International Journal of Enhanced Research in Science, Technology & Engineering
ISSN: 2319-7463, Vol. 14 Issue 3, March-2025

Quantitative Analysis of UPR Narratives on Severe Human Rights Violations

Farida Bouattoura

ABSTRACT

This dataset analyzes 618 Universal Periodic Review (UPR) Stakeholder Summary Reports, focusing on severe human rights violations such as genocide and crimes against humanity. Using a specialized dictionary constructed from international legal texts (Rome Statute, R2P, UDHR), the dataset supports quantitative, comparative, and exploratory analysis. Validation was conducted through expert feedback surveys and AI cross-referencing to ensure accuracy and alignment with legal standards. Statistical outputs, including chi-square tests and descriptive metrics, provide insights into narrative patterns across nations and sessions. The dataset is a robust tool for mitigating bias and enhancing human rights research.

DICTIONARY CONSTRUCTION PROCESS

The construction of the dictionary, word_dict, aimed at encapsulating serious human rights violations and international justice mechanisms, was methodically rooted in essential international legal texts and principles. First, core documents were identified, including the Rome Statute for legal definitions of international crimes, the Responsibility to Protect (R2P) doctrine for obligations on responding to atrocities, and the Universal Declaration of Human Rights for outlining fundamental freedoms. Relevant terms were meticulously extracted from these texts to ensure alignment with legal and human rights standards, covering categories such as specific types of violations, victimization processes, legal frameworks, and remedies for justice and accountability. Extracted terms were categorized to link crimes with legal responses and enforcement mechanisms. Following this, an interim peer review involved experts in international law scrutinizing the dictionary for accuracy, gaps, and relevance, while AI cross-referencing compared terms with databases, publications, and legal documentation to ensure comprehensive representation and alignment with current standards. Based on iterative feedback and machine validation, refinements were made to the dictionary by adding new terms, clarifying definitions, and removing inaccuracies, ensuring it was precise, exhaustive, and relevant for its intended application in human rights analysis.

Dictionary Design

Creating a comprehensive table that categorizes all the words from the provided dictionary into three main documentation categories—Responsibility to Protect (R2P), Rome Statute, and Universal Declaration of Human Rights (UDHR)—is a detailed task due to the broad and overlapping nature of these frameworks. Each term can fit into multiple categories because these documents share common goals related to preventing atrocities, protecting human rights, and ensuring justice. Below is an attempt to categorize these terms based on their most relevant associations with each of the three documents. Please note, many terms are applicable across all categories due to their interconnected nature in addressing human rights and crimes against humanity.

This categorization is not exhaustive and, as noted, many terms could be relevant to more than one category due to the wide-ranging scope of these documents. The overlap reflects the comprehensive approach needed to address complex human rights issues globally.

Here are the words listed in alphabetical order:

"abolish", "abduction", "abused", "against", "alarmed", "all forcibly closed", "annihilation", "arbitrary", "arbitrary detention", "armed militia", "army tribunals", "arrest", "atrocity", "attack", "authorization", "authorities", "authority", "civil war", "civilian in military court", "color", "compulsory impregnation", "corporal punishment", "crimes", "crimes against humanity", "crimes against peace", "crimes of terrorism", "criminalization", "criminalized gendered based violence",

"crisis", "cruel", "cruel inhuman", "cultural", "death penalty", "degrading", "demonizing resort to charges", "detention", "deportation", "discriminate", "discriminatory", "disfigurement", "disproportionately", "domestic violence", "enforced disappearances", "enforced sterilization", "enforcement action", "enslavement", "enslaved", "ethnic", "ethnic cleansing", "evidence", "excessive violence", "exclude", "excluding", "executions", "extermination", "extrajudicial executions", "extrajudicial killings", "extremist", "firearms", "forced", "forced pregnancy", "foreign", "gender", "gender based violence", "gender equality", "gendered based discrimination", "gross", "gross violations", "hatred", "high number of cases", "human trafficking", "humanity", "hurt", "imposed", "imprisonment", "incriminate", "indoctrination", "indirectly", "infringe", "inflict", "instruments", "integrating military training", "interferences", "international humanitarian law", "international obligations", "investigation", "judgment of death", "kidnappings", "killing", "language", "large number", "limited freedoms", "looting", "mass murder", "military tribunals", "minority", "misappropriations", "multiple", "murder", "national", "national convention for protection of all persons with enforced disappearances", "neglect", "no", "no access", "no hope", "no national plan", "no regard", "no rough consideration", "numerous", "obliteration", "organization", "overcrowded", "people smuggling", "political", "political instability", "political or other opinion", "political prisoners", "poor quality", "population", "prevent", "protestors", "protect", "punishment", "purpose", "race", "racial", "racial purification", "rape", "ratify", "recruitment of children", "refusal to acknowledge freedom", "refused", "refused", "religion", "religious", "restrictions", "responsibility", "rome statute", "servitude", "sex", "sharp", "slave", "slave trade", "slavery", "sovereignty", "state", "state of emergency", "steadily deteriorating", "sterilization", "strongly condemn", "support", "systematically", "terrorism", "terrorist", "terrorist acts", "threat", "torture", "trafficking", "treatment", "truth-seeking", "violations", "violations international human rights law", "violence", "war crimes", "war of aggression", "weapons", "widespread", "women"

Dictionary Validity

The validation of the specialized dictionary, designed for quantitative and comparative analysis of narratives related to severe human rights violations, employed a mixed-method approach combining text analysis, expert feedback, and AI cross-referencing. Two rounds of interim feedback were conducted; in the first, experts and academics provided general input during its construction, followed by a structured survey with binary, multiple-choice, and open-ended questions addressing the dictionary's precision, effectiveness, and opportunities for improvement. Participants included academics, professors, and human rights defenders across fields like international law and humanitarian aid, whose insights were incorporated to refine terms, address gaps, and bolster alignment with objectives. Simultaneously, AI cross-referencing was utilized to compare the dictionary's terms with established databases, publications, and legal texts to ensure accuracy and compliance with contemporary linguistic standards. Focused on analyzing narratives of genocide, war crimes, crimes against humanity, and extreme violations referenced by the ICC and ICJ, the dictionary is not designed to allege violations but to systematically mitigate bias through objective criteria, supporting qualitative analysis without supplanting it. Its purpose is exploratory and relational, aiming to evaluate narratives within UN documentation against international legal standards, with limitations acknowledged in text-based analysis and the need for evidence to substantiate the severity of allegations, thus emphasizing critical thinking as integral to its use.

Data Analysis Tools and Libraries

Using Python for the extraction, analysis, and visualization of data from PDF reports involves a multi-step process that leverages various libraries within the Python ecosystem. Here's an overview of how each part of the process can be implemented in Python, specifically using Jupyter notebooks in the Jupiter Cloud environment. The libraries mentioned include filecmp, shutil, NLTK (Natural Language Toolkit), SciPy, collections, Seaborn, Matplotlib, NumPy, pandas, and more. This workflow outlines the steps from extracting text from PDFs to performing statistical analysis and visualization.

1. Extracting Text from PDF Reports

Libraries: For handling PDF files and extracting text, you might use libraries like PyPDF2 or pdfminer.six. These are not explicitly listed but are essential for this step.

```
from PyPDF2 import PdfReader

def extract_text_from_pdf(pdf_path):
reader = PdfReader(pdf_path)
text = ''
for page in reader.pages:
text += page.extract_text() + '\n'
return text
```

2. Saving Extracted Text as .txt Files

Libraries: shutil for file operations.

```
def save_text_to_file(text, file_path):
with open(file_path, 'w', encoding='utf-8') as file:
file.write(text)
```

3. Data Cleaning

Libraries: nltk.corpus, stopwords, string for natural language processing tasks.

```
import nltk
from nltk.corpus import stopwords
import string
nltk.download('stopwords')
stop_words = set(stopwords.words('english') + list(string.punctuation))
def clean_text(text):
words = nltk.word_tokenize(text)
```

```
cleaned_text = [word for word in words if word.lower() not in stop_words]
return ' '.join(cleaned_text)
```

4. Dictionary Comparative Analysis

Libraries: collections.Counter for frequency analysis.

```
from collections import Counter
def compare_with_dictionary(text, dictionary_terms):
word_counts = Counter(text.split())
relevant_terms = {word: count for word, count in word_counts.items() if word in dictionary_terms}
return relevant_terms
```

5. Statistical Analysis and Visualization

Libraries: scipy.stats, seaborn, matplotlib, numpy, pandas for statistical analysis and data visualization.

```
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
from scipy import stats
# Assuming `data` is a DataFrame with your cleaned text and relevant metrics
sns.set(style="whitegrid")
# Example visualization - Distribution plot
sns.displot(data['metric_of_interest'], kde=True)
plt.title('Distribution of Metric')
plt.xlabel('Metric')
plt.ylabel('Frequency')
# Statistical Analysis Example
result = stats.ttest_1samp(data['metric_of_interest'], popmean=expected_mean_value)
print(f"Test statistic: {result.statistic}, p-value: {result.pvalue}"
```

Data Selection

For this study, it is vitally essential to create foundational solid criteria of quantitative comparative, directing for a full population analysis of the reports. Study accounts for the UPR in its fourth cycle with a total of 618 UPR reports (data cutoff of May 2023), with the previous three accounting for 193 each, deducting Ukraine's completed session leaves a total of 618 UPR Stakeholders summary reports, and account and South Sudan during its session hence one account is combined with Sudan. To quantify future comparative for sound inferences and conclusions, the utilization of the entire population produces the most accuracy (Office of the United Nations High Commissioner for Human Rights, n.d.).

Data Collection

Before extraction, there are multiple methods for conducting web mining; one can use a batch downloader software or browser extension such as JDownloader, or if the researcher has some grasp of coding, python can be used to construct a code to extract all the files for the study. The files can be found in the OHCHR website through documentation by nations, where all three UPR-generated reports are accounted for for all sessions and cycles. However, for this study, we are concerned with the stakeholders' summary report generated by OHCHR from NGOs and other non-governmental actors with consultative status (ECOSOC). One can even manually extract them from

https://www.ohchr.org/en/hr-bodies/upr/documentation. However, that would be efficient for 620 documents. This study aims to utilize Python and Jupiter cloud programming language for a notebook to extract files and rename all files by extracting the identifiers from the PDFs. The Idefinifers in the reports are crucial to identifying the nation in question and the session, providing the time scope accounted for.

For such a technique to be applied, the URL pattern must be identified to account for all URL segments to be specific and exact in our extraction. Providing this study needed reports, we can utilize a few URLs to identify the name construction and segments given:

1	07 April 2008 - 18 April 2008	Finland	"http://daccess-ods.un.org/access.nsf/get?open&DS=A/HRC/WG.6/1/ FIN/3&Lang=E",
1	07 April 2008 - 18 April 2008	United Kingdom	"http://daccess-ods.un.org/access.nsf/get?open&DS=A/HRC/WG.6/1/ GBR/3&Lang=E",

"http://daccess-ods.un.org/access.nsf/get?open&DS=" is the main page, "A/HRC/WG.6/" indicates that it is from OHCHR. The Office of the United Nations High Commissioner for Human Rights (OHCHR) is a department of the Secretariat of the United Nations that works to promote and protect the human rights that are guaranteed under international law and stipulated in the Universal Declaration of Human Rights of 1948. "1/FIN/3" is the session number, followed by the three-letter national code, and the number three indicates that it is the Summary report (one of three

documents prepared for the UPR). The final segment, "&Lang=E," indicates the language version of the report in question, with E in our case for the English language.

```
# Replace with a list of URLs
urls = [ "http://daccess-ods.un.org/access.nsf/get?open&DS=A/HRC/WG.6/1/ARG/3& Lang=E",
 "http://daccess-ods.un.org/access.nsf/get?open&DS=A/HRC/WG.6/1/BHR/3& Lang=E", "http://daccess-
 ods.un.org/access.nsf/get?open&DS=A/HRC/WG.6/1/BRA/3& Lang=E", "http://daccess-
 ods.un.org/access.nsf/get?open&DS=A/HRC/WG.6/1/CZE/3& Lang=E", "http://daccess-
 ods.un.org/access.nsf/get?open&DS=A/HRC/WG.6/1/DZA/3& Lang=E",]
for i, url in enumerate(urls):
    response = requests.get(url)
    with open(f'document_{i+1}.pdf', 'wb') as f:
        f.write(response.content)
    print(f"Downloaded file {i+1}")
```

The pattern can be repeated using Google Sheets and the "Power Tool" sheets add-on. This can split data, combine, duplicate, and manipulate data and format.

Once extracted the following is an example of the product "

File number	Year
G0910962	2009
G0910975	2009
G0910984	2009

Renamed From Original Number to File Identifier Code
Renamed first/G0910962.pdf to A_HRC_WG.6_5_CHL_3.pdf
Renamed first/G0910975.pdf to WG.6_5_YEM_3.pdf
Renamed first/G0910984.pdf to A_HRC_WG.6_5_BLZ_3.pdf

These files provide the years, but are not easily identifiable, so there is a need to rename the files with the identifier within the URL, as it provides the session and in turn, the time scope and the national code. This also serves to validate further that the correct document is collected and named, as the programming used to rename these files shall extract the UN document identifier extracted from the document itself. The following code will be used to rename the files accordingly:

Text Renaming

```
import os
from pdfminer.high_level import extract_text
import re
# Directory containing your PDFs
directory = "stuff" # Change to your actual path try:
for filename in os.listdir(directory):  if
filename.endswith(".pdf"):  full_path =
os.path.join(directory, filename)
 # Extract all text from the PDF file
text = extract_text(full_path)
# Split the text into pages
pages = text.split('\f')
# Get the last two pages of the document
last_pages = pages[-2:]
identifier = None
for page in last_pages:
    # Search for the pattern in the text match =
    re.search(r"A/HRC/WG.\d+/\d+/[A-Z]+/\d+",
    page) # Update regex if necessary
    # If we found a match, use it as the identifier
    if match:
```

```
identifier = match.group(0)
            break
    # If we found an identifier, rename the file
    if identifier:
        # Replace slashes with underscores
        identifier = identifier.replace("/", "_")
        new_filename = f"{identifier}.pdf"
        new_full_path = os.path.join(directory,
        new_filename)
        os.rename(full_path, new_full_path)
        print(f"Renamed {full_path} to {new_full_path}")
    except KeyboardInterrupt:
        print("Keyboard interrupt detected. Stopping the
        execution.")
```

Example of produce:

Using the NLTK library for text cleaning and os library for file handling:

Text cleaning

```
import os
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize,
sent_tokenize
from nltk.stem import WordNetLemmatizer
import string
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('wordnet')
        print(f'Cleaned text for report {filename}:
```

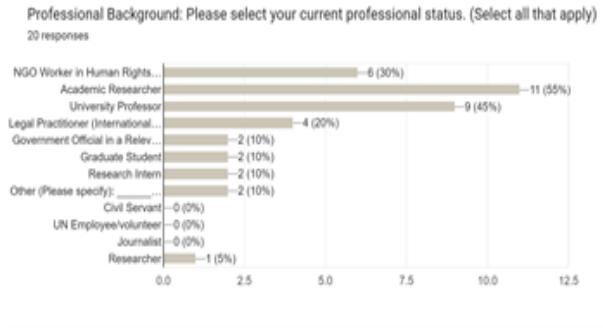
```
# Text cleaning function
def clean_text(text):
    # Convert the text into lowercase
    text = text.lower()
    # Remove punctuation
    text = text.translate(str.maketrans('', '',
    string.punctuation))
    # Tokenize the text
    tokens = word_tokenize(text)
    # Remove stopwords
    stop_words = set(stopwords.words('english'))
    tokens = [token for token in tokens if token not in
    stop_words]
    # Lemmatization
    lemmatizer = WordNetLemmatizer()
    tokens = [lemmatizer.lemmatize(token) for token
    in tokens]
    return tokens
# Specify the directory where the reports are located
directory = '/path/to/your/reports'
# Iterate over each file in the directory
for filename in os.listdir(directory):
    if filename.ends with('.txt'): # Change this if your
    reports are in another format
        with open(os.path.join(directory, filename), 'r')
        as f:
            text = f.read()
            cleaned_text = clean_text(text)

            {cleaned_text}')
```

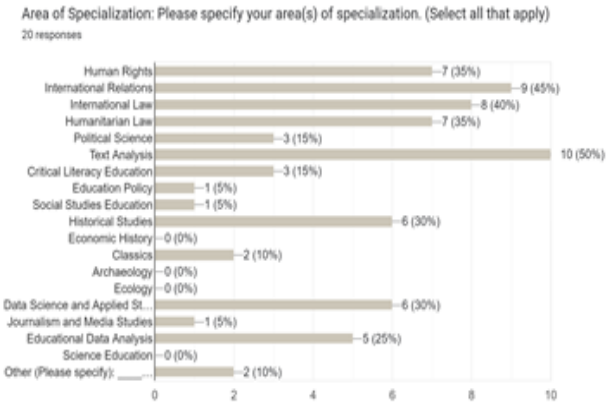
This script will iterate over each text file in the specified directory, apply the clean_text function to the file's content, and then print the cleaned text. Please replace /path/to/your/reports with the path to the directory where your reports are located. Also, this script assumes that the reports are text files (.txt). If they are in a different format, one should change .txt to the correct file extension. In the rapidly evolving domain of data science, text processing and file handling have emerged as crucial tasks. This analytical report critically examines the validity and reliability of various tools employed for these tasks, particularly emphasizing the Natural Language Toolkit (NLTK) library for text cleaning and the OS library for file handling.

VALIDATION AND RELIABILITY

Dictionary Refinement and Validation

The Dictionary feedback survey engaged 20 subject matter experts. Among these respondents, six were affiliated with NGOs and human rights advocacy; 11 were academic researchers, nine held positions as university professors, and various other roles as depicted in Survey Figure 1. The second survey figure reveals their fields of expertise; text analysis was the predominant specialization, chosen by 10 out of the 20 experts (50%). International relations were selected by 9 experts, international law by 8, humanitarian law and human rights each by 7, data science and applied statistics by 6, and educational analysis by 5.





HYPOTHESIS TESTING

In the execution of the Chi-Square Test of Independence through Python within a Jupyter Notebook environment, the analysis produced critical statistical outputs that informed the study's conclusions. The contingency table constructed from the UPR Stakeholder Summary reports and the predefined dictionary of legal terms facilitated this rigorous statistical testing. The calculated mean of occurrences of specified legal terms across reports was 33.14239482200647, with a standard deviation of 7.993268735247742. The confidence intervals for the mean, calculated at 95% and 99% confidence levels, were (32.51270426467293, 33.77208537934002) and (32.314841151356546, 33.9699484926564), respectively.

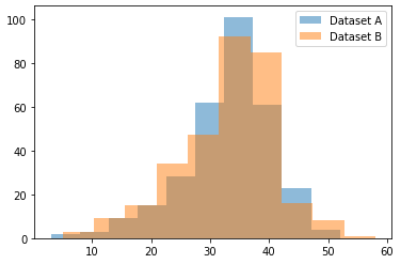
The pivotal outcome of the Chi-square statistic stood at 1077.1196733667223, accompanied by a p-value of 1.6443350286089006e-27. This p-value, significantly below the 0.05 threshold, mandates the rejection of the null hypothesis at both 95% and 99% confidence levels. Such a result unequivocally indicates a significant difference in the coverage of legal terminologies related to crimes against humanity and genocide across different UPR Stakeholder Summary reports.

Mean: 33.14239482200647
Standard Deviation: 7.993268735247742
95% Confidence Interval for the Mean: (3.251270426467293e+01, 33.77208537934002)
99% Confidence Interval for the Mean: (32.314841151356546, 33.9699484926564)
Chi-square Statistic: 1077.1196733667223, P-value: 1.6443350286089006e-27
Reject the null hypothesis at 95% confidence level - *There is a significant difference.*
Reject the null hypothesis at 99% confidence level - *There is a significant difference.*

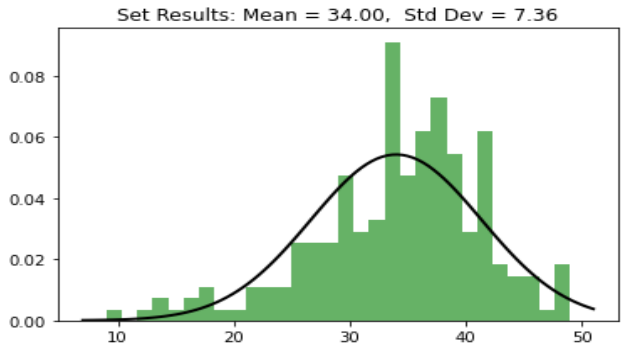
VALIDATION

Method 1
The model performs well across multiple validation techniques (random split, multiple random sets, and k-fold cross-validation

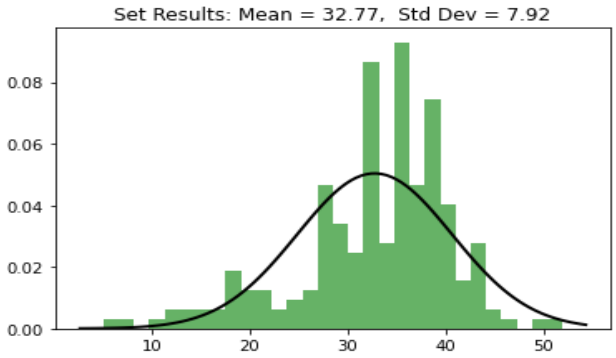
Dataset A - Mean: 33.396103896103895, Median: 35.0, Std: 7.796451766351777
Dataset B - Mean: 32.89032258064516, Median: 34.0, Std: 8.176337015602765



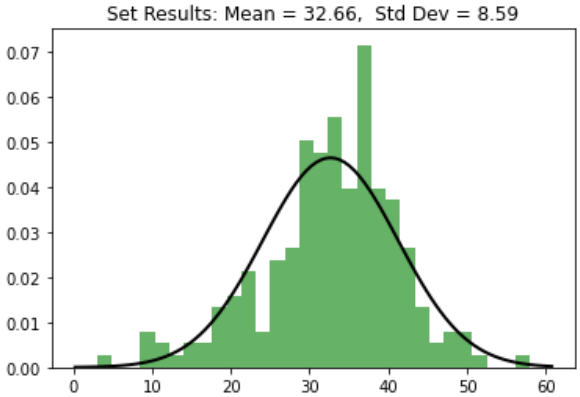
Method 2 Validation
Processing Set 1:
Number of files: 206
Mean: 34.00485436893204
Median: 35.0
Lower Quartile: 30.0
Upper Quartile: 39.0
Standard Deviation: 7.356060557155518
Variance: 54.11162692053916
Min Value: 9
Max Value: 49



Processing Set 2:
Number of files: 206
Mean: 32.76699029126213
Median: 34.0
Lower Quartile: 29.0
Upper Quartile: 38.0
Standard Deviation: 7.919134849647153
Variance: 62.71269676689604
Min Value: 5
Max Value: 52



Processing Set 3:
Number of files: 206
Mean: 32.65533980582524
Median: 34.0
Lower Quartile: 28.25
Upper Quartile: 38.0
Standard Deviation: 8.586659648178577
Variance: 73.73072391365822
Min Value: 3
Max Value: 58



Summary of Method 2

Validation Processing Set 1:	Validation Processing Set 2:	Validation Processing Set 3:
Number of files: 206 Mean: 34.00485436893204 Median: 35.0 Lower Quartile: 30.0 Upper Quartile: 39.0 Standard Deviation: 7.356060557155518 Variance: 54.11162692053916 Min Value: 9 Max Value: 49	Number of files: 206 Mean: 32.76699029126213 Median: 34.0 Lower Quartile: 29.0 Upper Quartile: 38.0 Standard Deviation: 7.919134849647153 Variance: 62.71269676689604 Min Value: 5 Max Value: 52	Number of files: 206 Mean: 32.65533980582524 Median: 34.0 Lower Quartile: 28.25 Upper Quartile: 38.0 Standard Deviation: 8.586659648178577 Variance: 73.73072391365822 Min Value: 3 Max Value: 58

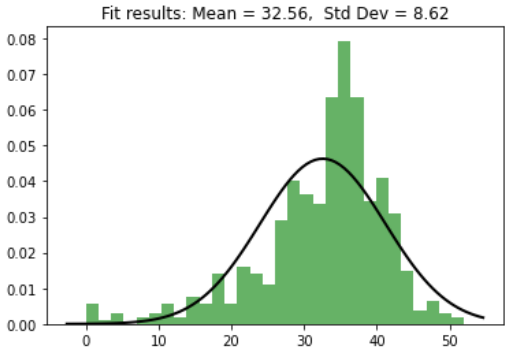
Method 3 Validation

The dataset was subjected to a detailed validation process through 5-Fold Cross-Validation, aiming to

Fold 1	Fold 2	Fold 3	Fold 4	Fold 5
Training Set	***Training Set***	***Training Set***	***Training Set***	***Training Set***
Mean word count: 33.093 Median word count: 34.0 Variance: 65.983 Standard Deviation: 8.123 Minimum word count: 3 Maximum word count: 58 First Quartile (Q1): 29.0 Third Quartile (Q3): 39.0	Mean word count: 33.328 Median word count: 34.0 Variance: 61.454 Standard Deviation: 7.839 Minimum word count: 3 Maximum word count: 58 Q1: 29.0 Q3: 38.0	Mean word count: 33.259 Median word count: 35.0 Variance: 61.957 Standard Deviation: 7.871 Minimum word count: 7 Maximum word count: 52 Q1: 29.0 Q3: 38.0	Mean word count: 33.079 Median word count: 34.0 Variance: 68.235 Standard Deviation: 8.260 Minimum word count: 3 Maximum word count: 58 Q1: 29.0 Q3: 39.0	Mean word count: 32.954 Median word count: 34.0 Variance: 62.397 Standard Deviation: 7.899 Minimum word count: 3 Maximum word count: 58 Q1: 29.0 Q3: 38.0
Testing Set	***Testing Set***	***Testing Set***	***Testing Set***	***Testing Set***
Mean word count: 33.339 Median word count: 34.0 Variance: 56.502 Standard Deviation: 7.517 Minimum word count: 10 Maximum word count: 50 Q1: 29.0 Q3: 38.0	Mean word count: 32.403 Median word count: 34.0 Variance: 74.015 Standard Deviation: 8.603 Minimum word count: 12 Maximum word count: 49 Q1: 27.75 Q3: 38.0	Mean word count: 32.675 Median word count: 34.0 Variance: 72.500 Standard Deviation: 8.515 Minimum word count: 3 Maximum word count: 58 Q1: 28.0 Q3: 39.0	Mean word count: 33.395 Median word count: 35.0 Variance: 47.444 Standard Deviation: 6.888 Minimum word count: 7 Maximum word count: 52 Q1: 30.0 Q3: 37.0	Mean word count: 33.902 Median word count: 36.0 Variance: 70.269 Standard Deviation: 8.383 Minimum word count: 9 Maximum word count: 51 Q1: 29.5 Q3: 40.0

Initial Python Code run produced the following descriptive statistic on the population of total population of UN OHCHR UPR Stakeholders Summary Report (up to May 2023) attained the following descriptive statistics results:

Mean: 32.55825242718446
Median: 34.0
Lower Quartile: 29.0
Upper Quartile: 38.0
Standard Deviation: 8.61758596617989
Variance: 74.26278788450058
Min Value: 0
Max Value: 52



The Challenge Identified

The documents in question were associated with the United Nations Human Rights Council Working Group on the Universal Periodic Review, specifically from sessions 7, 18, and 31. These sessions involved countries with notable geopolitical significance and connections to the natural gas industry, particularly with Russia and France. The files were:

- A_HRC_WG.6_31_MCO_3c.txt
- A_HRC_WG.6_18_SVK_3.txt
- A_HRC_WG.6_18_KHM_3c.txt
- A_HRC_WG.6_7_IRN_3c.txt
- A_HRC_WG.6_7_QAT_3c.txt
- A_HRC_WG.6_7_KAZ_3c.txt

Despite being accessible and readable manually, these files were interpreted as empty by the Python script, indicating a discrepancy between manual readability and programmatic accessibility.

Investigative Process and Findings

Upon encountering the issue, a thorough investigation was undertaken, involving several steps:

1. **File Integrity Check**: Initial suspicions of file corruption or partial downloads were dispelled through manual verification, confirming the files were intact.

2. **Format Compatibility Examination**: The possibility of unrecognized file formats hindering proper processing was explored. Utilizing libraries like PyPDF2 and python-docx did not resolve the issue, suggesting that the problem lay beyond simple format incompatibility.

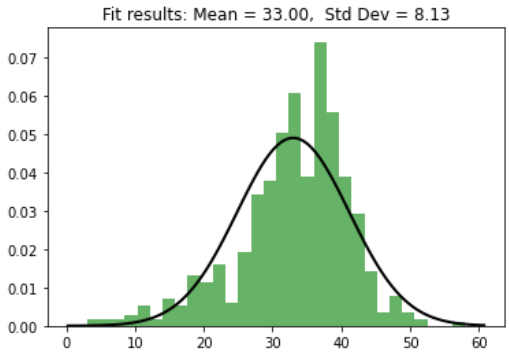
3. **Exploration of Hidden Characters/Metadata**: An in-depth search for hidden characters or excessive metadata yielded no findings that could explain the anomaly.

4. **Consideration of Protective Mechanisms**: Given the geopolitical sensitivity and the specific nature of the documents, it became apparent that protective mechanisms might be in place, preventing the script from accessing the content.

To circumvent the issue, the original PDFs were manually converted to Word documents and then re-inputted into the Python environment for text cleaning and processing. This approach proved successful, enabling the full functionality of the dataset for analysis.

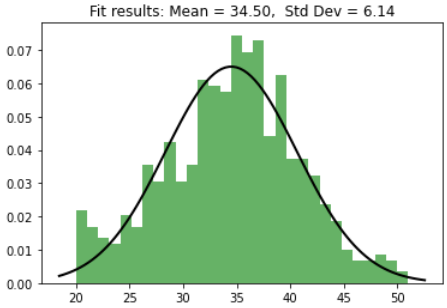
Second Python Code run produced the following descriptive statistic on the population of total population of UN OHCHR UPR Stakeholders Summary Report (up to May 2023) attained the following descriptive statistics results:

Mean: 33.14239482200647
Median: 34.0
Lower Quartile: 29.0
Upper Quartile: 38.0
Standard Deviation: 7.993268735247742
Variance: 63.89234507388905
Geometric Mean: 31.8615660586617
Min Value: 3
Max Value: 58



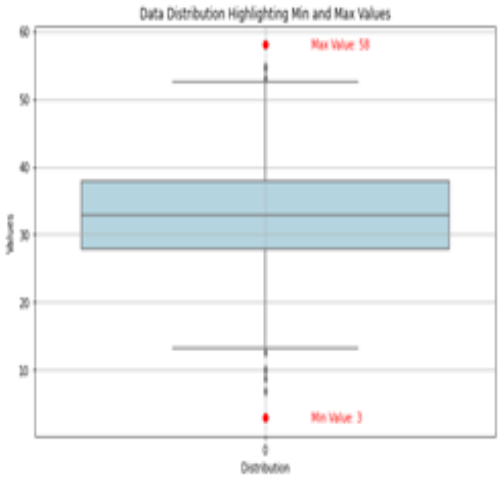
Third Python Code run produced the following descriptive statistic on the population of total population of UN OHCHR UPR Stakeholders Summary Report (up to May 2023) attained the following descriptive statistics results:

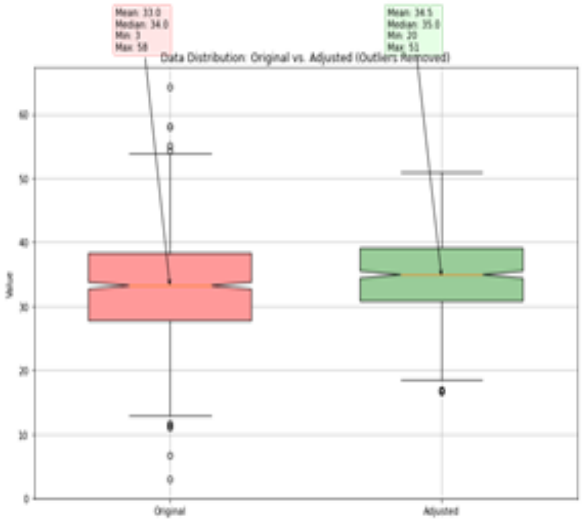
Number of files analyzed: 574
Mean: 34.49650349650349
Median: 35.0
Lower Quartile: 31.0
Upper Quartile: 39.0
Standard Deviation: 6.138693200350521
Variance: 37.683554208029726
Min Value: 20
Max Value: 51

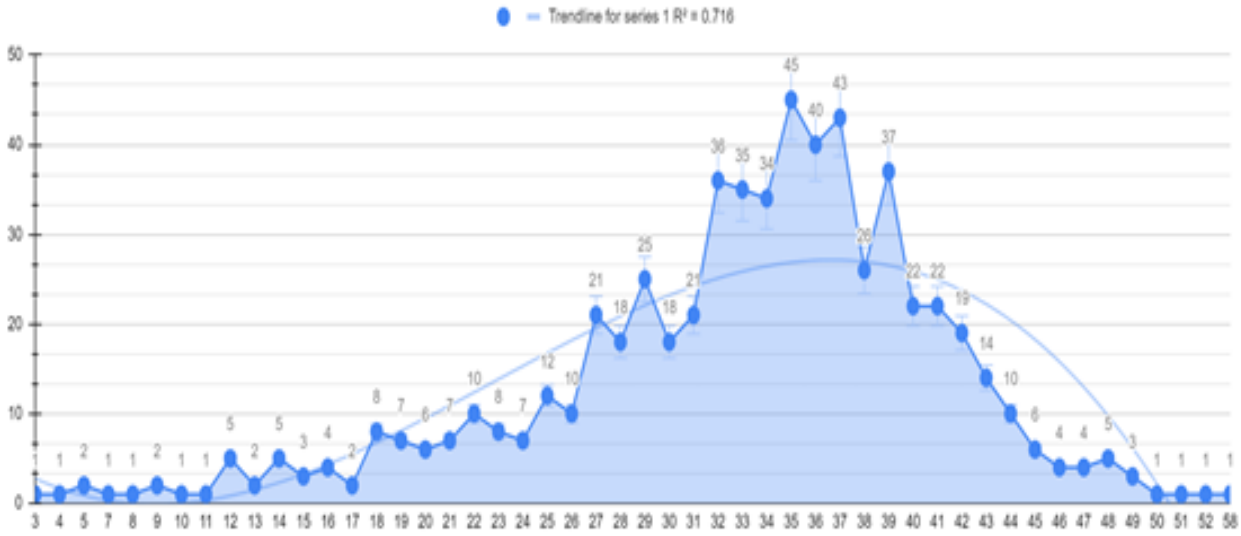


Files removed due to having a word count less than 20.Files removed due to being identified as outliers: ['A_HRC_WG.6_4_CMR_3c.txt', 'A_HRC_WG.6_38_MOZ_3c.txt'] were the low end, while interestingly on the High end it was A_HRC_WG.6_7_IRN_3c.txt, which required manual extract as the highest with 58. While Algeria's most recent report also was removed as an outlier with a 52.

Initial Python Code run	With manual extraction of the 6 unreadable files.	When removing the outliers (574)
Mean: 32.55825242718446 Median: 34.0 Lower Quartile: 29.0 Upper Quartile: 38.0 Standard Deviation: 8.61758596617989 Variance: 74.26278788450058 Min Value: 0 Max Value: 52	Mean: 33.003236245954696 Median: 34.0 Lower Quartile: 29.0 Upper Quartile: 38.0 Standard Deviation: 8.129413765302441 Variance: 66.08736816748882 Min Value: 3 Max Value: 58	Mean: 34.49650349650349 Median: 35.0 Lower Quartile: 31.0 Upper Quartile: 39.0 Standard Deviation: 6.138693200350521 Variance: 37.683554208029726 Min Value: 20 Max Value: 51







Score	Session	Nation	Score	Session	Nation	Score	Session	Nation	Score	Session	Nation
3	30	CPV	**29**	7	MDG	**34**	37	GEO	**38**	6	KHM
4	17	ISR	**29**	2	KOR	**34**	39	HUN	**38**	8	KGZ
5	16	CPV	**29**	30	TKM	**34**	40	ISL	**38**	14	UKR
5	23	STP	**29**	33	BTN	**34**	7	ITA	**38**	18	SVK
7	10	STP	**29**	6	BTN	**34**	1	BHR	**38**	27	GBR
8	10	KNA	**29**	9	MDV	**34**	16	CUB	**38**	3	SRB
9	19	BRN	**29**	10	EST	**34**	17	MEX	**38**	30	RUS

9	11	VCT	**29**	11	BEL	**34**	9	JAM	**38**	30	DEU
10	3	CPV	**29**	15	MNE	**34**	9	PAN	**38**	35	ESP
11	3	BRB	**29**	19	PRT	**34**	34	FJI	**38**	39	GRC
12	14	GAB	**29**	26	ISL	**34**	7	FJI	**38**	39	IRL
12	2	GAB	**29**	33	NOR	**34**	10	PRY	**38**	8	SWE
12	8	GRD	**29**	36	AND	**34**	28	ARG	**38**	8	ESP
12	11	PLW	**29**	15	BRB	**34**	3	COL	**38**	12	SYR
12	30	TUV	**29**	19	CRI	**35**	11	SOM	**38**	20	IRQ
13	10	LCA	**29**	37	LCA	**35**	12	ZWE	**38**	29	ARE
13	25	VCT	**29**	15	TON	**35**	16	CMR	**38**	33	NIC
14	28	GAB	**29**	18	URY	**35**	17	SEN	**38**	13	BRA
14	16	TUV	**29**	2	PER	**35**	21	KEN	**39**	10	NER
14	32	VUT	**29**	41	ECU	**35**	27	ZAF	**39**	17	NGA
14	36	MHL	**30**	1	ZAF	**35**	27	TUN	**39**	19	ETH
14	38	PLW	**30**	14	ZMB	**35**	38	MOZ	**39**	22	MWI
15	21	GNB	**30**	2	GHA	**35**	39	TZA	**39**	38	NER
15	25	ATG	**30**	20	AGO	**35**	41	ZAF	**39**	40	SSD
15	3	BHS	**30**	37	RWA	**35**	9	LBY	**39**	5	TCD
16	11	SYC	**30**	9	MWI	**35**	13	PHL	**39**	16	UZB
16	18	VUT	**30**	6	BRN	**35**	17	MYS	**39**	17	CHN
16	22	MHL	**30**	14	CHE	**35**	28	KOR	**39**	18	VNM
16	9	MHL	**30**	15	SRB	**35**	32	VNM	**39**	23	NPL
17	37	STP	**30**	2	CHE	**35**	5	VNM	**39**	33	BRN
17	9	AND	**30**	34	BIH	**35**	1	FIN	**39**	37	NPL
18	2	MLI	**30**	5	SVK	**35**	12	IRL	**39**	39	THA
18	5	COM	**30**	4	JOR	**35**	13	NLD	**39**	4	CHN
18	17	MCO	**30**	5	YEM	**35**	13	FIN	**39**	12	MDA
18	31	BLZ	**30**	36	PAN	**35**	16	AZE	**39**	12	LTU
18	6	DMA	**30**	42	GTM	**35**	18	MKD	**39**	2	FRA
18	37	FSM	**30**	43	BRB	**35**	20	BIH	**39**	21	ESP
18	5	VUT	**30**	39	PNG	**35**	21	ARM	**39**	24	DNK
18	25	SUR	**31**	25	SWZ	**35**	22	BGR	**39**	3	MNE
19	17	COG	**31**	26	TGO	**35**	25	HUN	**39**	34	ITA

19	15	LIE	**31**	35	GNB	**35**	26	MDA	**39**	4	DEU
19	25	TTO	**31**	6	GNQ	**35**	28	CHE	**39**	41	GBR
19	35	KIR	**31**	9	MRT	**35**	32	SVK	**39**	41	POL
19	8	KIR	**31**	26	TLS	**35**	34	SVN	**39**	43	FRA
19	35	GUY	**31**	3	TKM	**35**	40	MDA	**39**	5	MKD
19	7	BOL	**31**	4	AZE	**35**	43	MNE	**39**	18	YEM
20	32	COM	**31**	40	TLS	**35**	6	PRT	**39**	20	EGY
20	38	SYC	**31**	33	ALB	**35**	7	SVN	**39**	31	JOR
20	8	LSO	**31**	36	HRV	**35**	8	BLR	**39**	31	SAU
20	12	ATG	**31**	5	MCO	**35**	21	KWT	**39**	41	TUN
20	29	BRB	**31**	6	ALB	**35**	22	JAM	**39**	43	ARE
20	35	GRD	**31**	9	HRV	**35**	32	DOM	**39**	2	GTM
21	17	MUS	**31**	19	QAT	**35**	34	SLV	**39**	23	AUS
21	18	COM	**31**	33	QAT	**35**	43	BHS	**39**	18	CHL
21	19	BTN	**31**	16	CAN	**35**	7	SLV	**39**	41	BRA
21	22	AND	**31**	6	CRI	**35**	25	PNG	**40**	14	GHA
21	21	GRD	**31**	27	ECU	**35**	1	ARG	**40**	15	BDI
21	24	SLB	**31**	34	BOL	**35**	30	COL	**40**	22	LBR
21	11	SUR	**31**	42	PER	**35**	42	ARG	**40**	24	NER
22	16	DJI	**32**	1	TUN	**36**	2	BEN	**40**	39	SWZ
22	31	MUS	**32**	10	NAM	**36**	20	MDG	**40**	11	SGP
22	4	DJI	**32**	13	DZA	**36**	34	MDG	**40**	23	MMR
22	15	LUX	**32**	18	ERI	**36**	34	AGO	**40**	25	THA
22	37	KNA	**32**	19	COD	**36**	36	LBR	**40**	25	TJK
22	10	NRU	**32**	28	ZMB	**36**	38	SOM	**40**	31	MYS
22	2	TON	**32**	29	BWA	**36**	4	NGA	**40**	1	CZE
22	21	KIR	**32**	30	CMR	**36**	40	TGO	**40**	2	UKR
22	24	PLW	**32**	35	GIN	**36**	40	ZWE	**40**	21	TUR
22	9	FSM	**32**	42	BEN	**36**	41	MAR	**40**	29	FRA
23	14	BEN	**32**	42	ZMB	**36**	1	IND	**40**	35	SWE
23	15	MLI	**32**	5	COG	**36**	12	TLS	**40**	8	ARM
23	17	CAF	**32**	7	GMB	**36**	16	TKM	**40**	29	ISR
23	8	GNB	**32**	22	MNG	**36**	27	IND	**40**	3	ISR

23	3	LIE	**32**	11	DNK	**36**	27	PHL	**40**	22	USA
23	10	OMN	**32**	19	NOR	**36**	30	UZB	**40**	9	USA
23	29	BHS	**32**	24	LVA	**36**	33	PRK	**40**	5	NZL
23	23	FSM	**32**	24	BEL	**36**	36	MNG	**40**	27	BRA
24	16	BFA	**32**	31	MLT	**36**	42	PAK	**41**	11	SDN
24	21	LSO	**32**	32	MKD	**36**	1	GBR	**41**	24	SOM
24	7	SMR	**32**	32	CYP	**36**	10	AUT	**41**	26	UGA
24	5	BLZ	**32**	34	SMR	**36**	11	LVA	**41**	26	ZWE
24	3	TUV	**32**	37	AUT	**36**	12	ISL	**41**	33	COD
24	37	NRU	**32**	40	LTU	**36**	15	FRA	**41**	37	MRT
24	38	SLB	**32**	1	MAR	**36**	18	CYP	**41**	43	MLI
25	24	SYC	**32**	13	BHR	**36**	21	SWE	**41**	6	ETH
25	42	GAB	**32**	17	JOR	**36**	23	GEO	**41**	12	THA
25	29	LIE	**32**	9	LBN	**36**	25	GRC	**41**	13	IND
25	3	LUX	**32**	22	PAN	**36**	27	FIN	**41**	14	JPN
25	5	MLT	**32**	33	CRI	**36**	29	ROU	**41**	16	BGD
25	23	OMN	**32**	36	JAM	**36**	35	ARM	**41**	2	LKA
25	15	BHS	**32**	39	ATG	**36**	38	BEL	**41**	9	MNG
25	23	KNA	**32**	9	HND	**36**	38	DNK	**41**	16	RUS
25	33	DMA	**32**	14	PER	**36**	41	FIN	**41**	27	NLD
25	39	VCT	**32**	21	GUY	**36**	43	ROU	**41**	27	POL
25	11	SLB	**32**	38	PRY	**36**	9	BGR	**41**	8	TUR
25	23	NRU	**33**	10	RWA	**36**	7	EGY	**41**	32	YEM
26	28	GHA	**33**	12	UGA	**36**	22	HND	**41**	37	OMN
26	31	COG	**33**	13	MAR	**36**	10	AUS	**41**	36	HND
26	4	MUS	**33**	19	GNQ	**36**	12	VEN	**41**	4	MEX
26	1	POL	**33**	19	CIV	**37**	12	TZA	**42**	11	SLE
26	43	LIE	**33**	24	MOZ	**37**	13	ZAF	**42**	23	RWA
26	43	LUX	**33**	3	BFA	**37**	27	DZA	**42**	31	NGA
26	8	KWT	**33**	30	DJI	**37**	29	MLI	**42**	39	SDN
26	11	WSM	**33**	31	TCD	**37**	3	BDI	**42**	8	KEN
26	18	NZL	**33**	38	SLE	**37**	33	ETH	**42**	20	KAZ
26	29	TON	**33**	7	AGO	**37**	5	CAF	**42**	28	JPN

27	10	MOZ	**33**	3	UZB	**37**	6	COD	**42**	30	BGD
27	31	SEN	**33**	4	MYS	**37**	13	IDN	**42**	32	AFG
27	32	ERI	**33**	1	NLD	**37**	21	LAO	**42**	38	SGP
27	33	CIV	**33**	13	POL	**37**	22	MDV	**42**	41	PHL
27	38	NAM	**33**	16	DEU	**37**	24	SGP	**42**	41	IND
27	12	AFG	**33**	20	SVN	**37**	30	AZE	**42**	42	KOR
27	15	ROU	**33**	26	LTU	**37**	39	TJK	**42**	22	HRV
27	17	MLT	**33**	28	CZE	**37**	10	GEO	**42**	42	CHE
27	20	SMR	**33**	29	MNE	**37**	11	HUN	**42**	20	IRN
27	29	LUX	**33**	29	SRB	**37**	13	GBR	**42**	37	LBN
27	6	CYP	**33**	31	MCO	**37**	19	ALB	**42**	41	BHR
27	15	ARE	**33**	36	BLR	**37**	20	ITA	**42**	1	BRA
27	23	LCA	**33**	43	SRB	**37**	22	BLR	**43**	23	MRT
27	4	CUB	**33**	7	BIH	**37**	23	AUT	**43**	31	CAF
27	40	HTI	**33**	12	HTI	**37**	25	IRL	**43**	1	IDN
27	11	PNG	**33**	18	DOM	**37**	36	BGR	**43**	14	PAK
27	14	ARG	**33**	19	DMA	**37**	38	EST	**43**	2	JPN
27	24	PRY	**33**	20	SLV	**37**	41	NLD	**43**	31	CHN
27	39	SUR	**33**	30	CUB	**37**	42	CZE	**43**	35	KGZ
27	40	VEN	**33**	7	NIC	**37**	6	NOR	**43**	36	MDV
27	5	URY	**33**	1	ECU	**37**	1	DZA	**43**	41	IDN
28	12	TGO	**33**	16	COL	**37**	35	KWT	**43**	2	ROU
28	15	BWA	**33**	28	PER	**37**	4	SAU	**43**	17	SAU
28	28	BEN	**33**	32	URY	**37**	26	HTI	**43**	23	LBN
28	3	BWA	**34**	20	GMB	**37**	28	GTM	**43**	3	ARE
28	35	LSO	**34**	21	GIN	**37**	30	CAN	**43**	37	AUS
28	4	SEN	**34**	27	MAR	**37**	31	MEX	**44**	25	SDN
28	43	BWA	**34**	33	GNQ	**37**	6	DOM	**44**	10	NPL
28	14	CZE	**34**	36	MWI	**37**	20	FJI	**44**	2	PAK
28	38	LVA	**34**	4	CMR	**37**	32	NZL	**44**	27	IDN
28	13	TUN	**34**	40	UGA	**37**	39	WSM	**44**	34	KAZ
28	12	TTO	**34**	42	GHA	**37**	13	ECU	**44**	35	LAO
28	14	GTM	**34**	6	CIV	**37**	20	BOL	**44**	42	JPN

28	17	BLZ	**34**	8	GIN	**37**	26	VEN	**44**	35	TUR
28	19	NIC	**34**	9	LBR	**37**	32	CHL	**44**	27	BHR
28	39	TTO	**34**	12	TJK	**37**	5	CHL	**44**	34	EGY
28	25	WSM	**34**	14	KOR	**38**	17	TCD	**45**	22	LBY
28	43	TON	**34**	19	PRK	**38**	25	TZA	**45**	14	LKA
28	8	GUY	**34**	5	AFG	**38**	29	BDI	**45**	21	KGZ
29	12	SWZ	**34**	6	PRK	**38**	43	BDI	**45**	28	PAK
29	2	ZMB	**34**	8	LAO	**38**	6	ERI	**45**	4	RUS
29	24	NAM	**34**	24	EST	**38**	1	PHL	**45**	34	IRQ
29	30	BFA	**34**	28	UKR	**38**	10	MMR	**46**	26	SSD
29	34	GMB	**34**	33	PRT	**38**	32	KHM	**46**	35	KEN
48	18	KHM	**49**	7	KAZ	**47**	28	LKA	**46**	26	SYR
48	11	GRC	**50**	4	BGD	**47**	34	IRN	**46**	43	ISR
48	7	QAT	**51**	42	LKA	**47**	40	SYR	**49**	36	LBY
48	7	IRQ	**52**	41	DZA	**47**	4	CAN	**49**	37	MMR
			58	7	IRN	**48**	36	USA			