Exploratory Data Analysis (EDA)

Visualizing and Interpreting Job Market Trends

Connor Coulter

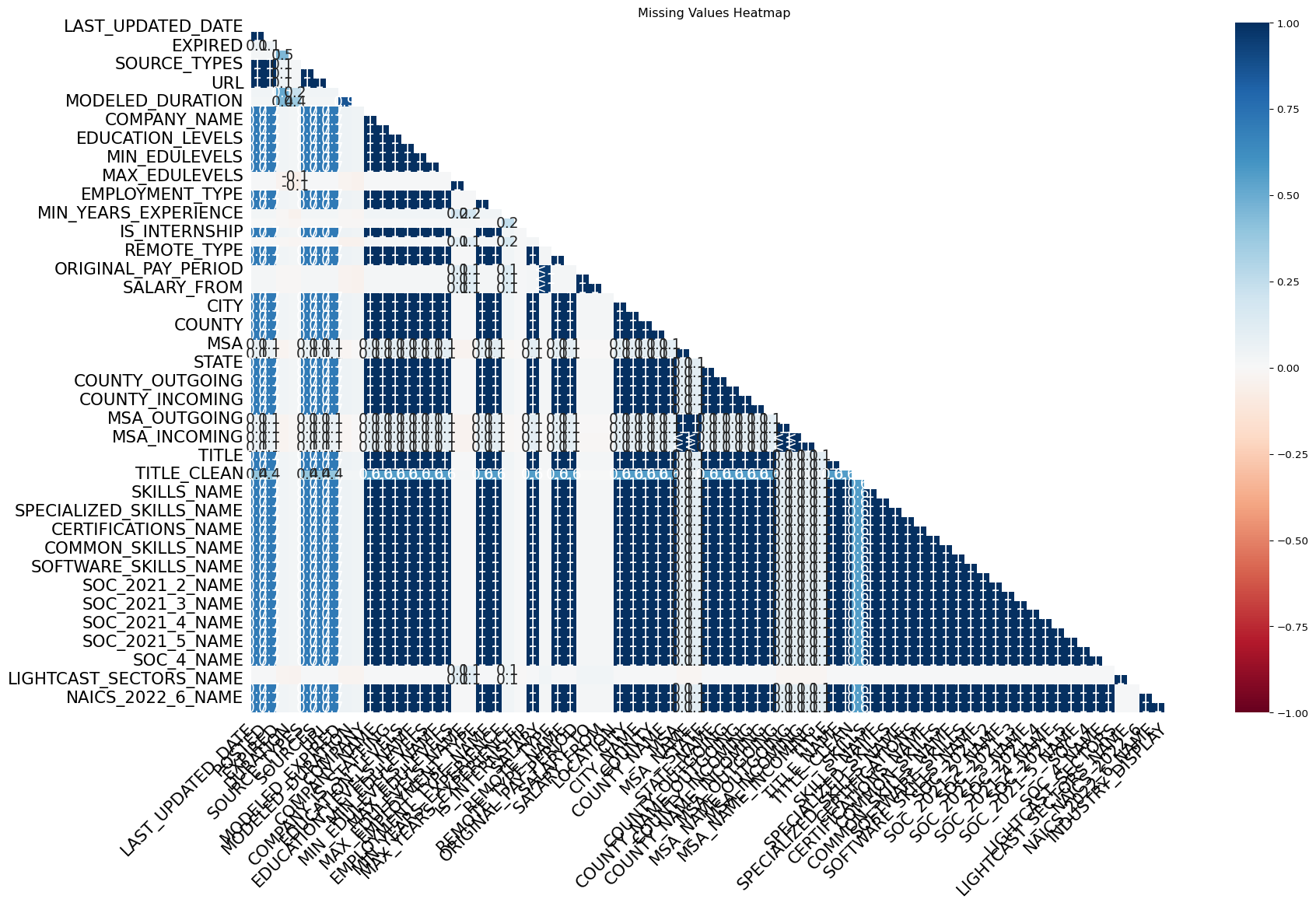
Wei Wang

Balqis Bevi Abdul Hannan Kanaga

### Exploratory Data Analysis (EDA)

Loaded dataset: (72498, 131)  
Derived non-null: {'INDUSTRY\_DISPLAY': np.int64(72454), 'SALARY\_DISPLAY': np.int64(72498)}  
Remaining columns (first 30): ['LAST\_UPDATED\_DATE', 'POSTED', 'EXPIRED', 'DURATION', 'SOURCE\_TYPES', 'SOURCES', 'URL', 'MODELED\_EXPIRED', 'MODELED\_DURATION', 'COMPANY', 'COMPANY\_NAME', 'COMPANY\_IS\_STAFFING', 'EDUCATION\_LEVELS', 'EDUCATION\_LEVELS\_NAME', 'MIN\_EDULEVELS', 'MIN\_EDULEVELS\_NAME', 'MAX\_EDULEVELS', 'MAX\_EDULEVELS\_NAME', 'EMPLOYMENT\_TYPE', 'EMPLOYMENT\_TYPE\_NAME', 'MIN\_YEARS\_EXPERIENCE', 'MAX\_YEARS\_EXPERIENCE', 'IS\_INTERNSHIP', 'SALARY', 'REMOTE\_TYPE', 'REMOTE\_TYPE\_NAME', 'ORIGINAL\_PAY\_PERIOD', 'SALARY\_TO', 'SALARY\_FROM', 'LOCATION']

/tmp/ipykernel\_2871/2537287560.py:32: FutureWarning:  
  
A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.  
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.  
  
For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform the operation inplace on the original object.



/tmp/ipykernel\_2871/2537287560.py:70: FutureWarning:  
  
A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.  
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.  
  
For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform the operation inplace on the original object.  
  
  
  
/tmp/ipykernel\_2871/2537287560.py:73: FutureWarning:  
  
A value is trying to be set on a copy of a DataFrame or Series through chained assignment using an inplace method.  
The behavior will change in pandas 3.0. This inplace method will never work because the intermediate object on which we are setting values always behaves as a copy.  
  
For example, when doing 'df[col].method(value, inplace=True)', try using 'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value) instead, to perform the operation inplace on the original object.

Removed 3300 duplicates using ['TITLE', 'COMPANY\_NAME', 'LOCATION', 'POSTED']  
Final SALARY\_DISPLAY Non-Null Count: 69198

# Job Postings by Industry (Top 15)

Unable to display output for mime type(s): text/html

Unable to display output for mime type(s): text/html

**Rationale & Insights** - **Why**: Highlights sectors where demand is concentrated, showing which industries are actively hiring. - **Key Insights**: The top three industries by job postings are Temporary Help Services, Miscellaneous Ambulatory Health Care Services, and Semiconductor and Related Device Manufacturing.

# Salary Distribution by Industry (Top 15)

Unable to display output for mime type(s): text/html

**Rationale & Insights** - **Why**: Shows where negotiation power exists and highlights industries paying well. - **Key Insights**: Automotive Parts and Accessories Retailers show a wide range (negotiation potential), while Barber Shops show a narrow range (little negotiation).

# Remote vs. On-Site Jobs

Unable to display output for mime type(s): text/html

**Rationale & Insights** - **Why**: Workplace flexibility is a major factor in today’s job market. - **Key Insights**: Most postings (78.3%) don’t specify remote status. About 17% are remote, 3.1% hybrid, and 1.6% explicitly not remote.