

RP Data-Cleaning

November 22, 2025

1 Intro

This code processes job description features from RavenPack

```
[1]: import pandas as pd

[2]: pd.set_option('display.max_colwidth', 100)
pd.set_option('display.max_rows', 2000)
pd.set_option('display.max_columns', 2000)

[3]: home = "D:/User.Data/Dan.Li/"
job_record_dir = "D:/User.Data/Dan.Li/RP LinkUp/"
job_record_output_dir = "D:/User.Data/Dan.Li/RP LinkUp_Proccsed/"
job_record_sum_output_dir = "D:/User.Data/Dan.Li/RP LinkUp_Sum/"

[4]: names_focal = []
import os
for root, dirs, files in os.walk(job_record_dir):
    for file in files:
        if file.endswith(".csv"):
            names_focal.append(root+"/"+file)

[5]: names_focal

[5]: ['D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2007/2007-08.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2007/2007-09.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2007/2007-10.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2007/2007-11.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2007/2007-12.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-01.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-02.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-03.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-04.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-05.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-06.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-07.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-08.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2008/2008-09.csv',
```

[illegible]

[illegible]


```

'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2020/2020-07.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2020/2020-08.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2020/2020-09.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2020/2020-10.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2020/2020-11.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2020/2020-12.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-01.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-02.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-03.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-04.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-05.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-06.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-07.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-08.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-09.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-10.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-11.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-12.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-01.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-02.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-03.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-04.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-05.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-06.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-07.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-08.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-09.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-10.csv',
'D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-11.csv']

```

```

[ ]: # Read each CSV file, extract relevant columns, and save as a pickle file for
      ↪faster access later.
      # Only process files up to 2022-01, as later samples are not relevant.
      for file in names_focal:

          pkl_path = job_record_output_dir + file.split("/")[5].split(".")[0] + ".pkl"
          print(file)
          job_records = pd.read_csv(file, low_memory = False,
          ↪usecols=["RP_DOCUMENT_ID",
                  "RP_ENTITY_ID",
                  "ENTITY_TYPE",
                  "ENTITY_NAME", "PROVIDER_DOCUMENT_CHAIN_ID"])

          job_records.to_pickle(pkl_path)
          print(file + " is finished.")

```

D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-06.csv

D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-06.csv is finished.

```

D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-07.csv
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-07.csv is finished.
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-08.csv
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-08.csv is finished.
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-09.csv
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-09.csv is finished.
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-10.csv
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-10.csv is finished.
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-11.csv
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-11.csv is finished.
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-12.csv
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2021/2021-12.csv is finished.
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-01.csv
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-01.csv is finished.
D:/User.Data/Dan.Li/RP LinkUp/RavenPackEdge_LINKUP_2022/2022-02.csv

```

```

[9]: csv_path = home + "jobs-taxonomy.csv"
      taxonomy = pd.read_csv(csv_path, low_memory = False, encoding= 'unicode_escape')

```

```

[10]: # Merge taxonomy features and aggregate the features into job posting level

      for file in names_focal:

          pk1_path = job_record_output_dir + file.split("/") [5].split(".")[0] + ".pk1"
          pk1_clean_path = job_record_sum_output_dir + file.split("/") [5].split(".")
          ↪ "[0] + "_clean.pk1"
          pk1_sum_path = job_record_sum_output_dir + file.split("/") [5].split(".")[0]
          ↪+ "_sum.pk1"

          job_records = pd.read_pickle(pk1_path)
          job_records2 = job_records.merge(taxonomy, left_on = ["RP_ENTITY_ID"],
          ↪right_on = ["RP_ENTITY_ID2"], how = "inner")

          job_records2.to_pickle(pk1_clean_path)

          job_records_sum = job_records2.groupby(["PROVIDER_DOCUMENT_CHAIN_ID",
          ↪"TOPIC2", "GROUP2", "TYPE2"]).count().reset_index()
          job_records_sum2 = job_records_sum.pivot(index='PROVIDER_DOCUMENT_CHAIN_ID',
          ↪columns='TYPE2', values='RP_DOCUMENT_ID').reset_index()

          job_records_sum2.to_pickle(pk1_sum_path)

```

```

[11]: for file in names_focal:

          pk1_sum_path = job_record_sum_output_dir + file.split("/") [5].split(".")[0]
          ↪+ "_sum.pk1"

```

```
job_records_sum = pd.read_pickle(pk1_sum_path)

pd_sum = pd_sum.append(job_records_sum)

print(pk1_sum_path + " is finished.")
```

```
[12]: pd_sum_clean = pd_sum.groupby(["PROVIDER_DOCUMENT_CHAIN_ID"]).sum()
```

```
[13]: pd_sum_clean = pd_sum_clean.reset_index()
```

```
[14]: pk1_sum_path = "pd_sum_clean.pkl"
pd_sum_clean.to_pickle(pk1_sum_path)
```