SUMMARY DOCUMENT

Introduction

This document provides the procedures and methods on how I cleaned the noisy data that is ‘messy_data.csv’. This csv file contains various anomalies or impurities thus this impure data set is cleaned with the help of Python codes. ‘Datacleaning.ipynb’ contains the code for how I cleaned the missing values, duplicate values, incorrect formats, etc from this messy_data.

Objective

The objectives are:

a. Handle Missing Values
b. Remove Duplicates
c. Correct Email Formats
d. Clean Name Fields
e. Standardize Date Formats
f. Correct Department Names
g. Handle Salary Noise

Procedure

1. Data loading

Firstly the data set messy_data.csv file is loaded to the JupiterNotebook which is a tool with Python as the programming language, provided by Anaconda. A library pandas is imported for loading the csv file.

2. Inspect the Data

The given noisy data is examined to understand the structure and the type of errors are identified, so here there are 7 columns and 11001 rows, the columns contain ID, Name, Age, Email, Join Date, Salary and Department.

3. Handle the missing values

The missing values are identified, and I understand that there are no values present in many cells in the csv file this problem is solved firstly by some methods such as if the percentage of missing values is more than 50% i.e., it provides less information or insufficient information then these cells are dropped from the dataset, another method is the cell are divided into categorical and numerical values if it's numerical then missing values were filled using the mean of the respective column or if the cell contains categorical values then the missing values were filled using the mode of the respective cell i.e., most frequently used values are provided.

4. Removing Duplicate Rows

After handling the missing values it’s important to remove the duplicate values in each cell, this involves checking for rows where all column values match another row exactly. After finding the duplicate values the cell is dropped or removed, and then these changes are saved to the csv file.

5. Correct Email formats

Then invalid email formats are identified and corrected thereby ensuring only professional email addresses are present in the csv file that is all the improper email ids are identified and corrected. For this, a library ‘re’ which is used for working with regular expressions is imported. Regular expression is used for validating an email, invalid Email ids are corrected some professional domains are added like gmail.com', 'yahoo.com', 'hotmail.com' etc to check whether they are in the correct formats or not.

6. Clean Name Fields

For cleaning the incorrect name formats, firstly non-alphabetic characters and extra spaces are removed. Split the name into parts and each part is capitalized.

7. Standardize Date Formats

The 'Join Date' column, which contained dates in various formats, was standardized to a single format. Here I used ‘*pd.to_datetime*’ to parse dates and convert them to the YYYY-MM-DD format. Replaces invalid dates with ‘None’.

8. Correct Department Names

When I checked the csv I found out that the department contains many improper values or alphabets for example instead of Sales it's given as SalesE, SalesY, SalesM, etc... and in the case of Support its SupportJ, SupportM,SupportF etc… thus to remove this I added a function which defined a mapping for correcting department names and I added the common typos and their correct forms thereby I corrected the names of each dept.

9. Handle Salary Noise

The salary column contains many noises thus first of all I removed the non-numeric values, then I defined a salary range i.e., an approximate range like here I put the range in between $30,000 - $200,000. Then I removed the outliers that is the salary value that falls outside the defined range. After the removal of outlier, I filled NaN values with the mean salary.

Conclusion

I cleaned the dataset 'messy_data' to improve the data quality this dataset can be further used for analysis and other operations, I tried to remove most of the noises in the dataset like duplicate values, standardized date formats, correct email formats etc. Now the dataset has more accuracy and thus can be used for further processing.