Unveiling Kyrgyzstan: A Natural Language Processing Journey

Exploring the breathtaking landscapes and rich biodiversity of Kyrgyzstan through text analysis.

Setting the Stage: Importing Libraries

We begin by importing essential Python libraries for data manipulation, regular expressions, and Natural Language Toolkit (NLTK).

```
import pandas as pd
import numpy as np
import re
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import WordNetLemmatizer
from collections import Counter
import matplotlib.pyplot as plt
```

NLTK Resources: The Foundation of Text Analysis

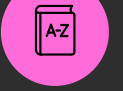
Downloading crucial NLTK packages for tokenization, stop word removal, and lemmatization.

Punkt Tokenizer

For splitting text into sentences and words.

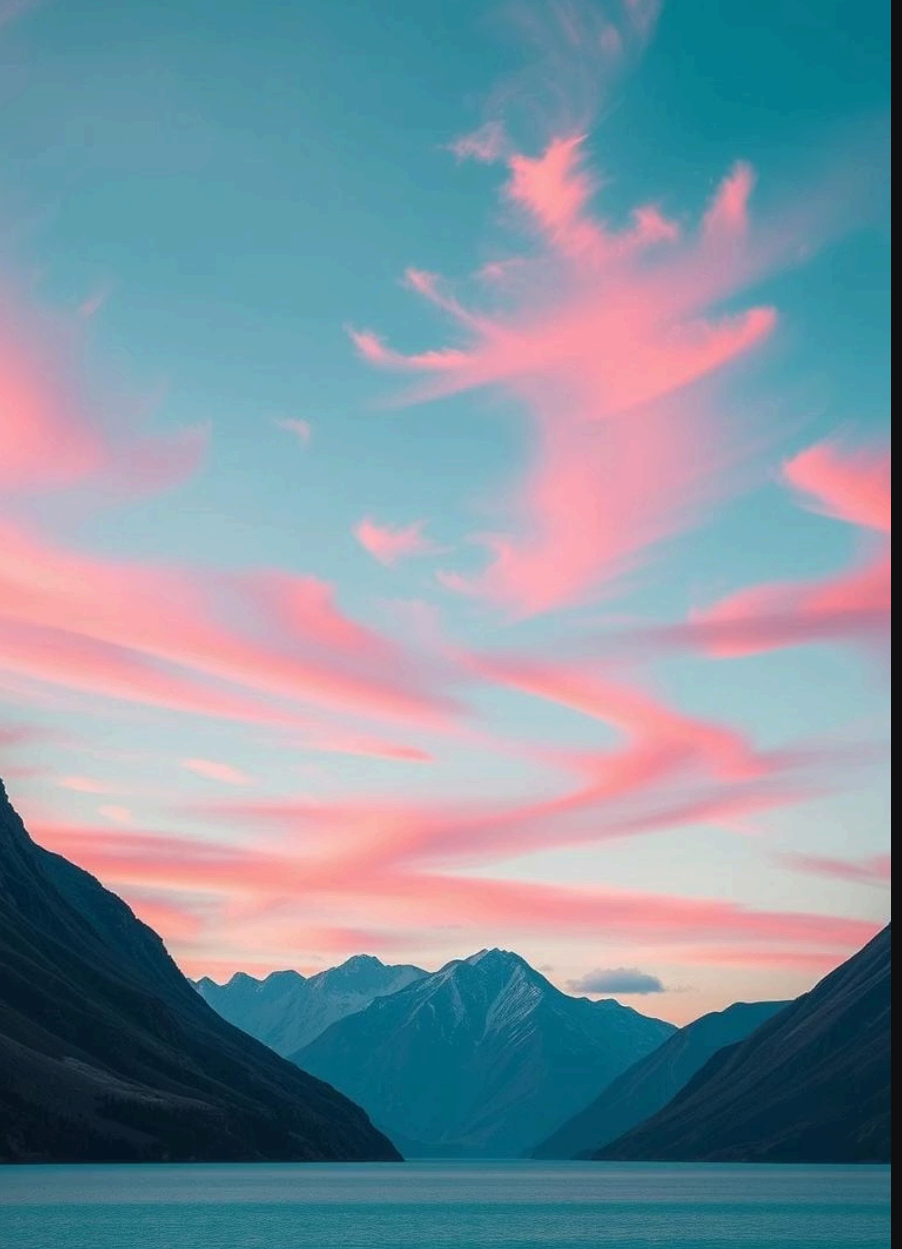
Stopwords

To filter out common, less informative words.



WordNet

For lemmatization, reducing words to their base form.



Loading Our Text: The Beauty of Kyrgyzstan

Our analysis focuses on a descriptive text about Kyrgyzstan's natural wonders, split into individual sentences.

> Kyrgyzstan is a mountainous country known for its breathtaking natural landscapes and untouched wilderness. More than ninety percent of its territory is covered by towering mountain ranges...

The Preprocessing Pipeline: Cleaning the Data

A multi-step process to prepare text for analysis, ensuring accuracy and relevance.

01

Lowercase Conversion

Standardising text to avoid case sensitivity issues.

02

Punctuation & Number Removal

Eliminating non-alphabetic characters for cleaner data.

03

Tokenization

Breaking down sentences into individual words.

04

Stopword Removal

Filtering out common words like 'the', 'is', 'a'.

05

Lemmatization

Reducing words to their root form (e.g., 'mountains' to 'mountain').

Applying Preprocessing: A Clean Dataset

The cleaning function transforms raw text into a list of meaningful tokens for each sentence.

Original Text Snippet	Cleaned Tokens
Kyrgyzstan is a mountainous country known for...	[kyrgyzstan, mountainous, country, known, brea...]
Surrounded by mountains, Issyk-Kul remains ic...	[surrounded, mountain, issykkul, remains, icef...]

N-gram Generation: Uncovering Word Patterns

Extracting unigrams, bigrams, and trigrams to identify frequently co-occurring words and phrases.

```
def get_ngrams(tokens_list, n):
 ngrams = []
 for tokens in tokens_list:
 for i in range(len(tokens)-n+1):
 ngrams.append(tuple(tokens[i:i+n]))
 return ngrams
```

Top N-grams: Key Insights from the Text

Discovering the most frequent single words, two-word phrases, and three-word phrases in our dataset.

Top Unigrams

('kyrgyzstan', 5), ('forest', 5), ('mountain', 4), ('lake', 4)

Top Bigrams

('kyrgyzstan', 'mountainous', 1), ('natural', 'landscape', 1)

Top Trigrams

('kyrgyzstan', 'mountainous', 'country', 1), ('natural', 'landscape', 'untouched', 1)

Visualising Unigrams: The Most Frequent Words

A horizontal bar chart showcasing the top 10 unigrams, highlighting key themes in the text.

Beyond Unigrams: Bigrams and Trigrams

Visualising how words combine to form more complex meanings, offering deeper contextual understanding.

Top 10 Bigrams

Reveals common two-word pairings, like "natural landscape".

Top 10 Trigrams

Highlights three-word sequences, such as "mountainous country known".