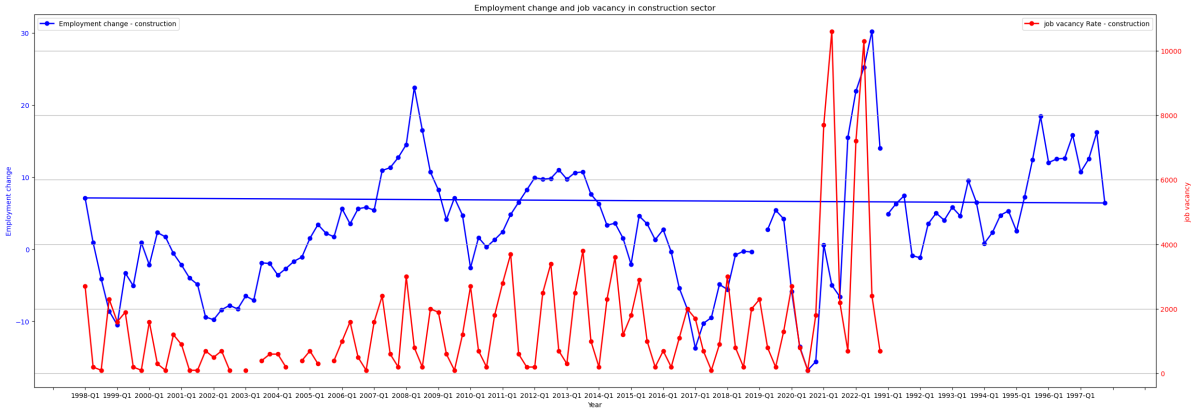
4.1 Analysing The construction sector

In [26]:
```
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib.ticker import MultipleLocator

#Loading of csv file
employChangeC = pd.read_csv("employmentChange_rate.csv")
jobVacancyC = pd.read_csv("jobVacancy_rate.csv")
resignC = pd.read_csv("average_resignation_rate.csv")
recruiteC = pd.read_csv("average_recruitment_rate.csv")
```

Construction Sector: Comparing the Employment change with the Job vacancy rate

In [27]:
```
# Load data for Employment change
filtered_employ = employChangeC[(employChangeC['industry1'] != 'total') & (e
filtered_employ['employment_change'] = pd.to_numeric(filtered_employ['employ

### Load data for Job Vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], er

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_employ.groupby('industry1'):
```

```
    # Plot job vacancy on the first y-axis
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for Employment Change
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for Job Vacancy rate
ax2 = ax1.twinx()
for industry, group in filtered_job.groupby('industry1'):
    # Plot Job vacancy rate on the second y-axis
    ax2.plot(group['year_quarter'], group['job_vacancy'], label=f'job vacanc

# Customize the plot for Job vacancy rate
ax2.set_ylabel('job vacancy ', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Employment change and job vacancy in construction sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```

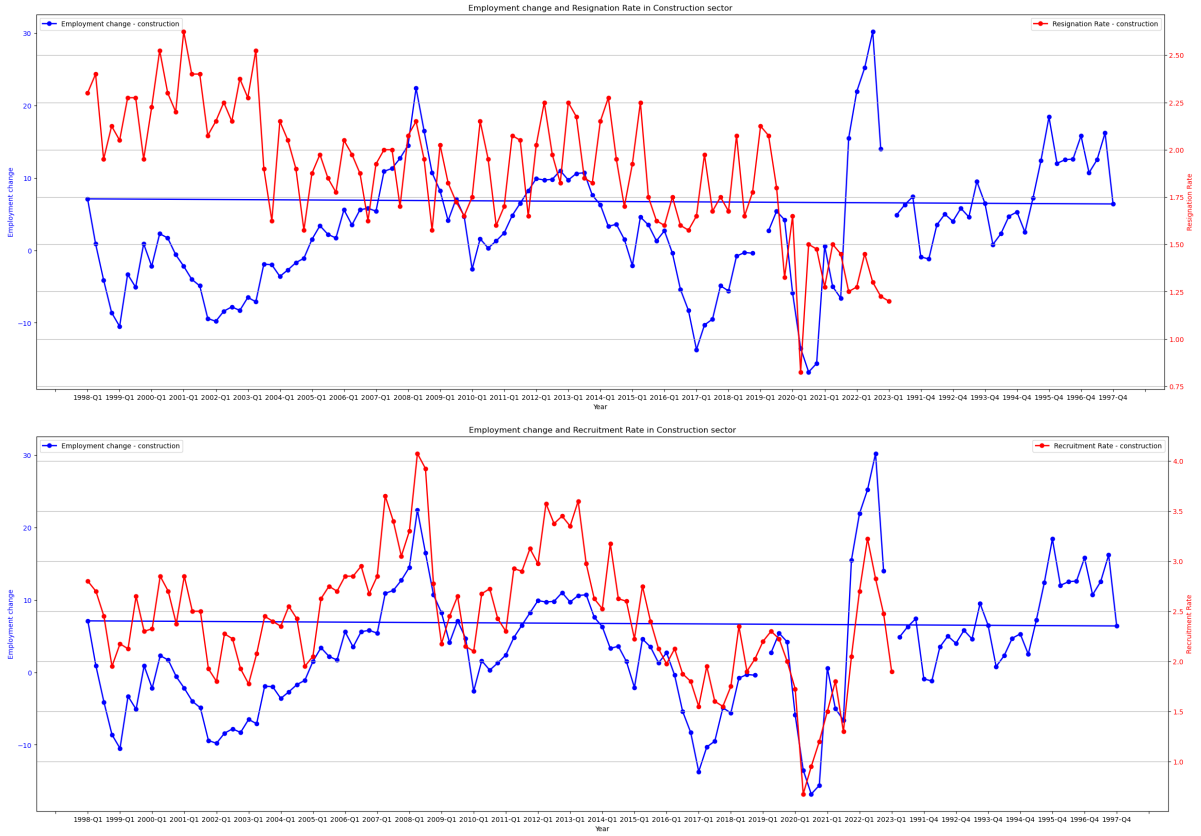


Construction Sector: Comparing the Employment change with Resignation and Recruitment rate

In [28]:

```
# Load data for Employment change
filtered_employ = employChangeC[(employChangeC['industry1'] != 'total') & (e
filtered_employ['employment_change'] = pd.to_numeric(filtered_employ['employ

# Load data for resignation rate
filtered_resign = resignC[(resignC['industry1'] != 'total') & (resignC['indu
filtered_resign['resignation_rate'] = pd.to_numeric(filtered_resign['resigna

#load data for recruitment rate
filtered_recruite = recruiteC[(recruiteC['industry1'] != 'total') & (recruit
filtered_recruite['recruitment_rate'] = pd.to_numeric(filtered_recruite['rec

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_employ.groupby('industry1'):
    # Plot job vacancy on the first y-axis
```

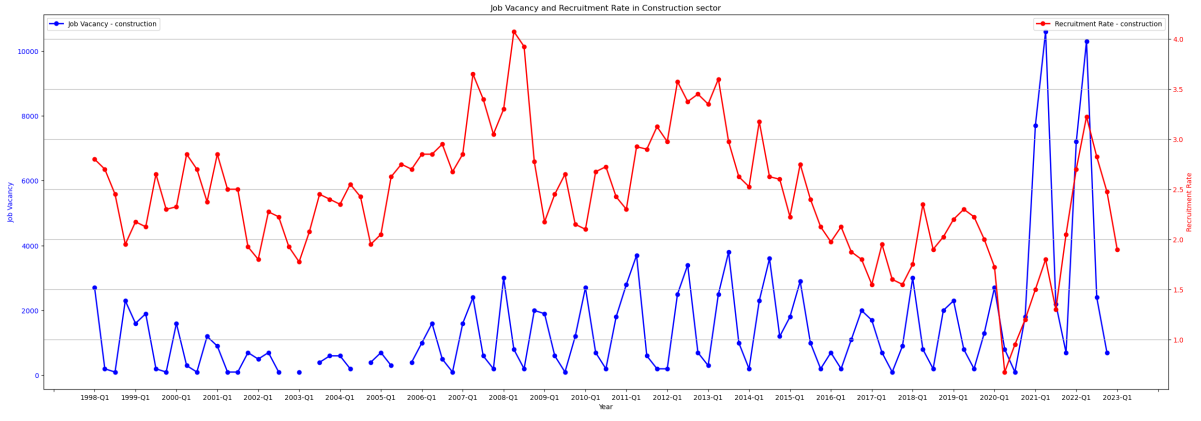
```
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for Employment change
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for resignation rate
ax2 = ax1.twinx()
for industry, group in filtered_resign.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['resignation_rate'], label=f'Resignatic

# Customize the plot for resignation rate
ax2.set_ylabel('Resignation Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Employment change and Resignation Rate in Construction sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1'
for industry, group in filtered_employ.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for Employment change
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for recruitment rate
ax2 = ax1.twinx()
for industry, group in filtered_recruite.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['recruitment_rate'], label=f'Recruitmer

# Customize the plot for recruitment rate
ax2.set_ylabel('Recruitment Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Employment change and Recruitment Rate in Construction sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```

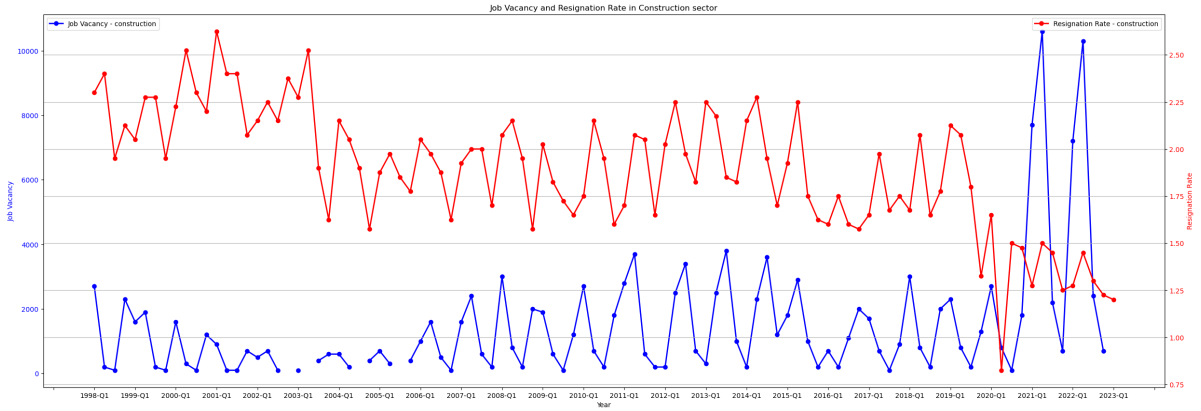


Construction Sector: Comparing Job vacancy with the Recruitment and Resignation rate

In [29]:
```
# Load data for job vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], er

# Load data for recruitment rate
filtered_recruite = recruiteC[(recruiteC['industry1'] != 'total') & (recruit
filtered_recruite['recruitment_rate'] = pd.to_numeric(filtered_recruite['rec

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_job.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['year_quarter'], group['job_vacancy'], label=f'Job Vacanc

# Customize the plot for job vacancy
ax1.set_xlabel('Year')
ax1.set_ylabel('Job Vacancy', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for recruitment rate
ax2 = ax1.twinx()
for industry, group in filtered_recruite.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['recruitment_rate'], label=f'Recruitmer

# Customize the plot for resignation rate
ax2.set_ylabel('Recruitment Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')
```

```
# Title and x-axis rotation
plt.title('Job Vacancy and Recruitment Rate in Construction sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()

# Load data for job vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], err

# Load data for resignation rate
filtered_resign = resignC[(resignC['industry1'] != 'total') & (resignC['indu
filtered_resign['resignation_rate'] = pd.to_numeric(filtered_resign['resigna

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_job.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['year_quarter'], group['job_vacancy'], label=f'Job Vacanc

# Customize the plot for job vacancy
ax1.set_xlabel('Year')
ax1.set_ylabel('Job Vacancy', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for resignation rate
ax2 = ax1.twinx()
for industry, group in filtered_resign.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['resignation_rate'], label=f'Resignatio

# Customize the plot for resignation rate
ax2.set_ylabel('Resignation Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Job Vacancy and Resignation Rate in Construction sector')

# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```

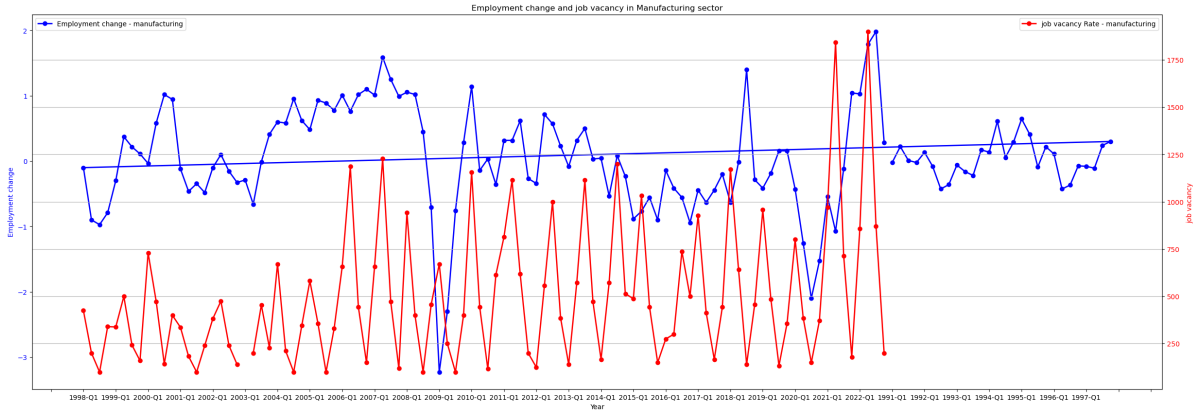




4.2 Analysing the Manufacturing sector

Manufacturing sector: Comparing Employment change with Job vacancy

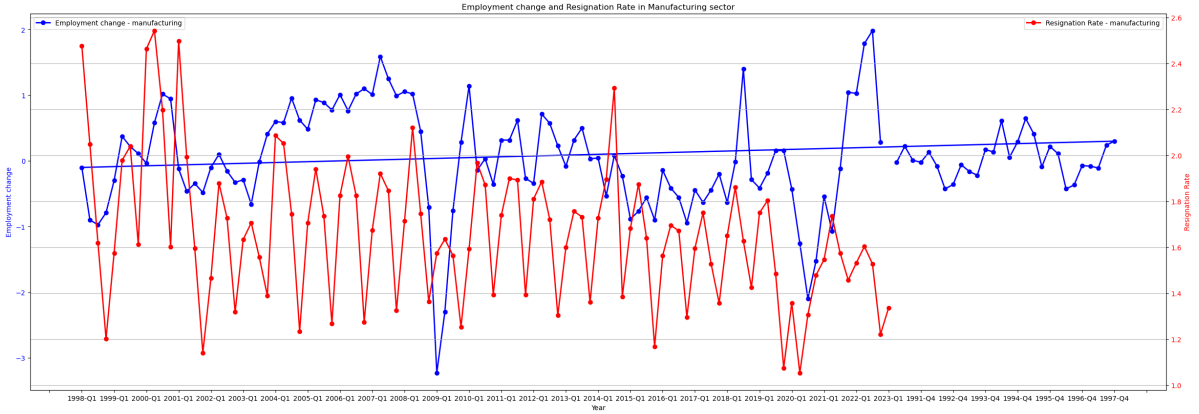
In [30]:
```
# Load data for Employment change
filtered_employ = employChangeC[(employChangeC['industry1'] != 'total') & (e
filtered_employ['employment_change'] = pd.to_numeric(filtered_employ['employ

### Load data for job vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], err

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_employ.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for Employment change
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for job vacancy rate
ax2 = ax1.twinx()
for industry, group in filtered_job.groupby('industry1'):
    # Plot Job vacancy on the second y-axis
    ax2.plot(group['year_quarter'], group['job_vacancy'], label=f'job vacanc

# Customize the plot for job vacancy rate
ax2.set_ylabel('job vacancy ', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Employment change and job vacancy in Manufacturing sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```

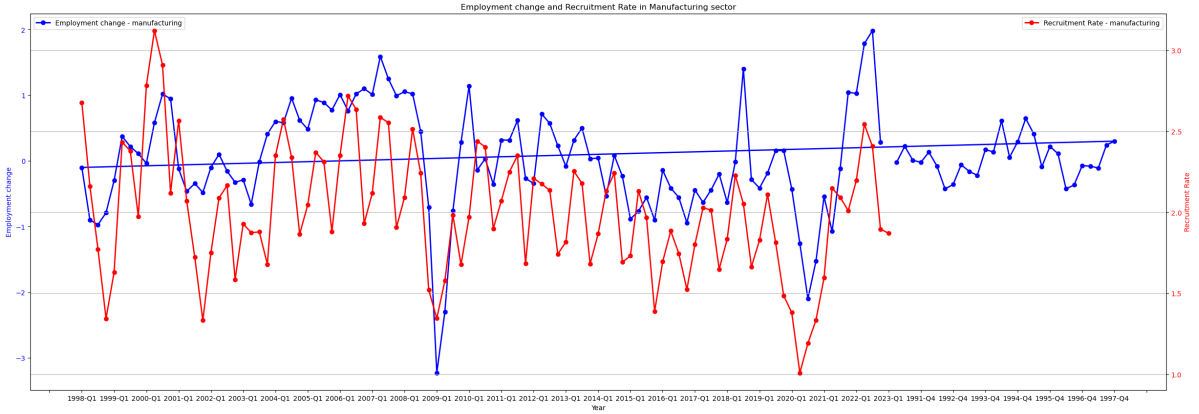


Manufacturing sector: Comparing Employment change with Resignation and Recruitment rate

In [31]:
```
# Load data for Employment change
filtered_employ = employChangeC[(employChangeC['industry1'] != 'total') & (e
filtered_employ['employment_change'] = pd.to_numeric(filtered_employ['employ

# Load data for resignation rate
filtered_resign = resignC[(resignC['industry1'] != 'total') & (resignC['indu
filtered_resign['resignation_rate'] = pd.to_numeric(filtered_resign['resigna

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_employ.groupby('industry1'):
    # Plot employment change  on the first y-axis
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for employment change
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for resignation rate
ax2 = ax1.twinx()
for industry, group in filtered_resign.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['resignation_rate'], label=f'Resignatic

# Customize the plot for resignation rate
ax2.set_ylabel('Resignation Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Employment change and Resignation Rate in Manufacturing sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()

# Load data for Employment change
filtered_employ = employChangeC[(employChangeC['industry1'] != 'total') & (e
filtered_employ['employment_change'] = pd.to_numeric(filtered_employ['employ

# Load data for recruitment rate
```

```
filtered_recruite = recruiteC[(recruiteC['industry1'] != 'total') & (recruit
filtered_recruite['recruitment_rate'] = pd.to_numeric(filtered_recruite['rec

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_employ.groupby('industry1'):
    # Plot employment change on the first y-axis
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for employment change
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for recruitment rate
ax2 = ax1.twinx()
for industry, group in filtered_recruite.groupby('industry1'):
    # Plot recruitment rate on the second y-axis
    ax2.plot(group['quarter'], group['recruitment_rate'], label=f'Recruitmer

# Customize the plot for recruitment rate
ax2.set_ylabel('Recruitment Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Employment change and Recruitment Rate in Manufacturing sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```

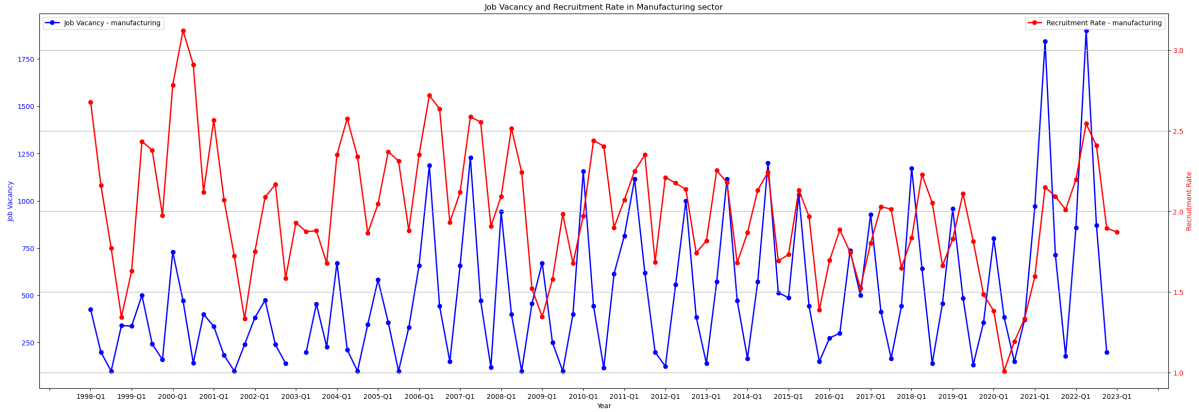


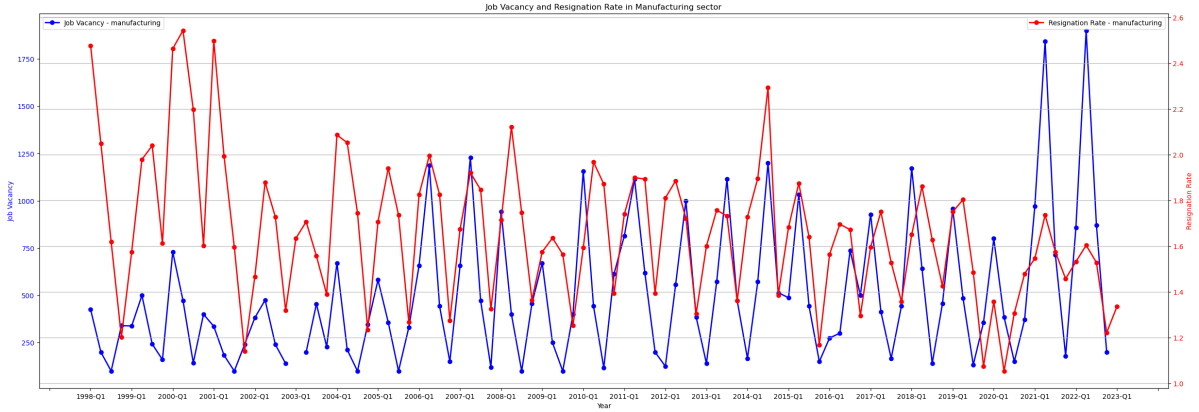


Manufacturing sector: Comparing Job vacancy with Recruitment rate and Resignation rate

In [32]:
```
# Load data for job vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], err

# Load data for recruitment rate
filtered_recruite = recruiteC[(recruiteC['industry1'] != 'total') & (recruit
filtered_recruite['recruitment_rate'] = pd.to_numeric(filtered_recruite['rec

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_job.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['year_quarter'], group['job_vacancy'], label=f'Job Vacanc

# Customize the plot for job vacancy
ax1.set_xlabel('Year')
ax1.set_ylabel('Job Vacancy', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for recruitment rate
ax2 = ax1.twinx()
for industry, group in filtered_recruite.groupby('industry1'):
    # Plot recruitment rate on the second y-axis
    ax2.plot(group['quarter'], group['recruitment_rate'], label=f'Recruitmer

# Customize the plot for recruitment rate
ax2.set_ylabel('Recruitment Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Job Vacancy and Recruitment Rate in Manufacturing sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
# Load data for job vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], err

# Load data for resignation rate
filtered_resign = resignC[(resignC['industry1'] != 'total') & (resignC['indu
filtered_resign['resignation_rate'] = pd.to_numeric(filtered_resign['resigna

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_job.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['year_quarter'], group['job_vacancy'], label=f'Job Vacanc

# Customize the plot for job vacancy
ax1.set_xlabel('Year')
ax1.set_ylabel('Job Vacancy', color='blue')
```

```
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for resignation rate
ax2 = ax1.twinx()
for industry, group in filtered_resign.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['resignation_rate'], label=f'Resignatic

# Customize the plot for resignation rate
ax2.set_ylabel('Resignation Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Job Vacancy and Resignation Rate in Manufacturing sector')

# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```



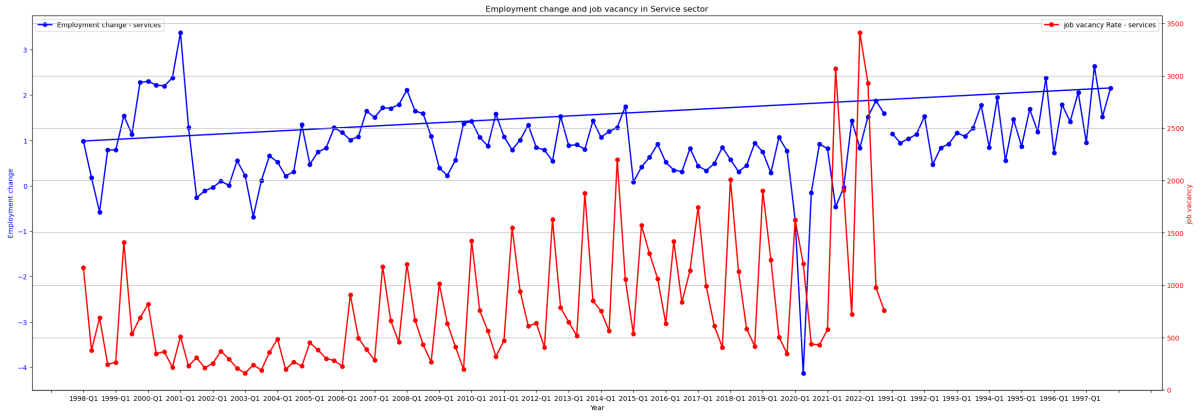


4.3 Analysing the Service sector

Service sector: Comparing Employment change with Job vacancy

In [33]:
```
# Load data for Employment change
filtered_employ = employChangeC[(employChangeC['industry1'] != 'total') & (e
filtered_employ['employment_change'] = pd.to_numeric(filtered_employ['employ

### Load data for job vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
```

```
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], er

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_employ.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for job vacancy
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for job vacancy
ax2 = ax1.twinx()
for industry, group in filtered_job.groupby('industry1'):
    # Plot job vacancy on the second y-axis
    ax2.plot(group['year_quarter'], group['job_vacancy'], label=f'job vacanc

# Customize the plot for job vacancy
ax2.set_ylabel('job vacancy ', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Employment change and job vacancy in Service sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```



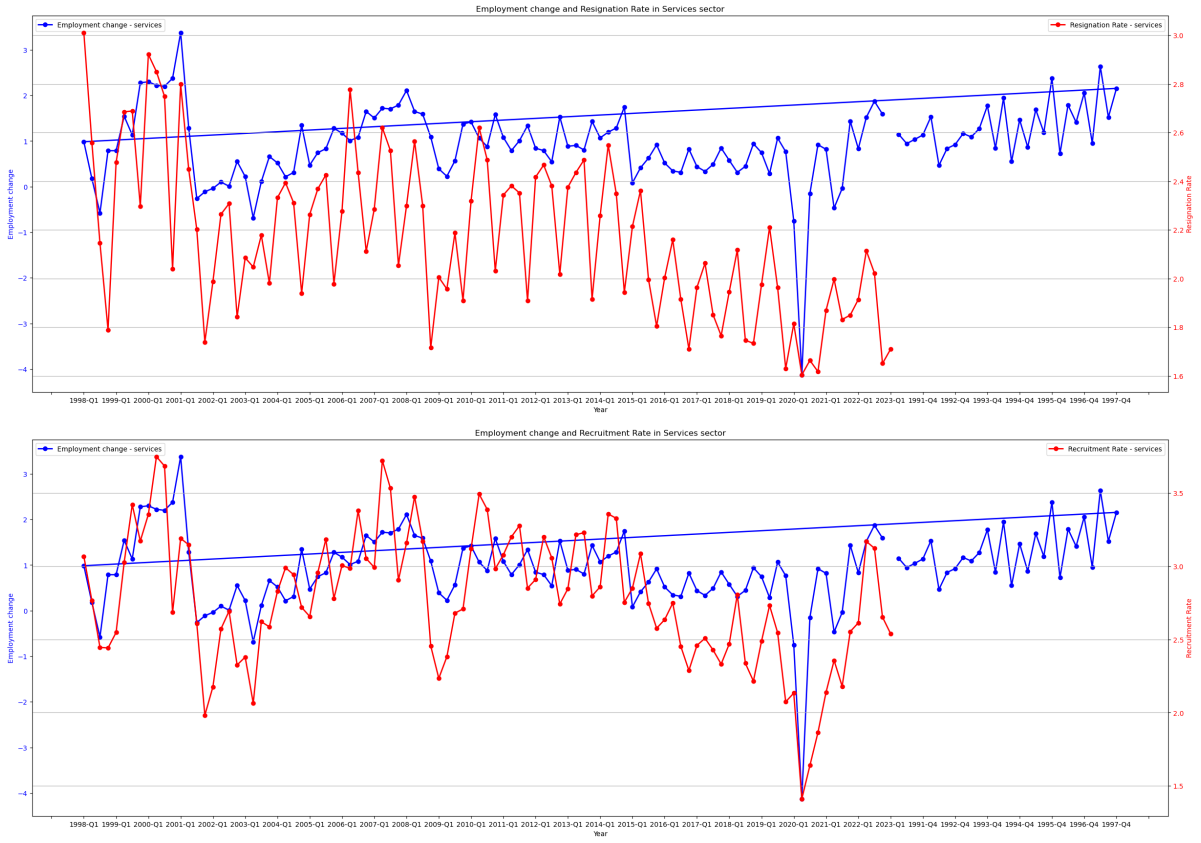
Service sector: Comparing Employment change with Resignation and Recruitment rate

In [34]:

```
# Load data for Employment change
filtered_employ = employChangeC[(employChangeC['industry1'] != 'total') & (e
filtered_employ['employment_change'] = pd.to_numeric(filtered_employ['employ

# Load data for resignation rate
filtered_resign = resignC[(resignC['industry1'] != 'total') & (resignC['indu
filtered_resign['resignation_rate'] = pd.to_numeric(filtered_resign['resigna

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))
```

```
# Iterate over unique values in 'industry1' and plot lines without connecti
for industry, group in filtered_employ.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for employment change
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for resignation rate
ax2 = ax1.twinx()
for industry, group in filtered_resign.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['resignation_rate'], label=f'Resignatic

# Customize the plot for resignation rate
ax2.set_ylabel('Resignation Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Employment change and Resignation Rate in Services sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()

# Load data for employment change
filtered_employ = employChangeC[(employChangeC['industry1'] != 'total') & (e
filtered_employ['employment_change'] = pd.to_numeric(filtered_employ['employ

# Load data for recruitment rate
filtered_recruite = recruiteC[(recruiteC['industry1'] != 'total') & (recruit
filtered_recruite['recruitment_rate'] = pd.to_numeric(filtered_recruite['rec

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_employ.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['quarter'], group['employment_change'], label=f'Employmer

# Customize the plot for employment change
ax1.set_xlabel('Year')
ax1.set_ylabel('Employment change', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for recruitment rate
ax2 = ax1.twinx()
for industry, group in filtered_recruite.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['recruitment_rate'], label=f'Recruitmer

# Customize the plot for recruitment rate
ax2.set_ylabel('Recruitment Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
```

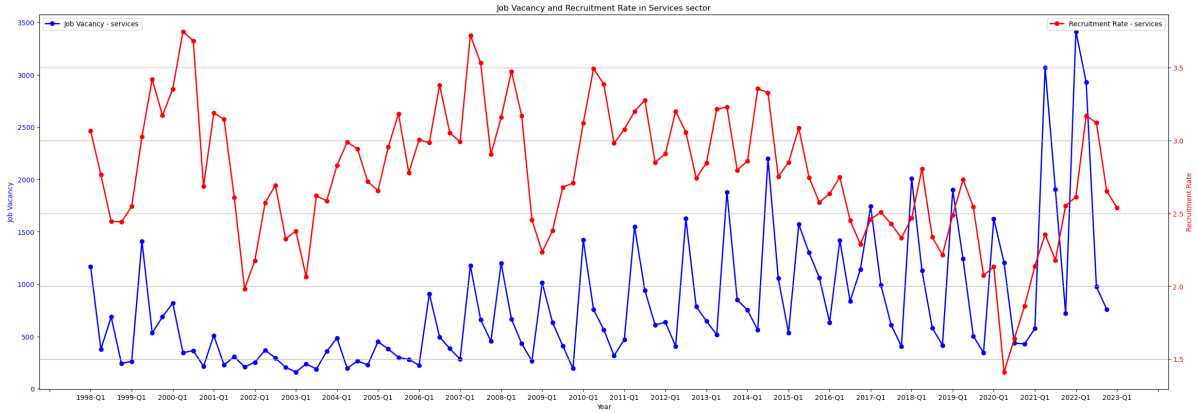
```
plt.title('Employment change and Recruitment Rate in Services sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```

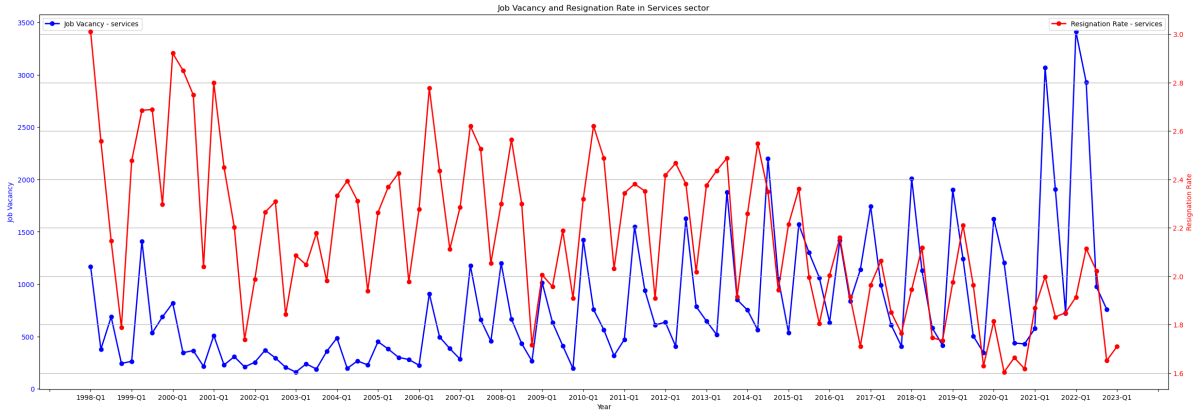


Service sector: Comparing Job vacancy with Recruitment and Resignation rate

In [35]:
```
# Load data for job vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], err

# Load data for recruitment rate
filtered_recruite = recruiteC[(recruiteC['industry1'] != 'total') & (recruit
filtered_recruite['recruitment_rate'] = pd.to_numeric(filtered_recruite['rec

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_job.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['year_quarter'], group['job_vacancy'], label=f'Job Vacanc

# Customize the plot for job vacancy
ax1.set_xlabel('Year')
ax1.set_ylabel('Job Vacancy', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for recruitment rate
ax2 = ax1.twinx()
for industry, group in filtered_recruite.groupby('industry1'):
    # Plot resignation rate on the second y-axis
```

```
    ax2.plot(group['quarter'], group['recruitment_rate'], label=f'Recruitmer

# Customize the plot for resignation rate
ax2.set_ylabel('Recruitment Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Job Vacancy and Recruitment Rate in Services sector')
# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()

# Load data for job vacancy
filtered_job = jobVacancyC[(jobVacancyC['industry1'] != 'total') & (jobVacar
filtered_job['job_vacancy'] = pd.to_numeric(filtered_job['job_vacancy'], err

# Load data for resignation rate
filtered_resign = resignC[(resignC['industry1'] != 'total') & (resignC['indu
filtered_resign['resignation_rate'] = pd.to_numeric(filtered_resign['resigna

# Plotting
fig, ax1 = plt.subplots(figsize=(30, 10))

# Iterate over unique values in 'industry1' and plot lines
for industry, group in filtered_job.groupby('industry1'):
    # Plot job vacancy on the first y-axis
    ax1.plot(group['year_quarter'], group['job_vacancy'], label=f'Job Vacanc

# Customize the plot for job vacancy
ax1.set_xlabel('Year')
ax1.set_ylabel('Job Vacancy', color='blue')
ax1.tick_params(axis='y', labelcolor='blue')
ax1.legend(loc='upper left')

# Create a second y-axis for resignation rate
ax2 = ax1.twinx()
for industry, group in filtered_resign.groupby('industry1'):
    # Plot resignation rate on the second y-axis
    ax2.plot(group['quarter'], group['resignation_rate'], label=f'Resignatic

# Customize the plot for resignation rate
ax2.set_ylabel('Resignation Rate', color='red')
ax2.tick_params(axis='y', labelcolor='red')
ax2.legend(loc='upper right')

# Title and x-axis rotation
plt.title('Job Vacancy and Resignation Rate in Services sector')

# Set x-axis to display every 4th iteration
ax1.xaxis.set_major_locator(MultipleLocator(4))

plt.grid(True)
plt.show()
```





5 Exploratory data analysis

5.1 Analysis on Construction sector

Employment Change vs Job Vacancy

Comparison:

- Employment change demonstrates cyclical flucturations of growth and decline. Whereas the job vacancy in the construction sector exhibited greater variability with notable spikes from 2020 Q4-2021 Q2.

Anomalies or Observations:

- There are 2 significant spikes in the job vacancy rate around 2020 Q4-2021 Q2 and 2021 Q4-2022 Q2, with only a single spike in employment change in 2021 Q4. The first anomaly could be attributed to the impact of the COVID-19 Pandemic where many projects may have been halted or delayed, leading to an increase in vacancies as companies anticipated a return to work.
- The second spike in both Job Vacancy and employment change could arise as many of these positions were not filled immediately, possibly due to lockdowns, health concerns, or economic uncertainty.This could have been a factor to the high rates in employment change as well.

Link Between the Graphs:

- Generally, one might expect a direct relationship between job vacancies and employment changes; as vacancies increase, employment should also rise as positions are filled. However, the graph shows that this is not always the case.
- Outside of the anomaly, there does seem to be some correlation between the two datasets. At various points where the job vacancy rate increases or decreases, there is often a corresponding change in employment, as one might expect under normal economic conditions. This indicates that, in general, the two are linked, with fluctuations in job vacancies influencing employment changes in the construction sector.

Employment change vs Resignation and Recruitment rate

Comparison:

- The recruitment rate, resignation rate and Employment change all exhibits seasonal patterns, with regular fluctuations within each year, with employment change containing peaks and trough at regular intervals. While resignation and recruitment rate does not mirror the same peaks and troughs as the employment change exactly.
- There is a correlation, at times, between employment change and recruitment rate, while resignation rate contains a looser correlation. When employment change increase, recruitment rates also go up, indicating industry growth and a higher demand for workers. However, the connection employment change and resignations is inconsistent over time.

Anomalies or Observations:

- Around 2008 Q3-2009 Q2, there is a significant dip in employment change, which is likely associated with the global financial crisis. Similarly, there are spikes in the resignation rate represent an increase during the same period.
- Similarly, around 2019 Q2-2020 Q2 there was a large dip in employment change and both resignation and recruitment rates, likely caused from both a recession and COVID-19.
- In the long term, employment change in construction increased until around 2008 and then became more volatile. Resignation rates remained relatively steady, while recruitment rate displayed cyclical patterns over the years without a clear overall trend.

Link Between the Graphs:

- While there are instances where these resignation and recruitment rates move in tandem with employment changes, there are also periods where this is not the case, highlighting that other factors are at play.
- The increase in employment is largely driven by non-residents in the construction sector (Buisness Times, 2022). Singapore's labour market statistics is also based on

residents only. As such, resignation rates, recruitment rates and employment change would greatly vary as there are external factors coming into play.(Gov.sg, 2022)
- The 2008-2009 financial crisis period along with COVID-19 in 2020 shows a significant downturn in employment change, which is mirrored by disruptions in the usual patterns of recruitment and resignation rates. However, the link between employment change and resignation rate is not all the same.

Job vacancy vs Recruitment and Resignation rate

Comparison:

- The job vacancy trend fluctuates over time with a general increasing trend from 2003 to around 2008, followed by a decline, then another 2 peakings and troughs from 2021 Q2 to 2022 Q4. The recruitment rate, follows a similar pattern to the job vacancies but with less volatility.

- The resignation rate, however, shows a different pattern and is relatively stable with minor fluctuations until around 2020, where there is a noticeable spike. This spike corresponds with the pandemic period, which could suggest a higher turnover in the construction sector during that time.

Anomalies or Observations:

- Post-2020, there is a significant spike in both job vacancies and recruitment rates, followed by a rapid decline.

- There is a sharp decrease in resignation rate and recruitment rate in Q2 of 2020.

- There are spikes in recruitment rate that do not always correspond to the highest peaks in job vacancies, such as 2008, where recruitment rate was at its peak and had a spike decrease whereas job vacancies and resignation rates remain constant.

Link between the Graphs:

- The increase in job vacancies and recruitment rates reflected the industry's efforts to meet the delayed project timelines and cope with the rising manpower costs due to strict border controls affecting the inflow of migrant workers. (Straits Times, 2021)

- The decline in job vacancies and recruitment rates reflected the aftermath of the economic slowdown after collapses of several companies due to insufficient capabilities to cope with the backlog of projects. (Straits Times, 2021)

5.2 Analysis on the Manufacturing sector

Employment change vs Resignation rate and Recruitment rate

Comparison:

- The resignation rate fluctuates more abruptly than the employment change, which seems to have a smoother trajectory over the years, whereas the recruitment rate exhibits more pronounced peaks and troughs compared to the resignation rate.
- There appears to be a general trend where increases in employment change is often followed by increases in resignation rates, although the two do not always move in tandem.
- There seems to be a rough inverse relationship, especially noticeable during times of sharp changes in employment. For example, when the employment change drops significantly, the recruitment rate tends to spike shortly after.

Anomalies or Observations:

- The period around the 2008-2009 financial crisis shows a significant drop in employment change followed by a substantial increase in recruitment rate, indicating a possible lagged response in hiring after job losses.
- Anomalies can be seen with spikes around 2001 Q1, 2008-2009, and 2020 Q2, where there are sharp drops or rises in the employment change, followed by subsequent drops or increase in resignation rates, with drops likely corresponding to economic events such as recessions or the COVID-19 pandemic.

Link between the graphs:

- Both graphs indicate that the employment change can be a leading indicator for movements in both resignation and recruitment rates, although the relationship is not perfectly consistent.
- The graphs suggest that as employment change grows, resignations may increase possibly due to more opportunities for workers to move to better opportunity jobs, while recruitment efforts might decrease as less new hiring is needed. (mycareersfuture, 2023)

Job Vacancy vs Resignation and Recruitment rate

Comparison:

- The job vacancy trend fluctuates with what appears to be a seasonal pattern, peaking typically in certain quarters and dipping in others.
- The recruitment rate and resignation rate follows a similar seasonal pattern with less pronounced peaks and valleys.
- The graphs show that while there is a general correspondence between job vacancies and recruitment rates, the relationship between job vacancies and resignation rates is less direct.

Anomalies or Observations:

- In both graphs, there are significant spikes in the recruitment and resignation rates around the year 2000 Q1 as compared to job vacancy.
- 2020 Q2 huge drop in recruitment rate and in resignation rate. General direct relationship between the recruitment rate, resignation rate and job vacancy.
- The link reveals that in 2000 Q1, there are high recruitment and resignation rates despite very low job vacancies.
- From 2008 Q4-2009 Q2 there was a significant drop in recruitment rate despite increase in resignation rate and job vacancy.

Link Between the Graphs:

- The graphs show that there is a general correspondonce between job vacancies, resignation and recruitment rates. Although resignation rates is less direct.
- The significant drop in recruitment rate from 2008-2009 could be attributed from the electronics cluster being affected by weak global semiconductor demand. (Ministry of trade and industry, 2008)

5.3 Analysis on the Services sector

Employment change vs Resignation rate and Recruitment rate

Comparison

- The resignation rate fluctuates significantly and appears to have a more volatile pattern than the employment change, which seems to exhibit a general upward trend over time, with some fluctuations.
- There are periods where the resignation rate spikes or drops sharply, such as around 2001, 2008, and particularly in 2020, possibly indicating responses to economic events.
- The overall trend suggests that the services sector has experienced growth in employment over the long term, but the resignation rate does not consistently follow the same pattern.

Anomalies or observations

- The recruitment rate shows a jagged pattern with significant ups and downs, and it seems to correlate with the employment change but not as directly as the resignation rate.
- The graph shows that recruitment rates do not always immediately follow declines in employment change. For example, during the 2008 financial crisis, employment change dropped significantly, but the recruitment rate did not show an immediate spike.

- The most recent data, around 2020 - 2022 Q2, indicates a sharp decline in employment change, likely due to the COVID-19 pandemic, followed by a sharp increase in the recruitment rate, suggesting a rebound or recovery effort in hiring.

Link between the graphs:

- The services sector seems to be more sensitive to different types of economic shocks, as indicated by the sharp changes in resignation and recruitment rates.
- The graphs suggest an underlying dynamic in the services sector where employment changes can lead to delayed responses in recruitment activities, while resignation rates may react more immediately to changes in employment, but with greater volatility.
- During the COVID-19 pandemic, Service industries were the last to re-open due to transmisssion risks. Resulting in the delay in recruitment rates. Whereas resignation rates could remain as volatile. (Unece, 2021)

Job Vacancy vs Resignation and Recruitment rate

Comparison:

- The general trend shows that job vacancy rate and both resignation rate in the service sector have a correlated pattern, with both rates often increasing or decreasing around the same periods.
- The general trend shows a more pronounced correlation between the recruitment rate and the job vacancy rate, with both tending to rise and fall together over the observed period.
- This pattern may indicate that an increase in job vacancies often leads to more recruitments as companies seek to fill open positions.

Anomalies or Observations:

- There are instances where resignation rate and job vacancy diverge. For example, around 2009 Q1, there is a significant increase in job vacancies while the recruitment rate decreases.
- The peak caused by COVID-19 in Job vacancies in both graphs have different impacts on recruitment and resignation rates. The rise in resignation rates are not as significant while recruitment rate contains general spikes similarly to job vacancies.
- There was an overall decrease in recruitment and resignation rate from 2014 Q2 to 2020 despite an increase in job vacancy.

Link between the graphs:

- Similar to the manufacturing sector, the graphs show that while there is a general correspondence between job vacancies and recruitment rates, the relationship between job vacancies and resignation rates is less direct.

- The rise in recruitment rates could be due to increasing non-resident employment as border restrictions were progressively lifted. On the contrary, the lower rise in resignation rates suggests that amidst uncertanties, individuals are more cautious about leaving their current employment. (Human resources director,2023)

6 Summary

6.1 Conclusion

In our data analysis, we analysed and discovered how different industries are interconnected in terms of job vacancy, employment change, resignation, and recruitment rates.

In the construction sector, we discovered that based on the data provided, there isn't a direct correlation between job vacancies and recruitment or resignation rates. Employment changes, however, showed a closer relationship with recruitment rates.

On the other hand, the manufacturing industry displayed a strong connection between job vacancy, employment change, recruitment, and resignation rates, indicating a more direct influence on one another.

Similarly, in the service industry, job vacancy, employment change and recruitment appeared closely linked, while resignation rates may differ, highlighting a more intricate relationship within this sector.

In summary, our analysis underscores the varied dynamics across industries. Although it may seem that job vacancy and employment change rates may directly affect recruitment and resignation rates, external factors such as types of regulations come into play in the complexities of these industries.

6.2 Summary of prepped data

The dataset I have chosen covers broad sectors like manufacturing, services, and construction, each comprising 32 to 36 distinct sub-sectors. However, as highlighted earlier, a comprehensive understanding of the industry requires a more nuanced approach.

Analysing the entire industry may not show the most accurate answers, given the numerous factors influencing each sub-sector. To gain a clearer and more detailed perspective, it makes sense to conduct a deeper analysis of the various sub-industries as there will be lesser external factors to take into consideration during the analysis.

7 References and Resources

7.1 References

- Buisness Times. (2022, April 28). 'Singapore employment grows by 41,100 in Q1 2022 as construction workers return'. Available:https://www.businesstimes.com.sg/singapore/economy-policy/singapore-employment-grows-41100-q1-2022-construction-workers-return

- Gov.sg (2020 Janurary 3). 'Why are labour market statistics based on residents and not Singapore Citizens'. Available: https://www.gov.sg/article/why-are-labour-market-statistics-based-on-residents-and-not-singapore-citizens

- Straits Times (2021, September 27).'Things just went south: Construction firms under pressure as pandemic bites deep' Available:https://www.straitstimes.com/business/property/things-just-went-south-construction-firms-under-pressure-as-pandemic-bites-deep

- mycareersfuture (2023, September 29).'The Great Reshuffle: What S'pore Employees Want and How Can Employers Support Them?' Available: https://content.mycareersfuture.gov.sg/great-reshuffle-what-employees-want-how-employers-support/#:~:text=%E2%80%9CThe%20Great%20Resignation%E2%80%9D%20was%2

- Ministry of Trade and Industry (2008, November 8). 'Economic Survey of Singapore Third Quarter 2008' Available: https://www.mti.gov.sg/-/media/MTI/Resources/Economic-Survey-of-Singapore/2008/Economic-Survey-of-Singapore-Third-Quarter-2008/ess_3q08_full_report.pdf?la=en

- Human resources director (2023, July 29).'Singapore's total employment sees slower expansion in second quarter' Available: https://www.hcamag.com/asia/specialisation/recruitment/singapores-total-employment-sees-slower-expansion-in-second-quarter/454409

- UNECE (2021, October 31).'Measurement of Quality of Employment during COVID-19, Singapore' pp9. Available: https://unece.org/sites/default/files/2021-11/Session_1_Singapore%20paper.pdf

7.2 Resources

Data sources

- Data.gov.sg (2022, Dec) Job Vacancy by Industry and Occupational Group, Annual. Available: https://beta.data.gov.sg/collections/688/datasets/d_83bb74eba03d9a16b7d479ccab14
- Data.gov.sg (2022, Dec) Employment Change By Industry, Quarterly. Available: https://beta.data.gov.sg/collections/647/view
- Data.gov.sg (2023, Mar) Average Monthly Recruitment/Resignation Rates by Industry and Occupational Group, Quarterly. Available: https://beta.data.gov.sg/collections/683/datasets/d_933bf86adb0d764dd9ba7ce967ee

Exploratory data analysis

- Matplotlib. Customizing Matplotlib with style sheets and rcParams [Online]. Available: https://matplotlib.org/stable/tutorials/introductory/customizing.html
- Matplotlib. legends guide [Online]. Available: https://matplotlib.org/stable/users/explain/axes/legend_guide.html
- Matplotlib. Axes ticks [Online]. Available: https://matplotlib.org/stable/users/explain/axes/axes_ticks.html
- Matplotlib. Dual Axis [Online]. Available: https://matplotlib.org/stable/gallery/subplots_axes_and_figures/two_scales.html

Data cleaning

- Pandas. DataFrame.to_csv [Online]. Available: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.to_csv.html
- Pandas. DataFrame.groupby [Online]. Available: https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.groupby.html