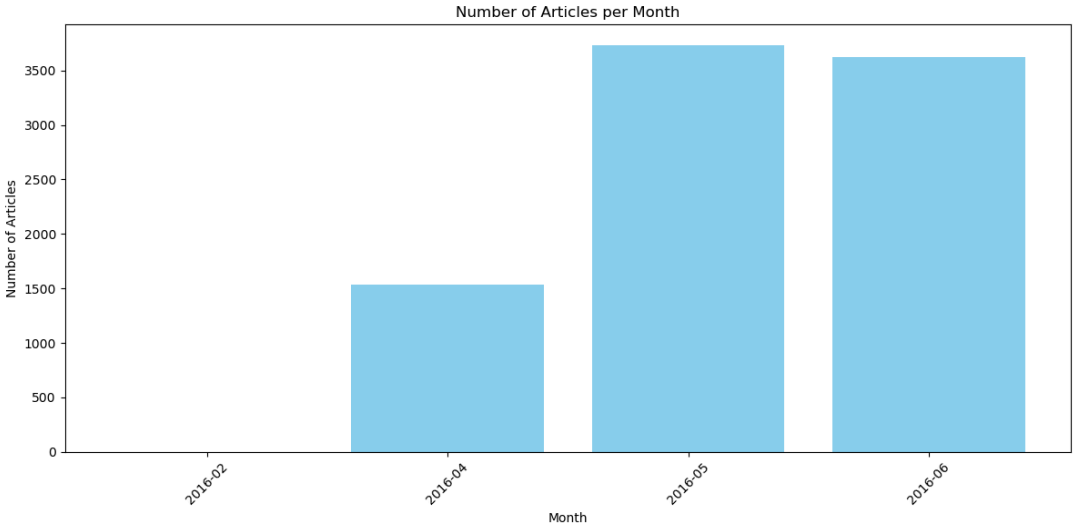
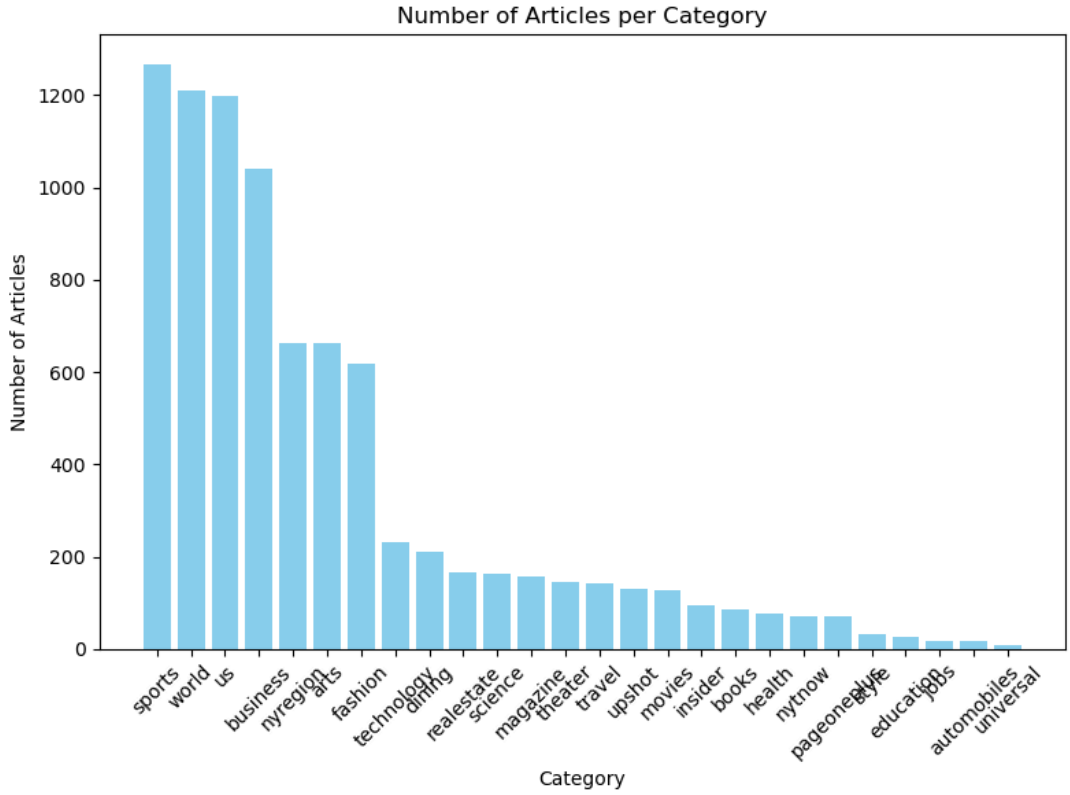
In [1]:

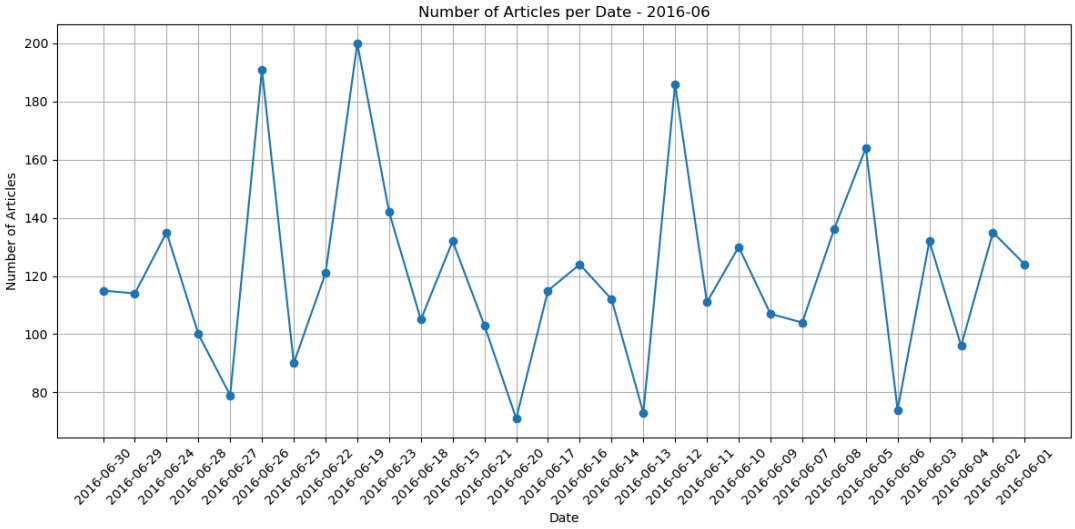
```python
import re
from collections import Counter
import matplotlib.pyplot as plt

# Load the text file containing news articles
file_path = "C:/Users/91830/Documents/cleaned_documet.txt"
with open(file_path, 'r', encoding='utf-8') as file:
    data = file.read()

# Extract URLs using regular expressions
urls = re.findall(r'(https?://\S+)', data)

# Extract publication dates and categories from URLs
dates = []
categories = []
for url in urls:
    date_match = re.search(r'/(\d{4})/(\d{2})/(\d{2})/', url)
    if date_match:
        year, month, day = date_match.groups()
        dates.append(f"{year}-{month}")
    category_match = re.search(r'nytimes\.com/\d{4}/\d{2}/\d{2}/(\w+)/'
    if category_match:
        categories.append(category_match.group(1))

# Plot graph for dates and categories
date_counts = Counter(dates)
category_counts = Counter(categories)

# Task 1: Plot graph for articles per month
months_sorted = sorted(date_counts.items())
months, counts = zip(*months_sorted)

plt.figure(figsize=(12, 6))
plt.bar(months, counts, color='skyblue')
plt.xlabel('Month')
plt.ylabel('Number of Articles')
plt.title('Number of Articles per Month')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()

# Task 2: Plot graph for categories
categories_sorted = sorted(category_counts.items(), key=lambda x: x[1],
plt.figure(figsize=(8, 6))
plt.bar([category for category, _ in categories_sorted], [count for _,
plt.xlabel('Category')
plt.ylabel('Number of Articles')
plt.title('Number of Articles per Category')
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

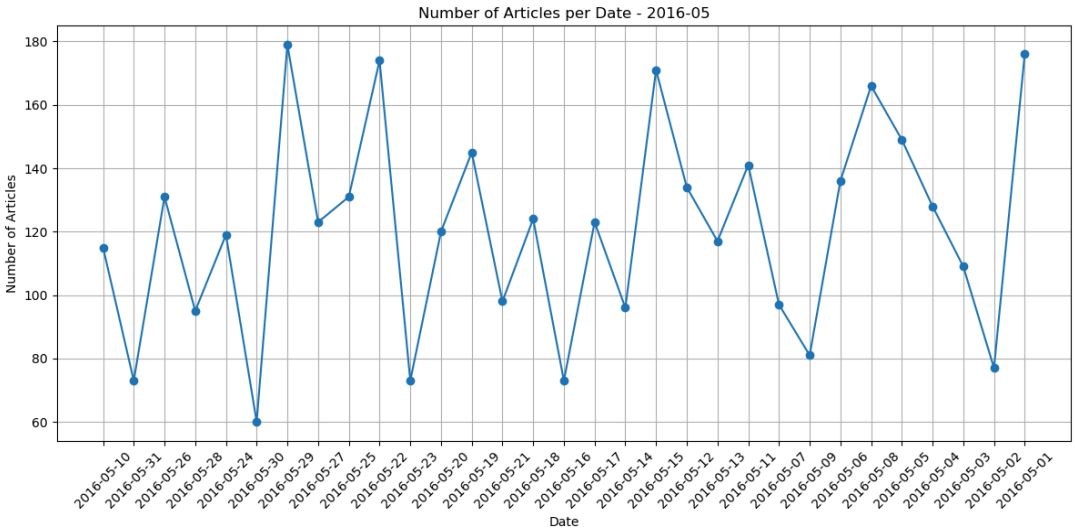

Number of Articles per Month
Number of Articles
0
500
1000
1500
2000
2500
3000
3500
2016-02
2016-04
2016-05
2016-06
Month

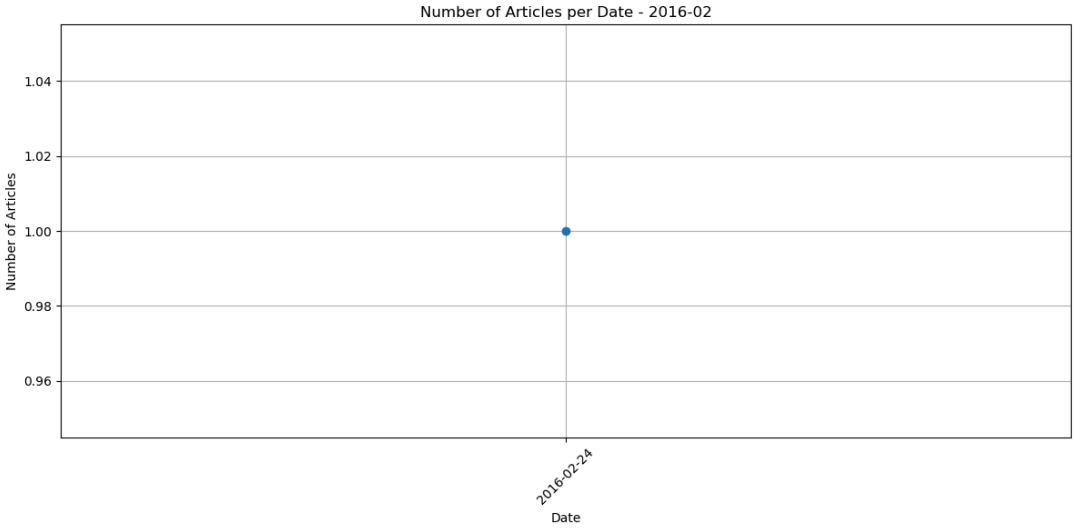

Number of Articles per Category
Number of Articles
0
200
400
600
800
1000
1200
sports
world
us
business
nyregion
arts
fashion
technology
dining
realestate
science
magazine
theater
travel
upshot
movies
insider
books
health
nytnow
pageoneplus
style
education
jobs
automobiles
universal
Category

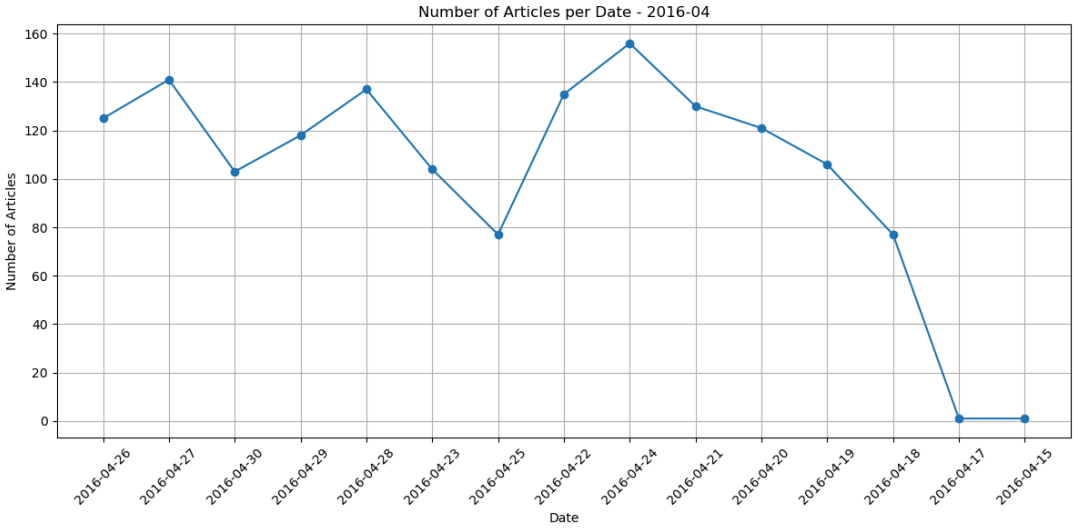
In [2]:

```python
import re
from collections import Counter
import matplotlib.pyplot as plt

# Load the text file containing news articles
file_path = "C:/Users/91830/Documents/cleaned_documet.txt"
with open(file_path, 'r', encoding='utf-8') as file:
    data = file.read()

# Extract URLs using regular expressions
urls = re.findall(r'(https?://\S+)', data)

# Extract publication dates and categories from URLs
dates = []
for url in urls:
    date_match = re.search(r'/(\d{4})/(\d{2})/(\d{2})/', url)
    if date_match:
        year, month, day = date_match.groups()
        dates.append(f"{year}-{month}-{day}")

# Plot graph for articles per date
date_counts = Counter(dates)

# Group article counts by month
month_groups = {}
for date, count in date_counts.items():
    year, month, _ = date.split('-')
    month_key = f"{year}-{month}"
    if month_key not in month_groups:
        month_groups[month_key] = []
    month_groups[month_key].append((date, count))

# Plot graph for each month
for month, counts in month_groups.items():
    dates, article_counts = zip(*counts)
    plt.figure(figsize=(12, 6))
    plt.plot(dates, article_counts, marker='o')
    plt.xlabel('Date')
    plt.ylabel('Number of Articles')
    plt.title(f'Number of Articles per Date - {month}')
    plt.xticks(rotation=45)
    plt.grid(True)
    plt.tight_layout()
    plt.show()
```


Number of Articles per Date - 2016-06
Number of Articles
Date

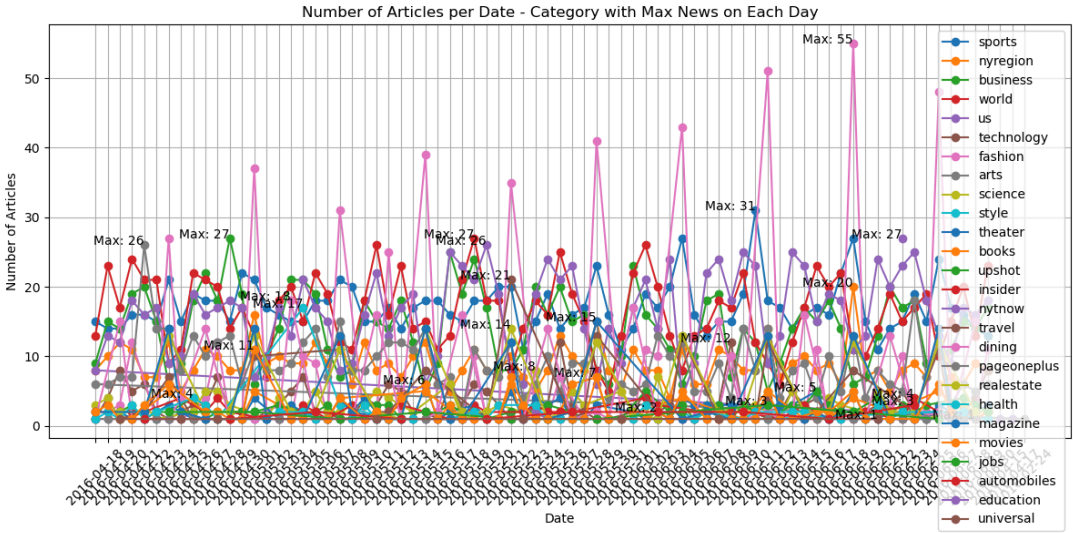

Number of Articles per Date - 2016-05
Number of Articles
Date


Number of Articles per Date - 2016-02
Number of Articles
Date


Number of Articles per Date - 2016-04
Number of Articles
Date
0
20
40
60
80
100
120
140
160
2016-04-26
2016-04-27
2016-04-30
2016-04-29
2016-04-28
2016-04-23
2016-04-25
2016-04-22
2016-04-24
2016-04-21
2016-04-20
2016-04-19
2016-04-18
2016-04-17
2016-04-15

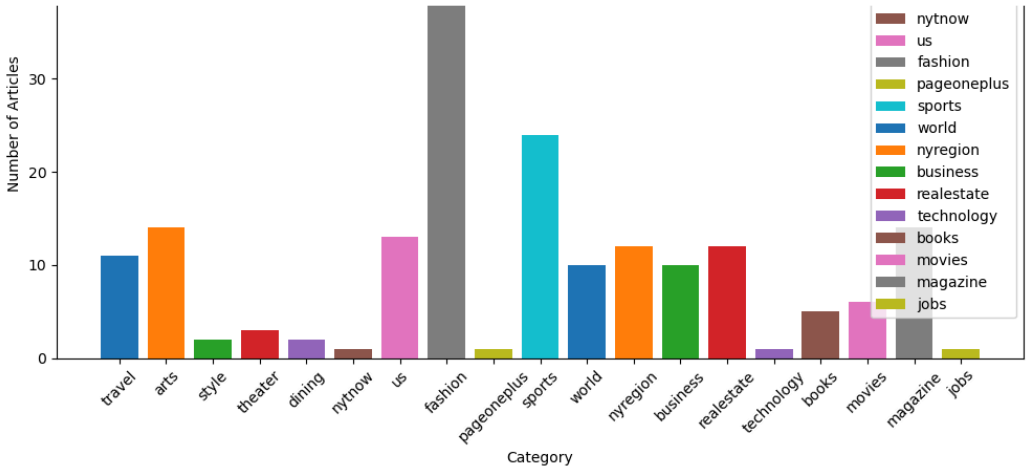
In [3]:

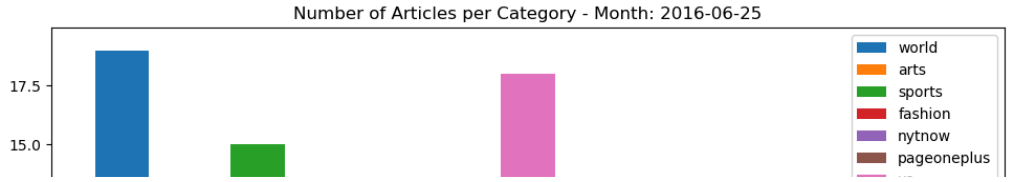
```
import re
from collections import defaultdict, Counter
import matplotlib.pyplot as plt

# Load the text file containing news articles
file_path = "C:/Users/91830/Documents/cleaned_documet.txt"
with open(file_path, 'r', encoding='utf-8') as file:
    data = file.read()

# Extract URLs using regular expressions
urls = re.findall(r'(https?://\S+)', data)

# Extract publication dates and categories from URLs
dates = []
categories = []
for url in urls:
    date_match = re.search(r'/(\d{4})/(\d{2})/(\d{2})/', url)
    if date_match:
        year, month, day = date_match.groups()
        dates.append(f"{year}-{month}-{day}")
    category_match = re.search(r'nytimes\.com/\d{4}/\d{2}/\d{2}/(\w+)/'
    if category_match:
        categories.append((f"{year}-{month}-{day}", category_match.grou

# Group articles by category and date
category_counts = defaultdict(Counter)
for date, category in categories:
    category_counts[category][date] += 1

# Find the day with the highest rate of news for each category
max_days = {category: max(counts, key=counts.get) for category, counts

# Plot the graph
plt.figure(figsize=(12, 6))
for category, counts in category_counts.items():
    dates, article_counts = zip(*sorted(counts.items()))
    plt.plot(dates, article_counts, marker='o', label=category)
    max_day = max_days[category]
    max_count = counts[max_day]
    plt.text(max_day, max_count, f"Max: {max_count}", ha='right')
plt.xlabel('Date')
plt.ylabel('Number of Articles')
plt.title('Number of Articles per Date - Category with Max News on Each
plt.xticks(rotation=45)
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()
```


Number of Articles per Date - Category with Max News on Each Day
Number of Articles
Date
Max: 55
Max: 31
Max: 27
Max: 26
Max: 20
Max: 19
Max: 17
Max: 15
Max: 14
Max: 12
Max: 11
Max: 8
Max: 7
Max: 6
Max: 5
Max: 4
Max: 3
Max: 2
Max: 1
sports
nyregion
business
world
us
technology
fashion
arts
science
style
theater
books
upshot
insider
nytnow
travel
dining
pageoneplus
realestate
health
magazine
movies
jobs
automobiles
education
universal

In [4]:

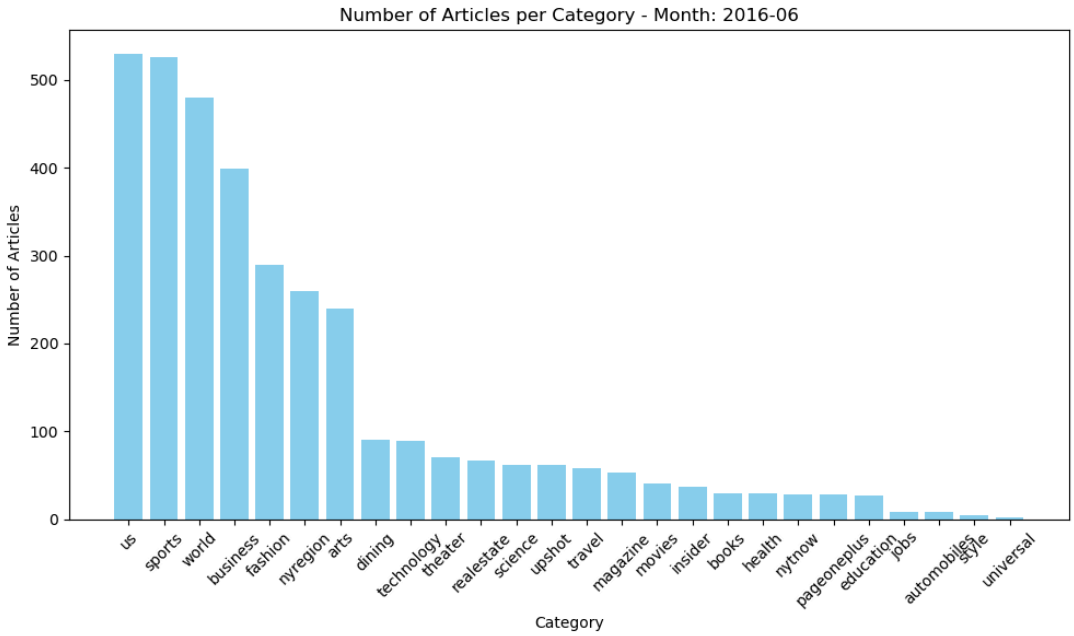
```
import re
from collections import defaultdict, Counter
import matplotlib.pyplot as plt

# Load the text file containing news articles
file_path = "C:/Users/91830/Documents/cleaned_documet.txt"
with open(file_path, 'r', encoding='utf-8') as file:
    data = file.read()

# Extract URLs using regular expressions
urls = re.findall(r'(https?://\S+)', data)

# Extract publication dates and categories from URLs
dates = []
categories = []
for url in urls:
    date_match = re.search(r'/(\d{4})/(\d{2})/(\d{2})/', url)
    if date_match:
        year, month, day = date_match.groups()
        dates.append(f"{year}-{month}")
    category_match = re.search(r'nytimes\.com/\d{4}/\d{2}/\d{2}/(\w+)/'
    if category_match:
        categories.append((f"{year}-{month}-{day}", category_match.grou

# Group articles by month and category
month_category_counts = defaultdict(lambda: defaultdict(int))
for date, category in categories:
    month_category_counts[date][category] += 1

# Plot individual graphs for each month
for month, category_counts in month_category_counts.items():
    plt.figure(figsize=(10, 6))
    for category, count in category_counts.items():
        plt.bar(category, count, label=category)
    plt.xlabel('Category')
    plt.ylabel('Number of Articles')
    plt.title(f'Number of Articles per Category - Month: {month}')
    plt.xticks(rotation=45)
    plt.legend()
    plt.tight_layout()
    plt.show()
```

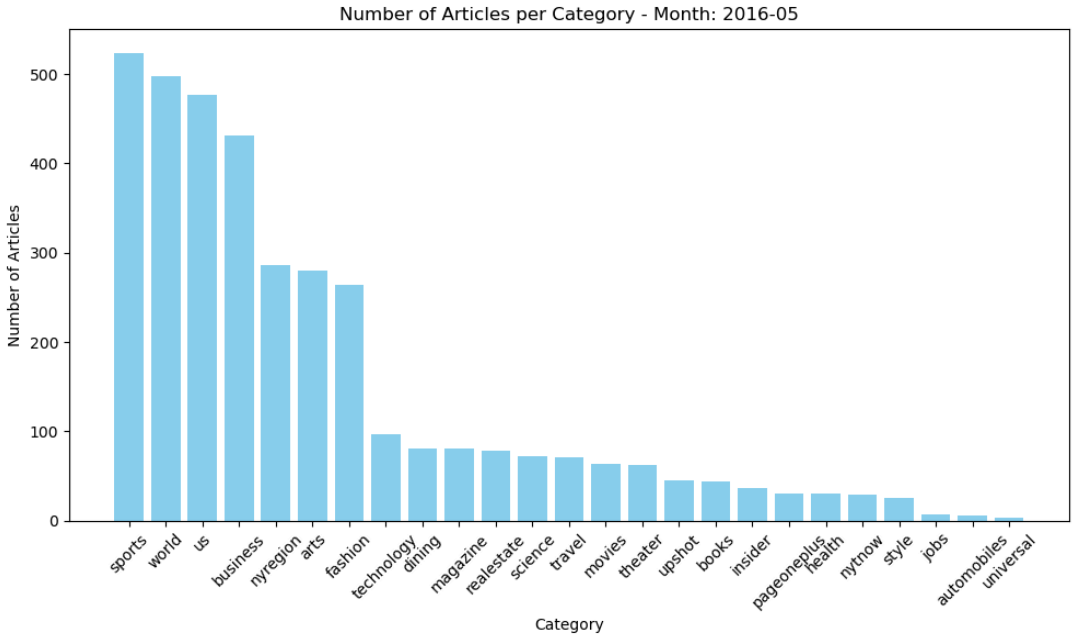

Number of Articles
0
10
20
30
travel
arts
style
theater
dining
nytnow
us
fashion
pageoneplus
sports
world
nyregion
business
realestate
technology
books
movies
magazine
jobs
Category
nytnow
us
fashion
pageoneplus
sports
world
nyregion
business
realestate
technology
books
movies
magazine
jobs

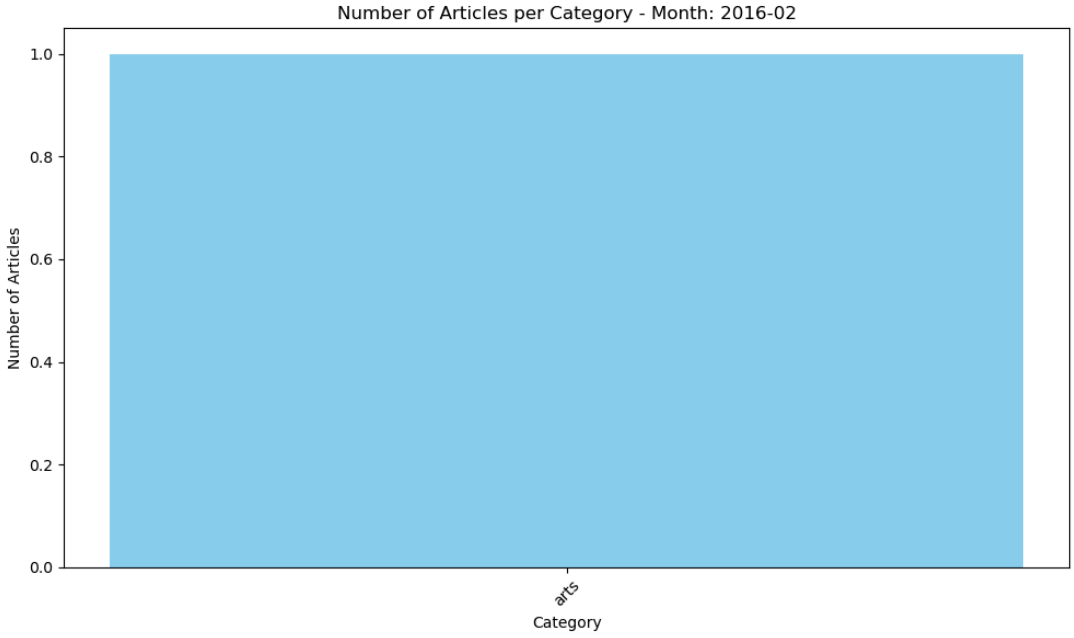

Number of Articles per Category - Month: 2016-06-25
17.5
15.0
world
arts
sports
fashion
nytnow
pageoneplus

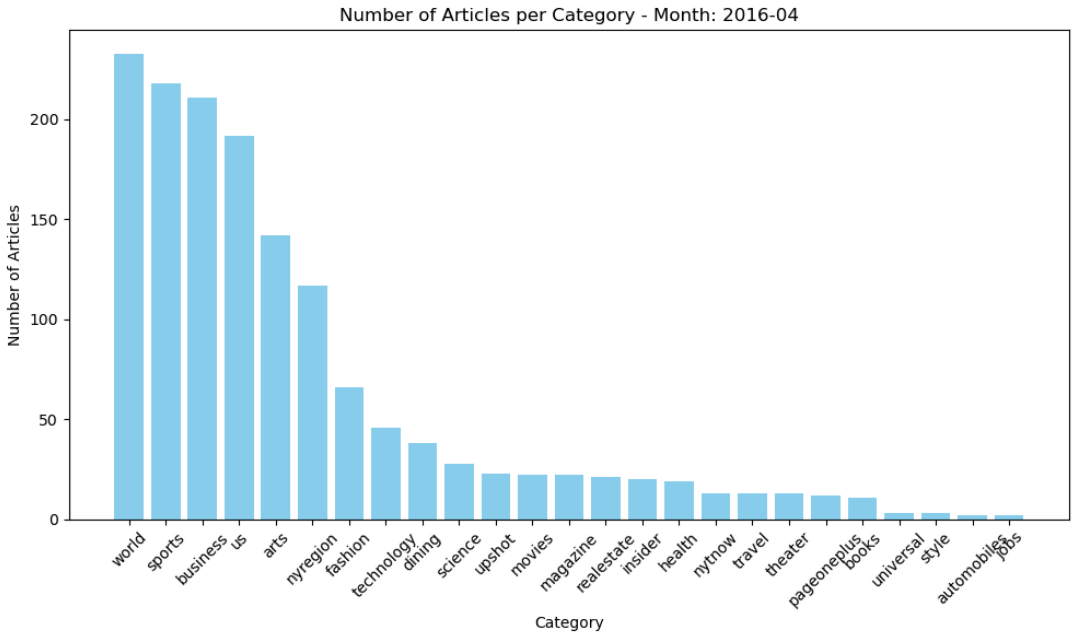
In [5]:

```
import re
from collections import defaultdict
import matplotlib.pyplot as plt
from matplotlib.dates import DateFormatter

# Load the text file containing news articles
file_path = "C:/Users/91830/Documents/cleaned_documet.txt"
with open(file_path, 'r', encoding='utf-8') as file:
    data = file.read()

# Extract URLs using regular expressions
urls = re.findall(r'(https?://\S+)', data)

# Extract publication dates and categories from URLs
dates = []
categories = []
for url in urls:
    date_match = re.search(r'/(\d{4})/(\d{2})/(\d{2})/', url)
    if date_match:
        year, month, day = date_match.groups()
        dates.append(f"{year}-{month}-{day}")
    category_match = re.search(r'nytimes\.com/\d{4}/\d{2}/\d{2}/(\w+)/'
    if category_match:
        categories.append((f"{year}-{month}-{day}", category_match.grou

# Group articles by month and category
month_category_counts = defaultdict(lambda: defaultdict(int))
for date, category in categories:
    year, month, _ = date.split('-')
    month_category_counts[f"{year}-{month}"][category] += 1

# Plot graphs for each month
for month, category_counts in month_category_counts.items():
    plt.figure(figsize=(10, 6))
    categories_sorted = sorted(category_counts.items(), key=lambda x: x
    categories, counts = zip(*categories_sorted)
    plt.bar(categories, counts, color='skyblue')
    plt.xlabel('Category')
    plt.ylabel('Number of Articles')
    plt.title(f'Number of Articles per Category - Month: {month}')
    plt.xticks(rotation=45)
    plt.tight_layout()
    plt.show()
```


Number of Articles per Category - Month: 2016-06
Number of Articles
0
100
200
300
400
500
us
sports
world
business
fashion
nyregion
arts
dining
technology
theater
realestate
science
upshot
travel
magazine
movies
insider
books
health
nytnow
pageoneplus
education
jobs
automobiles
style
universal
Category


Number of Articles per Category - Month: 2016-05
Number of Articles
0
100
200
300
400
500
sports
world
us
business
nyregion
arts
fashion
technology
dining
magazine
realestate
science
travel
movies
theater
upshot
books
insider
pageoneplus
health
nytnow
style
jobs
automobiles
universal
Category


Number of Articles per Category - Month: 2016-02
Number of Articles
0.0
0.2
0.4
0.6
0.8
1.0
arts
Category


Number of Articles per Category - Month: 2016-04
Number of Articles
0
50
100
150
200
world
sports
business
us
arts
nyregion
fashion
technology
dining
science
upshot
movies
magazine
realestate
insider
health
nytnow
travel
theater
pageoneplus
books
universal
style
automobiles
jobs
Category

In [ ]: