📌 Wikipedia Sentiment Analysis using Python
🔍 Project: Real-Time Big Data Analytics - Text Mining
👨‍💻 Author: Mohammed Zain Khan
📅 Date: May 4, 2025

In [1]:
```
import requests
from bs4 import BeautifulSoup
from textblob import TextBlob
import pandas as pd
import matplotlib.pyplot as plt
import os
```

In [7]:
```
def save_raw_html(url, filename="data/raw_wikipedia_ai.html"):
    response = requests.get(url)

    # Ensure the directory exists
    folder = os.path.dirname(filename)
    if not os.path.exists(folder):
        os.makedirs(folder)

    with open(filename, "w", encoding='utf-8') as f:
        f.write(response.text)

    return response.text
```

In [3]:

```python
# 📄 Extract content from Wikipedia
def extract_sections(soup):
    title = soup.find('h1').text.strip()

    intro = soup.find('p').text.strip()

    # Find History Section
    history_section = ""
    history_header = soup.find('span', {'id': 'History'})
    if history_header:
        history_paragraph = history_header.find_parent().find_next_sibling('p')
        if history_paragraph:
            history_section = history_paragraph.text.strip()

    # Main Content - Top 10 paragraphs
    main_content = ' '.join([p.text.strip() for p in soup.find_all('p')[:10]])

    # References
    references = soup.find('ol')
    references_text = references.text.strip() if references else "No references section found."

    return {
        'Title': title,
        'Intro': intro,
        'History': history_section,
        'Main Content': main_content,
        'References': references_text
    }
```

In [4]:
```python
# 🔍 Perform Sentiment Analysis
def analyze_sentiments(sections):
    analysis = []
    for section, text in sections.items():
        blob = TextBlob(text)
        polarity = blob.sentiment.polarity
        subjectivity = blob.sentiment.subjectivity
        analysis.append({
            'Section': section,
            'Polarity Score': polarity,
            'Subjectivity Score': subjectivity
        })
    return pd.DataFrame(analysis)
```

In [5]:
```python
# 📊 Plot Sentiment Results
def plot_sentiment(df):
    plt.figure(figsize=(10, 6))
    plt.bar(df['Section'], df['Polarity Score'], color='lightblue')
    plt.title("📊 Sentiment Polarity Across Wikipedia Sections", fontsize=14)
    plt.xlabel("Section")
    plt.ylabel("Polarity Score (-1 = Negative, 1 = Positive)")
    plt.xticks(rotation=30)
    plt.tight_layout()

    # Save for GitHub
    if not os.path.exists("assets"):
        os.makedirs("assets")
    plt.savefig("assets/sentiment_chart.png")
    plt.show()
```

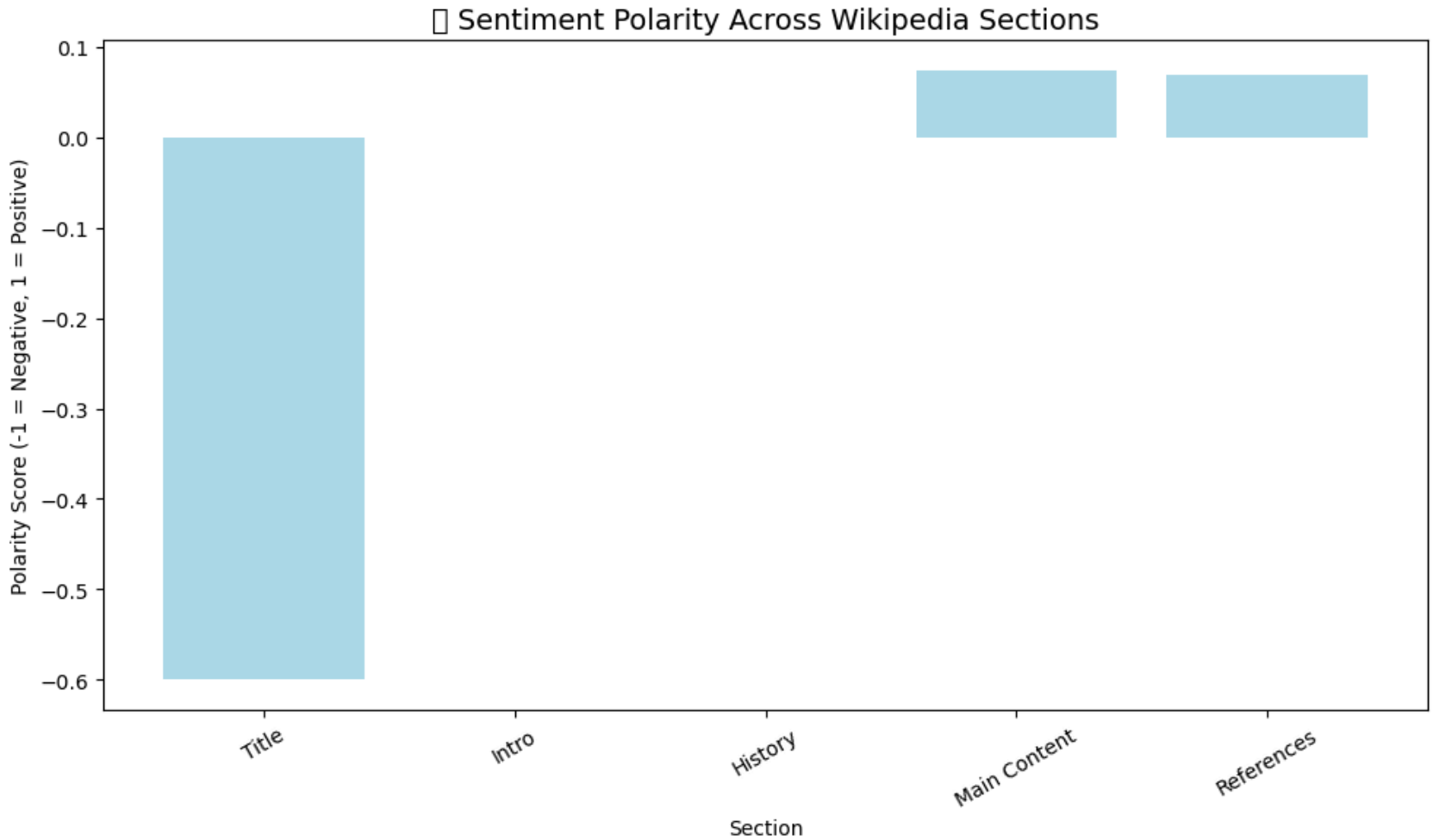
In [8]:
```python
# 🔽 MAIN EXECUTION
url = "https://en.wikipedia.org/wiki/Artificial_intelligence"
html = save_raw_html(url)  # Save and fetch
soup = BeautifulSoup(html, 'html.parser')

sections = extract_sections(soup)
df_sentiments = analyze_sentiments(sections)
print(df_sentiments)

plot_sentiment(df_sentiments)
```

```
        Section  Polarity Score  Subjectivity Score
0         Title       -0.600000            1.000000
1         Intro        0.000000            0.000000
2       History        0.000000            0.000000
3  Main Content        0.073825            0.476540
4    References        0.068733            0.508997

C:\Users\mkhan\AppData\Local\Temp\ipykernel_18976\2815478326.py:9: UserWarning: Glyph 128202 (\N{BAR CHART}) missing from current font.
  plt.tight_layout()
C:\Users\mkhan\AppData\Local\Temp\ipykernel_18976\2815478326.py:14: UserWarning: Glyph 128202 (\N{BAR CHAR
T}) missing from current font.
  plt.savefig("assets/sentiment_chart.png")
C:\ProgramData\anaconda3\Lib\site-packages\IPython\core\pylabtools.py:152: UserWarning: Glyph 128202 (\N{BAR
CHART}) missing from current font.
  fig.canvas.print_figure(bytes_io, **kw)
```



In [9]:
```
# 🏷️ Classify Sentiment Labels
def classify_sentiment(row):
    if row['Polarity Score'] > 0.1:
        return "Positive"
    elif row['Polarity Score'] < -0.1:
        return "Negative"
    else:
        return "Neutral"
```

In [10]:
```
def plot_subjectivity(df):
    plt.figure(figsize=(10, 6))
    plt.bar(df['Section'], df['Subjectivity Score'], color='orange')
    plt.title("🧠 Subjectivity Across Wikipedia Sections", fontsize=14)
    plt.xlabel("Section")
    plt.ylabel("Subjectivity Score (0 = Objective, 1 = Subjective)")
    plt.xticks(rotation=30)
    plt.tight_layout()
    plt.savefig("assets/subjectivity_chart.png")
    plt.show()
```

In [11]:
```
def plot_sentiment_pie(df):
    counts = df['Sentiment Label'].value_counts()
    plt.figure(figsize=(6, 6))
    plt.pie(counts, labels=counts.index, autopct='%1.1f%%', colors=['green', 'gray', 'red'])
    plt.title("📈 Sentiment Distribution")
    plt.savefig("assets/sentiment_pie_chart.png")
    plt.show()
```

In [12]:
```
def add_word_count(df, sections):
    word_counts = [len(sections[section].split()) for section in df['Section']]
    df['Word Count'] = word_counts
```

In [13]:
```
# 🔽 MAIN EXECUTION
url = "https://en.wikipedia.org/wiki/Artificial_intelligence"
html = save_raw_html(url)  # Save and fetch
soup = BeautifulSoup(html, 'html.parser')

sections = extract_sections(soup)
df_sentiments = analyze_sentiments(sections)

# Add Label & Word Count
df_sentiments['Sentiment Label'] = df_sentiments.apply(classify_sentiment, axis=1)
add_word_count(df_sentiments, sections)

# Print Full Analysis Table
print(df_sentiments.sort_values(by='Polarity Score', ascending=False))

# Visualizations
plot_sentiment(df_sentiments)
plot_subjectivity(df_sentiments)
plot_sentiment_pie(df_sentiments)
```

```
        Section  Polarity Score  Subjectivity Score Sentiment Label  \
3  Main Content        0.073825            0.476540         Neutral
4    References        0.068733            0.508997         Neutral
1         Intro        0.000000            0.000000         Neutral
2       History        0.000000            0.000000         Neutral
0         Title       -0.600000            1.000000        Negative

   Word Count
3         703
4         935
1           0
2           0
0           2

C:\Users\mkhan\AppData\Local\Temp\ipykernel_18976\2815478326.py:9: UserWarning: Glyph 128202 (\N{BAR CHAR
T}) missing from current font.
  plt.tight_layout()
C:\Users\mkhan\AppData\Local\Temp\ipykernel_18976\2815478326.py:14: UserWarning: Glyph 128202 (\N{BAR CHAR
T}) missing from current font.
  plt.savefig("assets/sentiment_chart.png")
```

In [ ]: