Phân tích dữ liệu sao kê đóng góp thiệt hại của bão Yagi

Objective:

Đây là một dự án nhỏ nhằm phân tích dữ liệu sao kê chuyển khoản MTTQ đợt bão Yagi tính từ ngày 1/9 - 10/9 thông qua tổng hợp và đánh giá tổng số tiền quyên góp, xu hướng quyên góp theo thời gian, mức độ đóng góp của từng cá nhân/tổ chức, và phân tích sự phân phối các khoản đóng góp.

Trong dự án này tôi sẽ thông qua Google Colab sử dụng Python để xử lý và trực quan hóa dữ liệu cũng như các khám phá mà tôi tìm được

Tổng quan dữ liệu

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200347 entries, 0 to 200346
Data columns (total 3 columns):
 #   Column  Non-Null Count   Dtype 
---  ------  --------------   ----- 
 0   date    200347 non-null  object
 1   credit  200347 non-null  int64 
 2   detail  200347 non-null  object
dtypes: int64(1), object(2)
memory usage: 4.6+ MB
```

Bộ dữ liệu của chúng ta sẽ có 200347 nghìn quan sát với 3 cột là thời gian, số tiền chuyển khoản và nội dung chuyển khoản

Tổng số tiền quyên góp.

Dưới đây sẽ là đoạn code về tổng số tiền được quyên góp

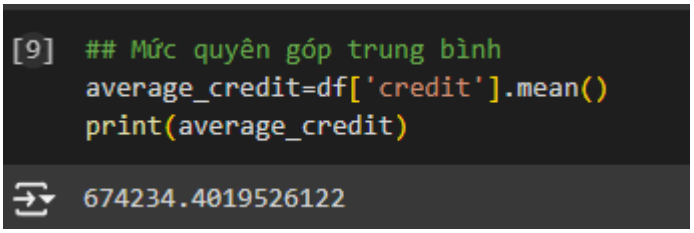
```
## Tổng số tiền quyên góp
total_credit=df['credit'].sum()
print(total_credit)
```

```
135080839728
```

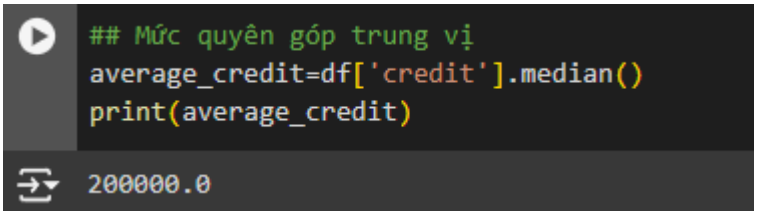
Kết quả cho thấy tổng số tiền quyên góp đạt **135 tỷ đồng** chỉ trong vòng 10 ngày. Điều này phản ánh quy mô ủng hộ rất lớn từ cộng đồng trong khoảng thời gian ngắn, giúp MTTQ Việt Nam có thêm nhiều nguồn lực để hỗ trợ công tác cứu trợ và khắc phục hậu quả bão lũ.

Mức quyên góp trung bình.

Dưới đây là đoạn code về mức quyên góp trung bình



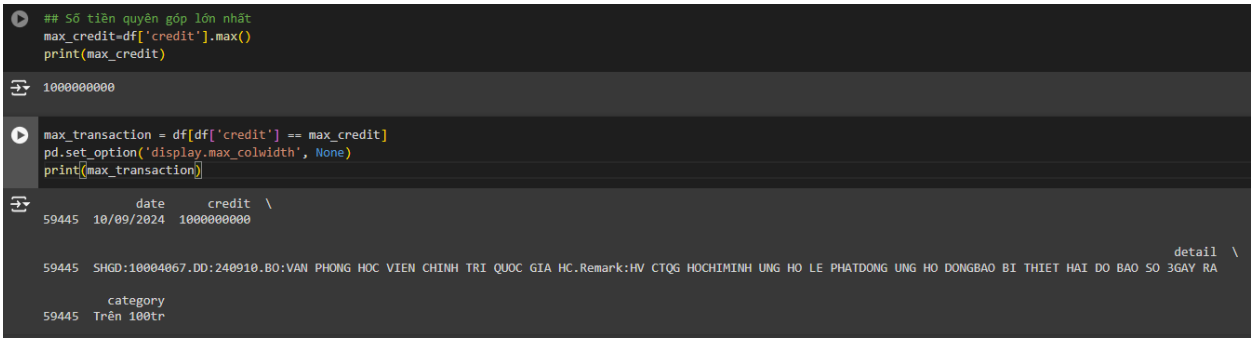
Ta có thể thấy mức quyên góp trung bình đạt 670 nghìn đồng, con số này có vẻ hơi lớn so với con số thực tế, điều này xảy ra là do có nhiều cá nhân hoặc tập thể quyên góp số tiền cực lớn đã nâng số trung bình lên. Vì vậy, tôi sẽ sử dụng median để đưa ra đánh giá khách quan hơn, không bị ảnh hưởng nhiều bởi outliers



Số tiền quyên góp trung vị là 200 nghìn đồng ,số tiền này có vẻ thực tế hơn so với con số trên, dẫu vậy thì đây vẫn là một số tiền lớn với mỗi lượt đóng góp. Điều này cho thấy rằng cộng đồng đã đóng góp với lòng hảo tâm rất lớn, đồng thời thể hiện tinh thần tương thân tương ái mạnh mẽ trong đợt cứu trợ bão lũ

Mức quyên góp lớn nhất

Dưới đây là thông tin mức quyên góp lớn nhất



Mức quyên góp lớn nhất đạt 1 tỷ đồng do Học viện Chính trị Quốc gia Hồ Chí Minh quyên góp vào ngày 10/9. Đây là thời điểm sau khi bão Yagi đổ bộ, số tiền trên là nhằm khắc phục thiệt hại do bão gây ra

Phân phối của số tiền chuyển khoản

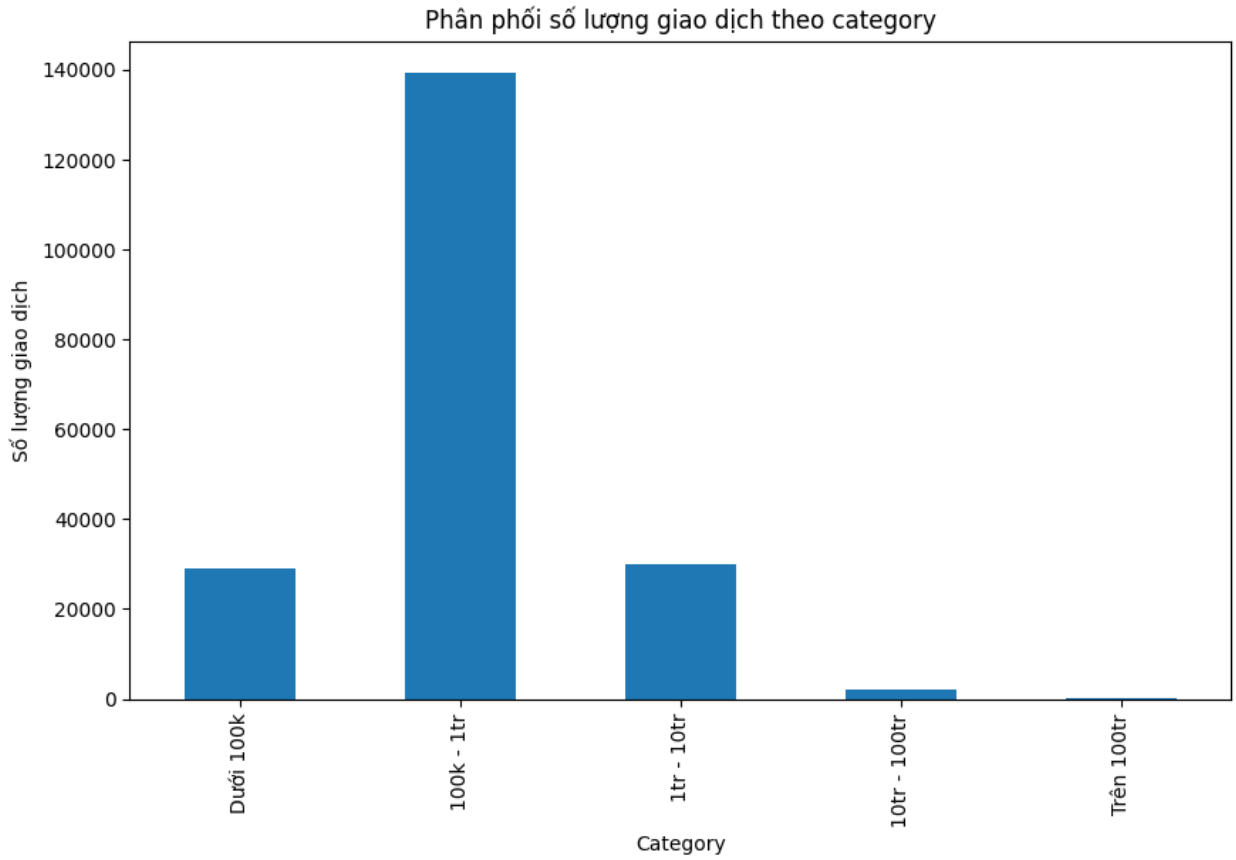
Tiếp theo ta sẽ tìm phân phối của số tiền được chuyển , đầu tiên ta sẽ tạo thêm cột phân loại số tiền theo các mốc: Dưới 100k, 100k-1tr, 1tr-10tr, 10tr-100tr, trên 100tr. Đoạn code phân loại như sau

```
[11] ## Xem phân phối số tiền chuyển khoản

     ## Tạo thêm cột phân loại số tiền
     def categorize_amount(amount):
         if amount < 100000:
             return 'Dưới 100k'
         elif amount < 1000000:
             return '100k - 1tr'
         elif amount < 10000000:
             return '1tr - 10tr'
         elif amount < 100000000:
             return '10tr - 100tr'
         else:
             return 'Trên 100tr'

     df['category'] = df['credit'].apply(categorize_amount)
```

Sau đó tôi sẽ tạo chart countplot để xem phân phối quyên góp bằng code dưới đây

```
[12] ## Tạo biểu đồ
     order = ['Dưới 100k', '100k - 1tr', '1tr - 10tr', '10tr - 100tr', 'Trên 100tr']
     counts = df['category'].value_counts().reindex(order)
     plt.figure(figsize=(10,6))
     counts.plot(kind='bar')
     plt.title('Phân phối số lượng giao dịch theo S')
     plt.xlabel('Category')
     plt.ylabel('Số lượng giao dịch')
     plt.show()
```

Dưới đây sẽ là chart phân phối số tiền được chuyển



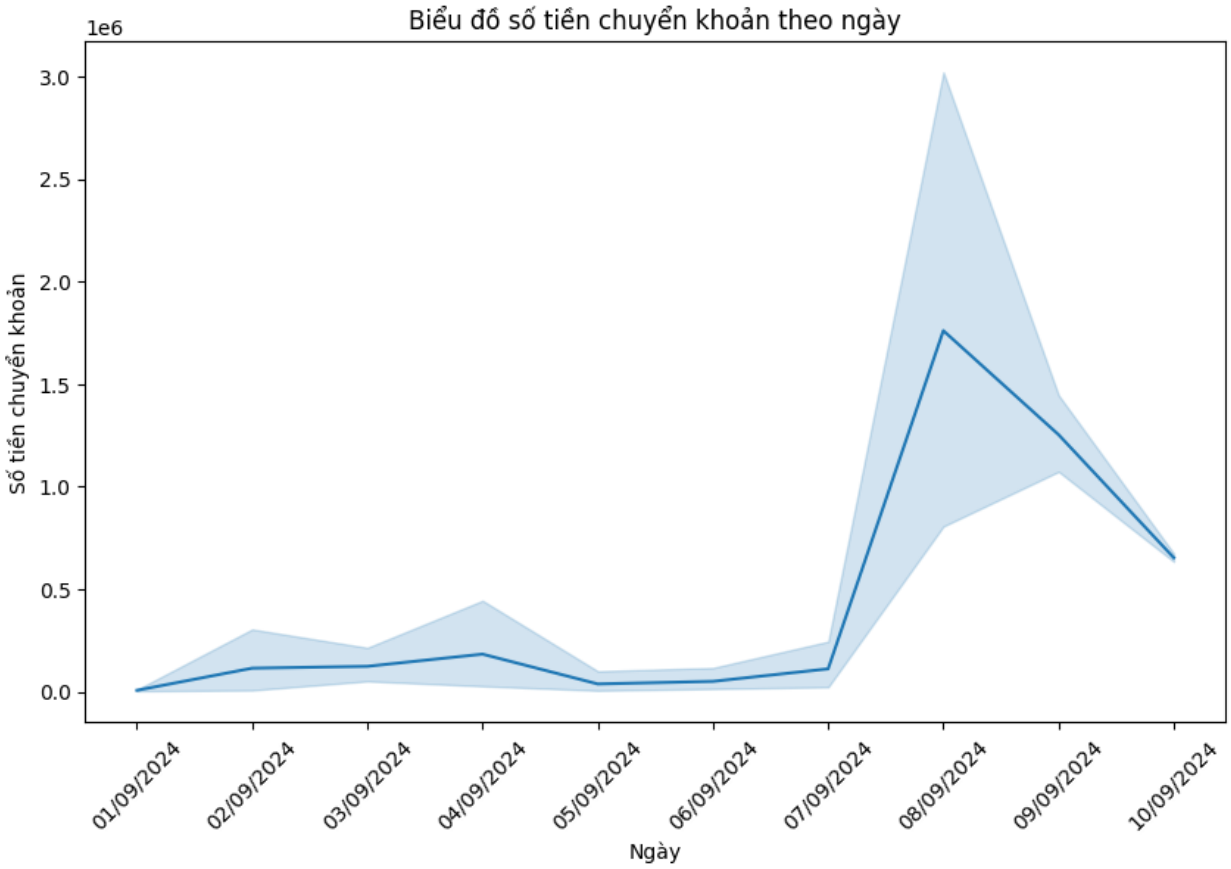
Biểu đồ cho thấy số tiền được ủng hộ nhiều nhất là từ 100k-1tr, đây là mức đóng góp phù hợp với khả năng tài chính của nhiều người. Cũng có một số lượng đáng kể giao dịch dưới 100 nghìn đồng cho thấy nhiều người dù không đủ điều kiện nhưng vẫn chung tay đáng góp. Từ 1tr-10tr là số tiền không nhỏ nhưng độ phổ biến vẫn ngang với số người ủng hộ dưới 100k. Ta có thể thấy ở biểu đồ này phần lớn các khoản đóng góp có giá trị vừa phải, phản ánh tinh thần "góp gió thành bão" của người dân.

Biểu đồ số tiền được chuyển theo thời gian

Dưới đây là đoạn code để vẽ chart số tiền được chuyển theo thời gian.

```
## Biểu đồ số tiền chuyển khoản theo thời gian
plt.figure(figsize=(10,6))
sns.lineplot(df,x='date',y='credit')
plt.title('Biểu đồ số tiền chuyển khoản theo ngày')
plt.xlabel('Ngày')
plt.ylabel('Số tiền chuyển khoản')
plt.xticks(rotation=45)
```

Còn đây sẽ là chart của chúng ta

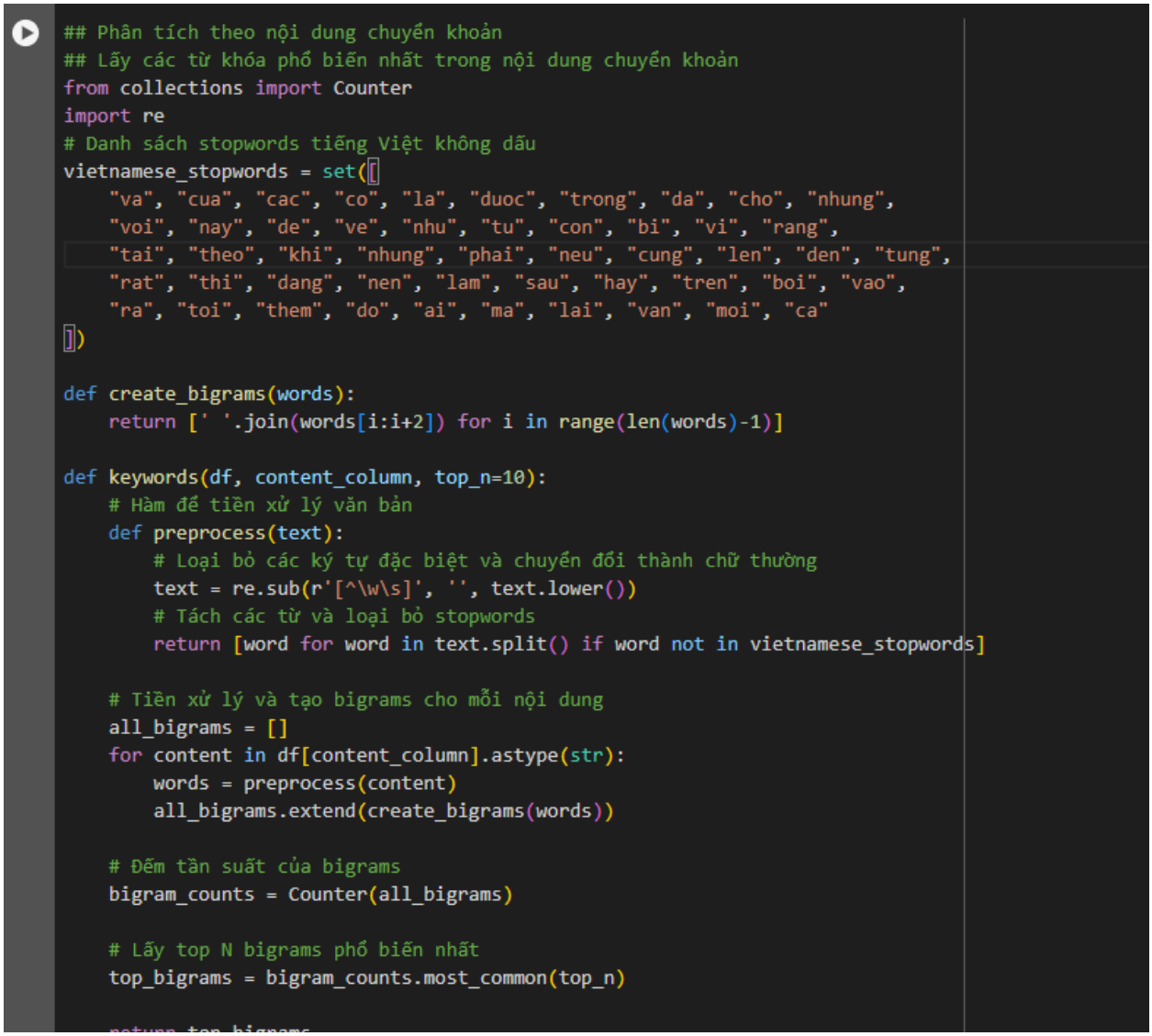


Từ biểu đồ trên ta có thể thấy số tiền quyên góp có biến động nhẹ trong ngày quốc khánh là 1/9. Số tiền cũng đạt một mốc tăng trưởng mới vào ngày 4/9, chúng ta đã bắt đầu nghe những tin tức về bão Yagi, đây là thời điểm những tin tức cảnh báo về siêu bão xuất hiện liên tục và đồng thời cũng là thời điểm sau khi cơn bão quét qua Philippines. Cột mốc đỉnh mới nằm ở ngày 8/9, đây là thời điểm sau khi siêu bão đã đi qua Hà Nội, người dân đã nhận thức rõ về thiệt hại và nhu cầu hỗ trợ nên đã ủng hộ nhiều hơn. Những ngày tiếp theo số tiền quyên góp có xu hướng giảm nhưng vẫn cao hơn so với những ngày đầu tháng.

Biểu đồ cho thấy sự phản ứng nhanh chóng và mạnh mẽ của cộng đồng đối với thảm họa thiên nhiên và sự duy trì mức quyên góp cao trong vài ngày sau đỉnh điểm thể hiện tinh thần tương thân tương ái kéo dài của cộng đồng.

Phân tích theo nội dung chuyển khoản

Tôi sẽ tạo code lấy hai từ khóa liên tiếp phổ biến nhất trong nội dung chuyển khoản như sau



```python
## Phân tích theo nội dung chuyển khoản
## Lấy các từ khóa phổ biến nhất trong nội dung chuyển khoản
from collections import Counter
import re
# Danh sách stopwords tiếng Việt không dấu
vietnamese_stopwords = set([
    "va", "cua", "cac", "co", "la", "duoc", "trong", "da", "cho", "nhung",
    "voi", "nay", "de", "ve", "nhu", "tu", "con", "bi", "vi", "rang",
    "tai", "theo", "khi", "nhung", "phai", "neu", "cung", "len", "den", "tung",
    "rat", "thi", "dang", "nen", "lam", "sau", "hay", "tren", "boi", "vao",
    "ra", "toi", "them", "do", "ai", "ma", "lai", "van", "moi", "ca"
])

def create_bigrams(words):
    return [' '.join(words[i:i+2]) for i in range(len(words)-1)]

def keywords(df, content_column, top_n=10):
    # Hàm để tiền xử lý văn bản
    def preprocess(text):
        # Loại bỏ các ký tự đặc biệt và chuyển đổi thành chữ thường
        text = re.sub(r'[^\w\s]', '', text.lower())
        # Tách các từ và loại bỏ stopwords
        return [word for word in text.split() if word not in vietnamese_stopwords]

    # Tiền xử lý và tạo bigrams cho mỗi nội dung
    all_bigrams = []
    for content in df[content_column].astype(str):
        words = preprocess(content)
        all_bigrams.extend(create_bigrams(words))

    # Đếm tần suất của bigrams
    bigram_counts = Counter(all_bigrams)

    # Lấy top N bigrams phổ biến nhất
    top_bigrams = bigram_counts.most_common(top_n)

    return top_bigrams
```

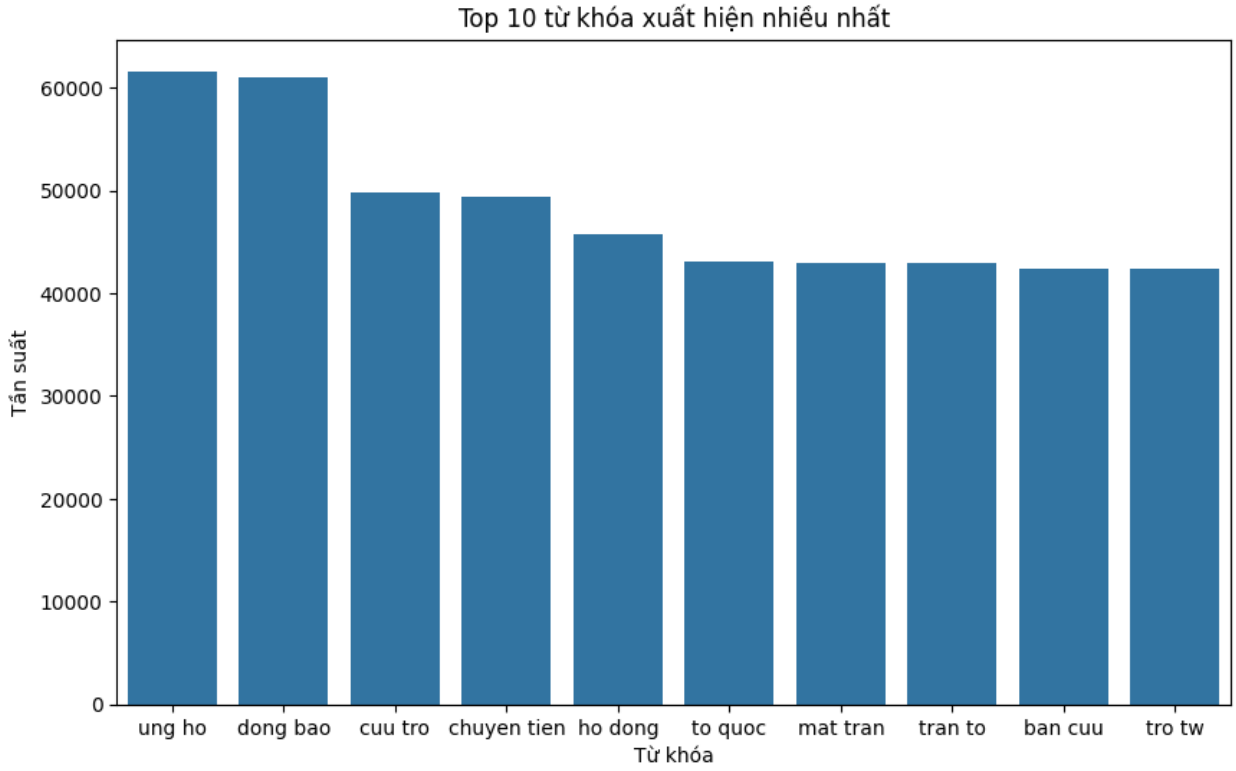
Đoạn code này tiến hành phân tích nội dung chuyển khoản để xác định các cụm từ hai từ (**bigrams**) phổ biến nhất bằng cách loại trừ những từ không mang nhiều ý nghĩa (**stopwords**). Quá trình này bắt đầu bằng cách tiền xử lý văn bản: loại bỏ các ký tự đặc biệt và chuyển toàn bộ văn bản thành chữ thường. Sau đó, từ văn bản đã được làm sạch này, các từ được tách ra và những từ thuộc danh sách từ dừng sẽ được bỏ qua. Các từ còn lại được ghép thành cặp để tạo thành **bigrams**. Cuối cùng, hàm này sử dụng **counter** để đếm tần suất xuất hiện của mỗi **bigram** trong toàn bộ dataset và trả về danh sách các **bigrams** xuất hiện nhiều nhất.

Tạo biểu đồ top 10 các từ khóa xuất hiện nhiều nhất

Biểu đồ sẽ được code như sau

```
## Tạo biểu đồ top các từ khóa xuất hiện nhiều nhất
top_keyword=keywords(df,'detail',10)
print(top_keyword)
plt.figure(figsize=(10,6))
sns.barplot(x=[keyword[0] for keyword in top_keyword],y=[keyword[1] for keyword in top_keyword])
plt.title('Top 10 từ khóa xuất hiện nhiều nhất')
plt.xlabel('Từ khóa')
plt.ylabel('Tần suất')
```

Và dưới đây là biểu đồ



Có thể thấy nội dung chuyển khoản tập trung vào mục đích ủng hộ, cứu trợ đồng bào gặp nạn do lũ, người dân sử dụng ngôn ngữ đơn giản và trực tiếp để thể hiện mục đích quyên góp của mình.

Số lượng quyên góp của tập thể và cá nhân

Tiếp theo tôi sẽ tìm số lượng quyên góp của tập thể và cá nhân thông qua lọc các từ khóa có khả năng là tập thể và xét những trường hợp không phải là tập thể thì có thể là cá nhân. Phần đánh giá này chỉ mang tính chất tương đối, không phản ánh được chính xác hoàn toàn về đóng góp của tập thể hoặc cá nhân.

Đầu tiên tôi sẽ lọc các từ khóa của nội dung chuyển khoản có khả năng là tập thể đóng góp

```
def classify_donor(content):
    # Danh sách từ khóa gợi ý về tập thể
    collective_keywords = ['cty', 'congty', 'company', 'corp', 'tap doan', 'tapdoan', 'to chuc', 'tochuc', 'nhom', 'group','tap the','tapthe','chi doan',
                           'chidoan','trung tam','trungtam','lien doan','liendoan','anh em','anhem','cong ty','media','hoi','club','studio']

    # Chuyển nội dung về chữ thường và loại bỏ dấu cách thừa
    content = ' '.join(content.lower().split())

    # Kiểm tra xem có từ khóa nào của tập thể xuất hiện không
    if any(keyword in content for keyword in collective_keywords):
        return 'Tập thể'

    # Mặc định là cá nhân
    return 'Cá nhân'

def analyze_donations(df, content_column, amount_column, top_n=10):
    def preprocess(text):
        text = re.sub(r'[^\w\s]', '', text.lower())
        return [word for word in text.split() if word not in vietnamese_stopwords]

    all_bigrams = []
    donor_types = {'Tập thể': 0, 'Cá nhân': 0}
    total_amount = {'Tập thể': 0, 'Cá nhân': 0}

    for content, amount in zip(df[content_column].astype(str), df[amount_column]):
        words = preprocess(content)
        all_bigrams.extend(create_bigrams(words))

        donor_type = classify_donor(content)
        donor_types[donor_type] += 1
        total_amount[donor_type] += amount

    bigram_counts = Counter(all_bigrams)
    top_bigrams = bigram_counts.most_common(top_n)

    return top_bigrams, donor_types, total_amount
```

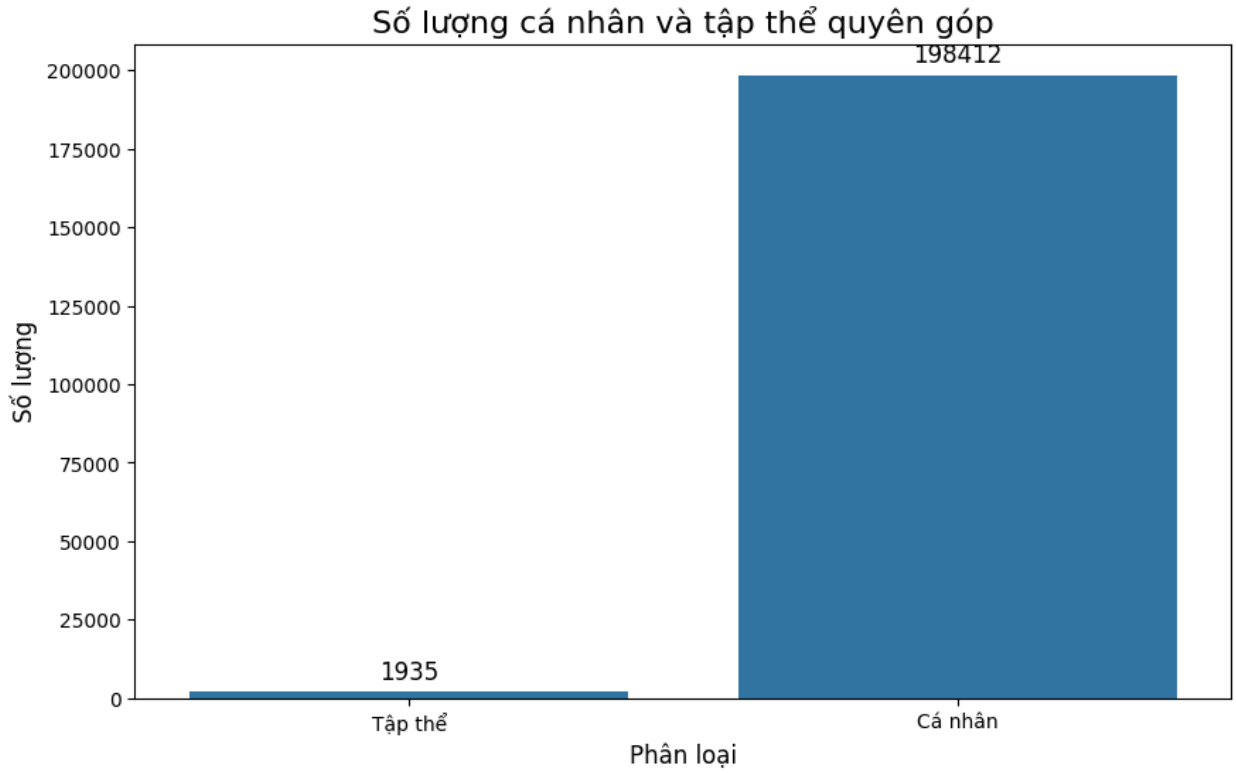
Đầu tiên, hàm **classify_donor** sẽ kiểm tra nội dung chuyển khoản để xác định người quyên góp là "Tập thể" hay "Cá nhân" dựa trên sự hiện diện của các từ khóa liên quan đến tổ chức hoặc tập thể (ví dụ như "cty", "congty", "group", "club", v.v.).

Hàm analyze_donation thực hiện việc phân tích toàn diện. Nó tiền xử lý nội dung chuyển khoản bằng cách chuyển văn bản thành chữ thường và loại bỏ ký tự đặc biệt, sau đó tạo các bigrams từ danh sách các từ đã qua xử lý. Đồng thời, hàm cũng sử dụng **classify_donor** để phân loại mỗi giao dịch là từ cá nhân hay tập thể, sau đó đếm số lần xuất hiện và tổng số tiền quyên góp của từng loại.

Dưới đây là đoạn code vẽ biểu đồ đếm số tập thể và cá nhân

```
## Đếm số lượng tập thể và cá nhân
plt.figure(figsize=(10,6))
top_keyword, donor_types, total_amount = analyze_donations(df, 'detail', 'credit')
ax = sns.barplot(x=list(donor_types.keys()), y=list(donor_types.values()))
ax.set_title('Số lượng cá nhân và tập thể quyên góp', fontsize=16)
ax.set_xlabel('Loại người quyên góp', fontsize=12)
ax.set_ylabel('Số lượng', fontsize=12)
for p in ax.patches:
    ax.annotate(f'{int(p.get_height())}',
                (p.get_x() + p.get_width() / 2., p.get_height()),
                ha='center', va='center', fontsize=12, color='black',
                xytext=(0, 10), textcoords='offset points')

plt.show()
```

Và dưới đây là biểu đồ



Có thể thấy lượng chênh lệch giữa cá nhân và tập thể trong biểu đồ trên, cụ thể Số lượng cá nhân quyên góp cao hơn gấp khoảng 102 lần so với tập thể và số lượng cá nhân chiếm khoảng 99% tổng số lượt quyên góp. Nhưng chúng ta cũng nên có một lưu ý rằng số

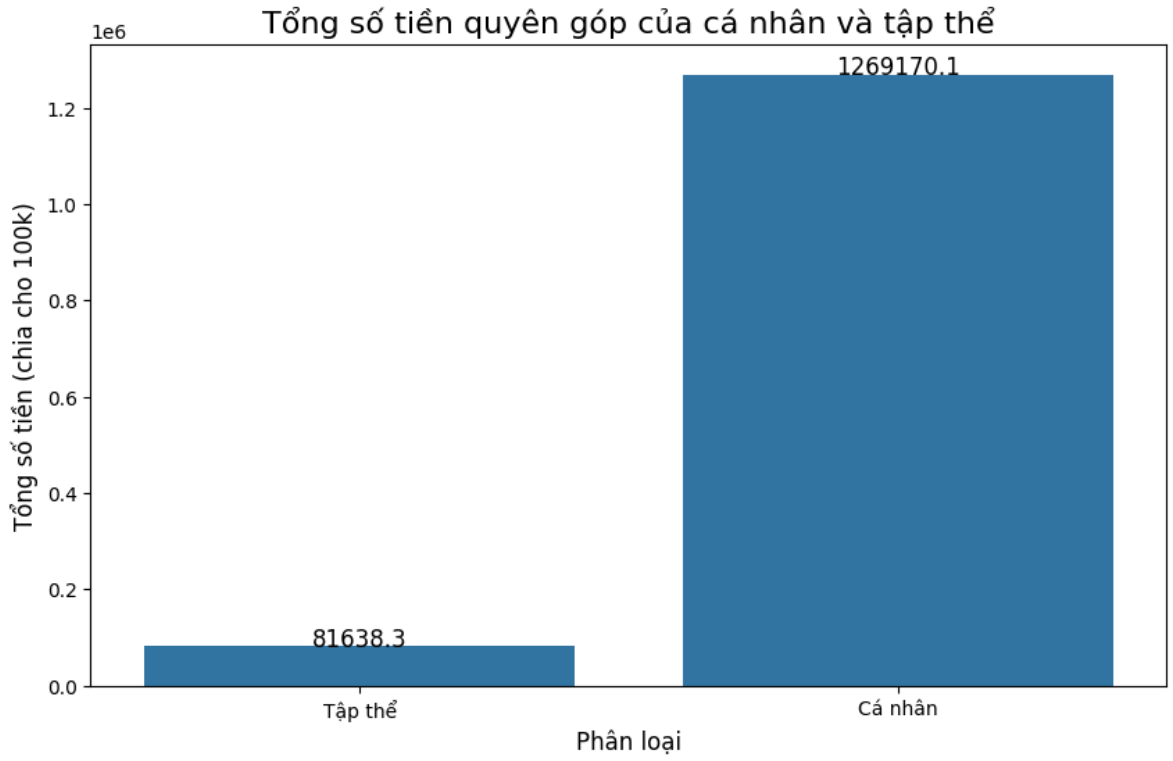
lượng tập thể cũng không phải là một con số nhỏ với lượng từ khóa không đại diện được hết cho họ như vậy.

Số tiền quyên góp của tập thể và cá nhân

Dưới đây sẽ là đoạn code thể hiện số tiền quyên góp của tập thể và cá nhân.

```
## Số tiền quyên góp của tập thể và cá nhân
plt.figure(figsize=(10,6))
scaled_values = [value / 100000 for value in total_amount.values()]
ax = sns.barplot(x=list(total_amount.keys()), y=scaled_values)
ax.set_title('Tổng số tiền quyên góp của cá nhân và tập thể', fontsize=16)
ax.set_xlabel('Loại người quyên góp', fontsize=12)
ax.set_ylabel('Tổng số tiền (chia cho 100k)', fontsize=12)
for i, value in enumerate(scaled_values):
    ax.text(i, value + 0.1, f'{value:.1f}', ha='center', fontsize=12)

plt.show()
```

Dưới đây sẽ là chart của đoạn code trên.



Ta có thể thấy tuy số lượng tập thể ít nhưng tỉ lệ số tiền đóng góp trên mỗi lần chuyển khoản cao gấp 8 lần cá nhân, phản ánh sự tham gia của các tổ chức, doanh nghiệp lớn với khả năng tài chính mạnh. Mặc dù số lượt quyên góp của tập thể chỉ chiếm 1%, nhưng giá trị đóng

góp lại chiếm 6% tổng số tiền.Có thể rút ra rằng sự kết hợp giữa số lượng lớn các khoản đóng góp nhỏ từ cá nhân và các khoản đóng góp lớn từ tập thể tạo nên một nguồn lực đáng kể cho công tác cứu trợ.

Tổng kết.

Quy mô và tác động của chiến dịch quyên góp:

- Tổng số tiền quyên góp đạt khoảng 135 tỷ đồng, thể hiện sự thành công đáng kể của chiến dịch.
- Có sự tham gia rộng rãi với hơn 200,000 lượt quyên góp, chủ yếu từ cá nhân (198,412 lượt).

Vai trò của cá nhân và tập thể:

- Cá nhân đóng góp chủ đạo với 94% tổng số tiền (126.9 tỷ đồng) và 99% số lượt quyên góp.
- Tập thể, dù chỉ chiếm 1% số lượt, nhưng đóng góp 6% tổng số tiền (8.16 tỷ đồng), cho thấy giá trị trung bình mỗi khoản đóng góp của tập thể cao hơn.

Xu hướng quyên góp:

- Đa số khoản quyên góp có giá trị từ 100,000 đến 1 triệu đồng, phản ánh khả năng tài chính phổ biến của người dân.
- Số lượng giao dịch giảm khi giá trị tăng, nhưng vẫn có những khoản đóng góp lớn.

Thời điểm và động lực quyên góp:

- Số tiền quyên góp tăng mạnh sau khi có thông tin về bão và đạt đỉnh sau khi bão đi qua, cho thấy phản ứng nhanh chóng của cộng đồng.
- Từ khóa phổ biến như "ung ho", "dong bao", "cuu tro" phản ánh tinh thần đoàn kết và mục đích rõ ràng của người quyên góp.

Hiệu quả của phương thức quyên góp:

- Khả năng tiếp cận dễ dàng (có thể qua hình thức online) đã thúc đẩy sự tham gia rộng rãi của cá nhân.
- Truyền thông hiệu quả góp phần lan tỏa thông tin và kêu gọi sự đóng góp.

Ý nghĩa xã hội:

- Chiến dịch thể hiện rõ tinh thần tương thân tương ái và trách nhiệm xã hội cao của cộng đồng.
- Sự kết hợp giữa số lượng lớn các khoản đóng góp nhỏ và các khoản đóng góp lớn tạo nên nguồn lực đáng kể cho công tác cứu trợ.