Documentation: 1 Step - Sanction List Check

Overview

This document explains how the `I. Step - Sanction List Check` script works. The script is designed to compare a list of company names with official sanction lists to identify potential matches.

Purpose

The main goals of this script are:

- Download and process official sanction lists.
- Extract and combine unique names from these lists.
- Compare company names with sanctioned names using a similarity check.
- Save the results for further analysis.

Requirements

To run this script, you need:

- **Python 3.x**
- **Libraries:**
  - `pandas`
  - `fuzzywuzzy`
  - `openpyxl`

You can install the required libraries with this command:

pip install pandas fuzzywuzzy openpyxl

How It Works

Step 1: Download and Combine Sanction Lists

What happens in this step:

1. The script downloads two official sanction lists from these websites:
    - List 1
    - List 2
2. It extracts unique names from each list.
3. It combines these names into one list and removes duplicates.
4. The combined list is saved as `sanction_list.csv`.

Code Example:

```
import pandas as pd

# URLs for the CSV files
url1 = 'https://www.treasury.gov/ofac/downloads/sdn.csv'
url2 = 'https://www.treasury.gov/ofac/downloads/consolidated/cons_alt.csv'

# Read data from the first CSV file
df1 = pd.read_csv(url1, on_bad_lines='skip')
sanction_list_url1 = df1.iloc[:, 1].dropna().unique()

# Read data from the second CSV file
df2 = pd.read_csv(url2, on_bad_lines='skip')
sanction_list_url2 = df2.iloc[:, 3].dropna().unique()

# Combine and save names
sanction_list = list(set(sanction_list_url1) | set(sanction_list_url2))
sanction_list_df = pd.DataFrame({'Sanctioned Names': sanction_list})
sanction_list_df.to_csv('sanction_list.csv', index=False)

print("Sanctioned names saved to 'sanction_list.csv'")
```

Step 2: Compare Company Names with Sanction List

What happens in this step:

1. The script reads company names from a file named `BELGIUM_companies.csv`.
2. It compares each company name with names from the `sanction_list.csv` file.
3. The comparison uses an approximate matching method to account for minor differences in spelling.
4. Results are saved in a new file called `Step_1_evaluated_companies.xlsx`.

Code Example:

```
import pandas as pd
from fuzzywuzzy import fuzz

# Load files
```

```
companies_df = pd.read_csv('BELGIUM_companies.csv', low_memory=False,
encoding='utf-8')
sanction_list_df = pd.read_csv('sanction_list.csv')

# Prepare names
company_names = companies_df['OriginalCompanyName'].str.lower()[0:500].tolist()
sanctioned_names = sanction_list_df['Sanctioned Names'].str.lower().tolist()

def approximate_match(name, sanctioned_names, threshold=85):
    """Check if a name closely matches any sanctioned name."""
    name = name.lower()
    for sanctioned_name in sanctioned_names:
        similarity = fuzz.ratio(name, sanctioned_name)
        if similarity >= threshold:
            return 46
    return 0

# Apply matching
companies_df['Score_Step_1'] = companies_df['OriginalCompanyName'].apply(
    lambda name: approximate_match(name, sanctioned_names)
)

# Save results
companies_df.to_excel('Step_1_evaluated_companies.xlsx', index=False)

print("Results saved to 'Step_1_evaluated_companies.xlsx'")
```

Key Terms

- **Sanction List:** A list of individuals, companies, or entities that are restricted by law.
- **Approximate Matching:** A method to compare two names even if they are not exactly the same.
- **Threshold:** A similarity score (e.g., 85%) used to determine if two names match.

Output Files

- `sanction_list.csv`: A combined list of sanctioned names.
- `Step_1_evaluated_companies.xlsx`: A file showing which company names matched the sanction list.

Limitations

- Only the first 500 company names are processed.
- Matching accuracy depends on the threshold value.

Improvements for the Future

- Support for processing larger datasets.
- Add error handling and logging for better debugging.
- Improve performance with parallel processing.

Author

- **Name:** Agita Ferstere
- **Date:** 03/12/2024

End of Documentation