

# 基于关键词词频技术的气候风险指数

## 1. Idea

气候风险评估通常需要从多个维度（如关键词频率、情感倾向等）收集信息，而这些维度的信息往往具有不同的量纲和重要性。

关键词的出现频率反映了文本中与特定风险指标（如“洪水”、“碳排放”）相关的关注程度。频率越高，可能表明该风险在文本中被强调的程度越高，因此是风险评估的重要信号。

## 2. Datasets

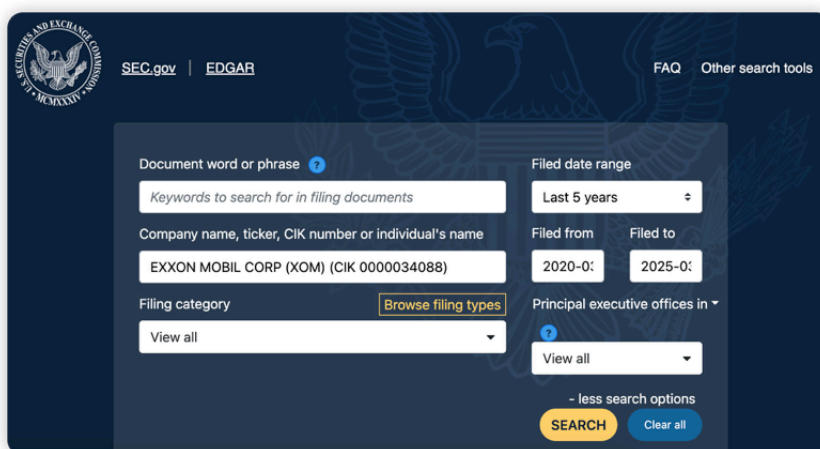
下载 NextEra Energy, Inc. (NEE) 的 10-Q 报告。NextEra Energy, Inc. (NEE) 是美国最大的可再生能源公司之一。在其 10-Q 报告中，NextEra Energy 可能会提及以下关键词：

- 可再生能源 (renewable energy)
- 太阳能发电 (solar power)
- 风能发电 (wind power)
- 能源转型 (energy transition)
- 减排目标 (emission reduction target)

该公司积极投资低碳技术，并可能披露与气候政策相关的风险，例如：

- 碳定价 (carbon pricing)
- 气候监管 (climate regulation)

我们可以开发代码，从 [SEC 的 EDGAR 数据库](#) 自动检索此类文件，通常这些报告以 **HTML 格式** 提供。



The image shows the SEC EDGAR search interface. The header includes the SEC logo, "SEC.gov | EDGAR", and links for "FAQ" and "Other search tools". The search form has several fields: "Document word or phrase" with a placeholder "Keywords to search for in filing documents"; "Company name, ticker, CIK number or individual's name" with "EXXON MOBIL CORP (XOM) (CIK 0000034088)" entered; "Filing category" with a dropdown menu showing "View all" and a "Browse filing types" link; "Filed date range" with a dropdown menu showing "Last 5 years"; "Filed from" and "Filed to" date pickers showing "2020-01" and "2025-01"; and "Principal executive offices in" with a dropdown menu showing "View all". At the bottom, there is a "SEARCH" button and a "Clear all" button.

10-Q 1 xom10q1q2020.htm FORM 10-Q

**UNITED STATES  
SECURITIES AND EXCHANGE COMMISSION**  
Washington, D.C. 20549

**FORM 10-Q**

☒ QUARTERLY REPORT PURSUANT TO SECTION 13 OR 15(d) OF  
THE SECURITIES EXCHANGE ACT OF 1934  
For the quarterly period ended March 31, 2020  
or  
☐ TRANSITION REPORT PURSUANT TO SECTION 13 OR 15(d) OF  
THE SECURITIES EXCHANGE ACT OF 1934  
For the transition period from \_\_\_\_\_ to \_\_\_\_\_  
Commission File Number 1-2256

**Exxon Mobil Corporation**  
(Exact name of registrant as specified in its charter)

New Jersey  
(State or other jurisdiction of  
incorporation or organization)

13-5409005  
(I.R.S. Employer  
Identification Number)

5959 Las Colinas Boulevard, Irving, Texas 75039-2298  
(Address of principal executive offices) (Zip Code)

## 3. Implementation

### 3.1 Extract text from HTML file

我们可以实现一个类似 `extract_text_from_html(html_path)` 的函数，该函数使用 UTF-8 编码读取 HTML 文件，利用 BeautifulSoup 进行解析，提取可读文本（去除标签和多余空白），处理错误，并返回一个字符串格式的文本，从而简化报告内容分析。

```
In [14]: from bs4 import BeautifulSoup
def extract_text_from_html(html_path):
    text = ""
    try:
        # Read the HTML file
        with open(html_path, 'r', encoding='utf-8') as file:
            soup = BeautifulSoup(file, 'html.parser')
            text = soup.get_text(separator=' ', strip=True)
    except Exception as e:
        print(f"Error reading HTML: {e}")
    return text
```

### 3.2 Keyword frequency counting

```
In [15]: climate_keywords = {
    "Extreme Weather Risk Index":
        ["flood", "hurricane",
         "drought", "extreme heat", "storm"],
    "Greenhouse Gas Emission\n Intensity Index":
        ["carbon emission", "greenhouse gas",
         "emission intensity", "carbon dioxide",
         "emission reduction target"],
    "Climate Adaptability and\n Resilience Index":
        ["climate resilience",
         "infrastructure upgrade",
         "emergency plan",
         "climate adaptation",
         "adaptation plan"],
    "Renewable Energy Investment\n and Transition Progress Index":
        ["renewable energy", "solar power",
```

```

        "wind power", "low-carbon technology",
        "energy transition"],
    "Policy and Legal Risk Index":
        ["carbon tax", "emission trading",
        "climate regulation", "policy uncertainty",
        "carbon pricing"]
}

```

考虑到这些词在每个季度的报告中，出现频率较低，所以将这些关键词结合在一起，它们共同反映了气候相关的风险、适应能力、政策影响和能源转型进程，可以概括为：综合气候风险与适应能力指数（Comprehensive Climate Risk and Adaptability Index, CCRA Index）这个指数可以用来衡量企业在气候变化相关领域的综合表现，既考虑风险因素，也涵盖应对和政策层面，适用于金融投资、政策评估和可持续发展研究。

```

In [16]: climate_keywords['Comprehensive Climate Risk\n and Adaptability Index'] =
    "flood", "hurricane", "drought", "extreme heat", "storm",
    "carbon emission", "greenhouse gas", "emission intensity",
    "carbon dioxide", "emission reduction target", "climate resilience",
    "infrastructure upgrade", "emergency plan", "climate adaptation",
    "adaptation plan", "renewable energy", "solar power", "wind power",
    "low-carbon technology", "energy transition", "carbon tax",
    "emission trading", "climate regulation", "policy uncertainty",
    "carbon pricing"
]

```

This function, `extract_keywords(text, keywords_dict)`, takes a text string and a dictionary of keywords, converts the text to lowercase, and counts the occurrences of each keyword (case-insensitive) within the text using regular expressions, returning a dictionary of keyword counts organized by indicator.

```

In [17]: import re
    from collections import Counter

    def extract_keywords(text, keywords_dict):
        text = text.lower()
        keyword_counts = {}

        for indicator, keywords in keywords_dict.items():
            counts = Counter()
            for keyword in keywords:
                keyword = keyword.lower()
                counts[keyword] = len(re.findall(r'\b' + re.escape(keyword) +
            keyword_counts[indicator] = counts
        return keyword_counts

```

### 3.3 Quantify indicators

```

In [18]: def quantify_indicators(keyword_counts, text,
    weight_keywords=0.7, weight_sentiment=0.3):
    indicators = {}
    for indicator, counts in keyword_counts.items():
        total_mentions = sum(counts.values())
        indicators[indicator] = total_mentions

```

```
return indicators
```

### 3.4 Visualize indicators

```
In [19]: import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from matplotlib_inline import backend_inline
backend_inline.set_matplotlib_formats('svg')

def plot_indicators(indicators, title, figsize=(6,4)):
    df = pd.DataFrame.from_dict(indicators, orient='index',
                                columns=['Score'])
    #df = df.sort_values(by='Score', ascending=False)
    plt.figure(figsize=figsize)
    sns.barplot(x=df.index, y='Score', data=df,
                palette='viridis', hue=df.index, legend=False)
    plt.xticks(rotation=45, ha='right', fontsize=9)
    plt.xlabel('Date')
    plt.title(title)
    plt.tight_layout()
    plt.show()
```

### 3.5 Exact date from 10-Q

```
In [20]: def extract_10_Q_date(text):
    date = text.split('For the quarterly period ended ')[-1].split(' Comm
    return date
```

### 3.6 Main function: Process a single report

```
In [21]: from tqdm.notebook import tqdm
import time
def process_report(html_path):
    with tqdm(total=4, desc=f"{html_path.split('/')[-1]}",
              bar_format="{l_bar}{unit}{bar}",
              unit="", disable=True) as pbar:

        # Step 1 Extracting HTML text
        text = extract_text_from_html(html_path)
        if not text:
            print("No text extracted from the HTML.")
            return
        pbar.update(1)

        # Step 2 Extracting keyword counts
        climate_keyword_counts = extract_keywords(text, climate_keywords)
        pbar.update(1)

        # Step 3 Quantifying climate indicators
        climate_indicators = quantify_indicators(climate_keyword_counts,
        pbar.update(1)
```

```
# Step 4 Extracting date
date = extract_10_Q_date(text)
pbar.update(1)
return date, climate_indicators
```

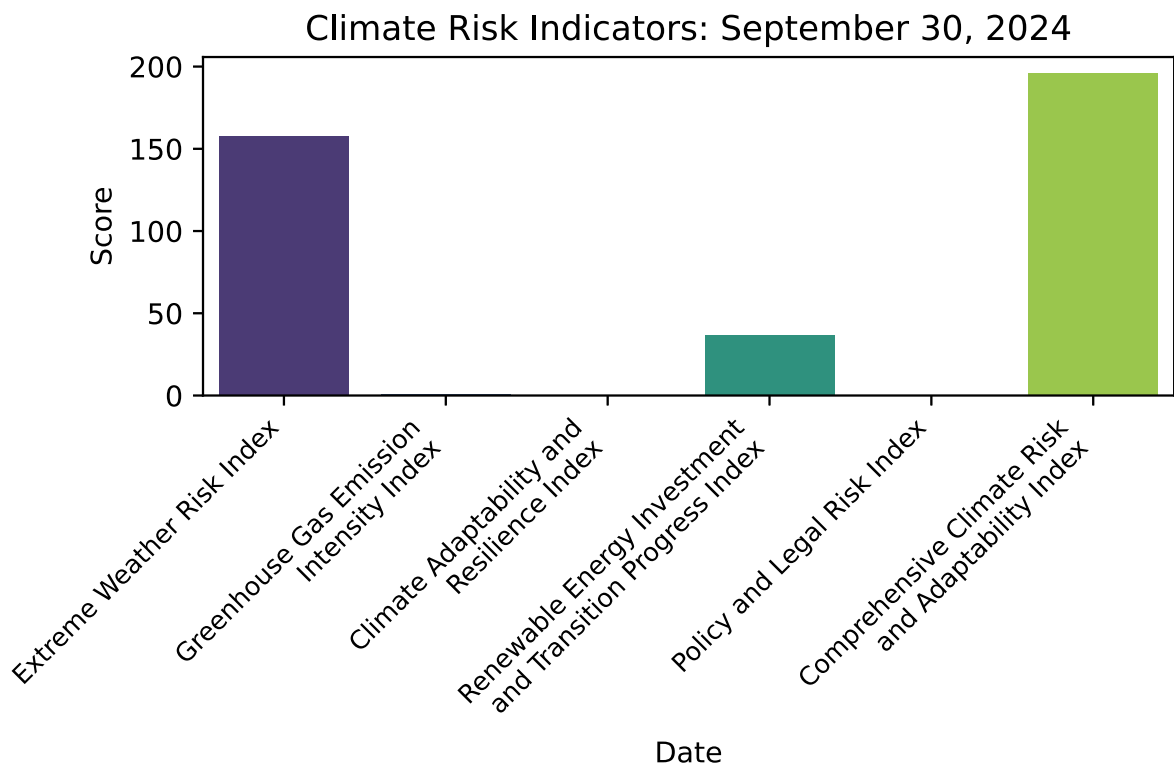
## 4. Test

```
In [22]: html_path = "datasets/sec-edgar-filings/NEE/10-Q-hm/full-submission-0000
date, climate_indicators = process_report(html_path)
```

```
In [23]: climate_indicators
```

```
Out[23]: {'Extreme Weather Risk Index': 158,
'Greenhouse Gas Emission\n Intensity Index': 1,
'Climate Adaptability and\n Resilience Index': 0,
'Renewable Energy Investment\n and Transition Progress Index': 37,
'Policy and Legal Risk Index': 0,
'Comprehensive Climate Risk\n and Adaptability Index': 196}
```

```
In [24]: plot_indicators(climate_indicators, f"Climate Risk Indicators: {date}")
```



这些关键词的频率反映了报告中对不同气候风险相关指数的关注程度。例如，极端天气风险指数和综合气候风险与适应能力指数的较高频率表明，这些指标在文本中得到了更广泛的讨论。相比之下，温室气体排放强度指数和政策与法律风险指数的较低频率表明，它们在报告中的强调程度较低。而气候适应性与韧性指数的零频率则意味着该指数在文本中完全未被提及。

## 5. Comprehensive Climate Risk and Adaptability Index

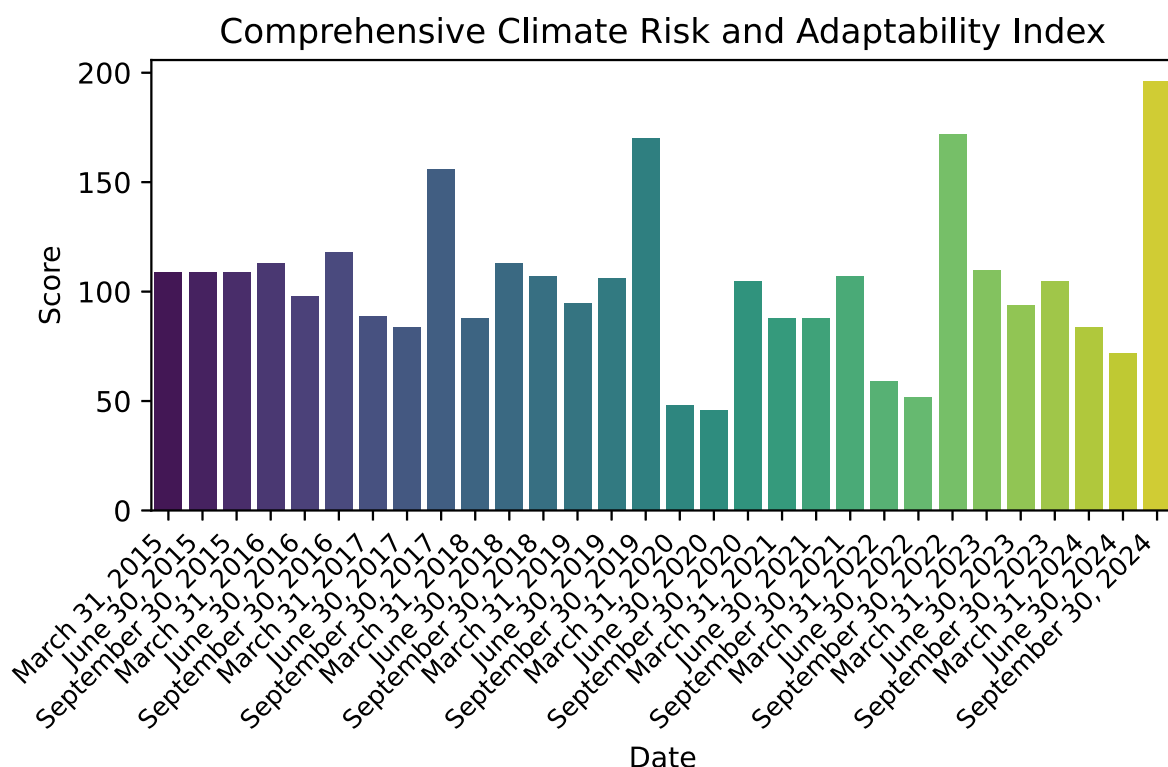
```
In [25]: import os

path = 'datasets/sec-edgar-filings/NEE/10-Q-htm/'
l = os.listdir(path)
l.sort()
EWR, GGEI, CAR, PEITP, PLR, CCRA = {}, {}, {}, {}, {}, {}

for i in tqdm(l, desc="Processing files", unit="file"):
    if 'full' not in i: continue
    html_path = os.path.join(path, i)
    date, climate_indicators = process_report(html_path)
    CCRA[date] = climate_indicators['Comprehensive Climate Risk and Ada
```

Processing files: 0%| | 0/31 [00:00<?, ?file/s]

```
In [26]: plot_indicators(CCRA, "Comprehensive Climate Risk and Adaptability Index")
```



从上图可以看出，NEE 在 2020 至 2024 年期间，每年 3 月和 6 月的 10-Q 报告中披露的气候风险相关关键词较少。这导致综合指数呈周期性下降。这是一个有趣的发现。

## 6. Disadvantages of Keyword Frequency Technique

在单个季度的 10-Q 报告中，由于缺少匹配的关键词，一些指数的数值可能为零，而其他指数则可能具有具体的数值。这凸显了基于关键词的索引方法的局限性——当报告使用同义词或语境上等价的表达方式而非预定义关键词时，这些内容可能不会被捕捉到，从而导致指数的偏差。

为缓解这一问题，可以结合高级自然语言处理（NLP）技术，例如语义相似性模型或上下文嵌入，以增强关键词识别能力，提高气候相关指数的准确性。

当然，更有效的方法是使用 BERT 和 GPT 等预训练模型。这些基于人工智能和深度学习的技

术已被证明在文本理解和推理方面表现卓越。通过利用其语义理解能力，可以更准确地识别相关内容，减少因关键词匹配局限性导致的指数偏差，从而提高气候相关指数的可靠性和有效性。