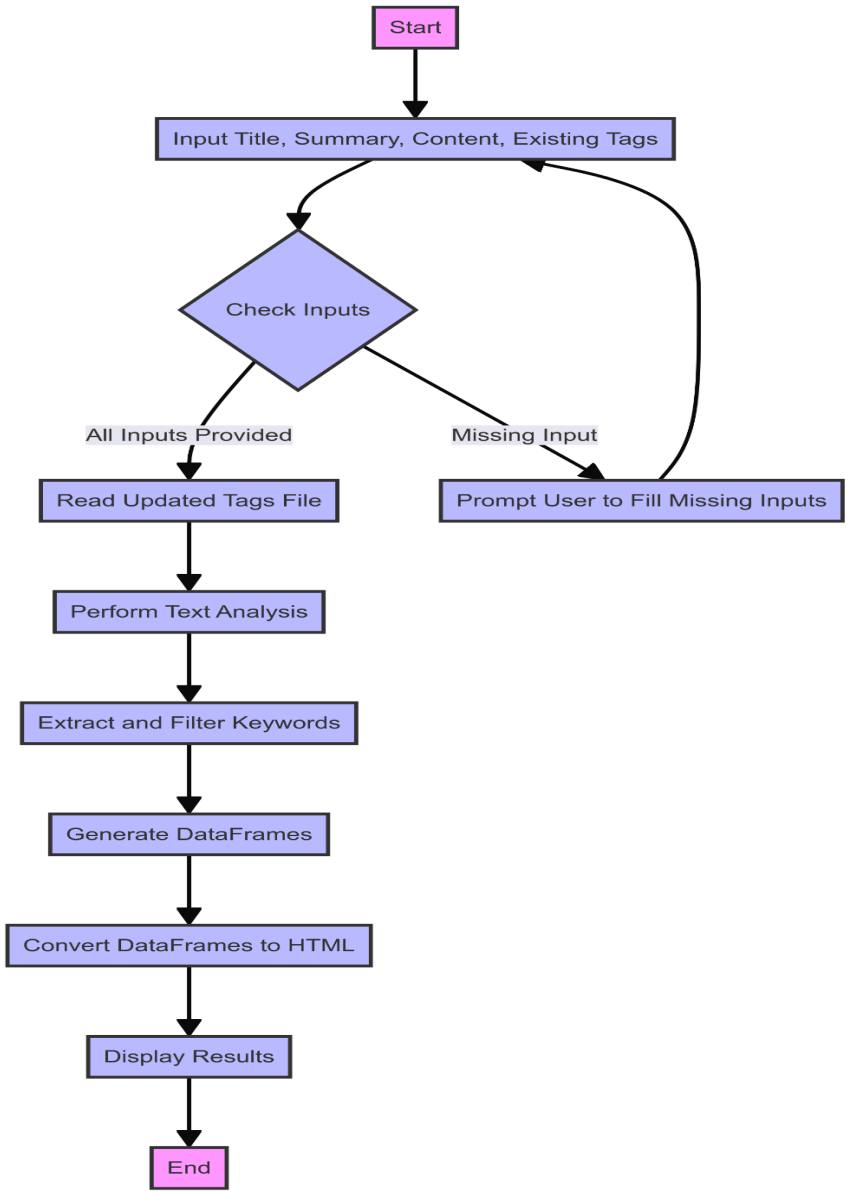
Description:

The code provides a Gradio-based text analysis tool that allows users to input a title, summary, content, and existing tags. It performs comprehensive text analysis using Spacy for Named Entity Recognition (NER), TextBlob for noun phrases, and YAKE for keyword extraction. The analysis includes checking for tag matches and generating HTML tables that summarize the results. These tables include NER tags, existing tags, extracted tags, and a comparison of different keyword extraction methods. The Gradio interface displays these results, allowing users to interactively view and assess the text analysis outcomes.



Libraries Used:

- Gradio for creating web interfaces
- Spacy for NER
- Pandas for data manipulation
- TextBlob for extracting noun phrases
- YAKE for keyword extraction
- Spacy's STOP_WORDS for filtering common words.

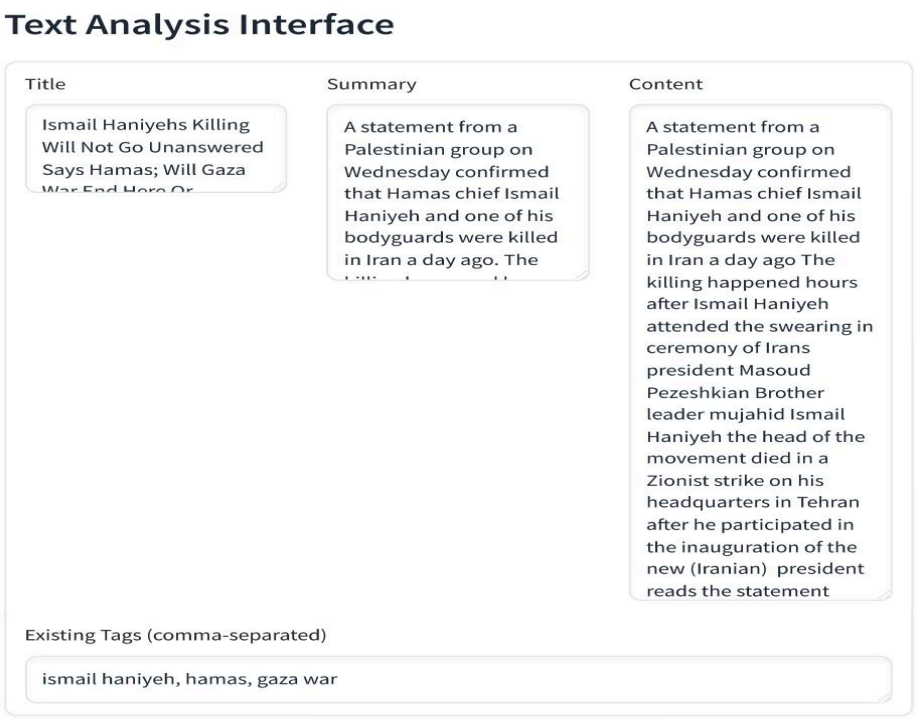
Issues Encountered:

- If the existing tags aren't mentioned in any of the text/input , there is a high possibility of not finding a match of the given tag. **For example:** If sandalwood is an existing tag and that is not mentioned anywhere in the text might be an issue.
- Initially it was very time consuming, but now it will reflect back the output within 120 seconds(maximum).
- There are few tags like cookery, promotion, lottery result etc which required extra attention as exact matches of those tags weren't found in the text.

Idea/Solution:

- As I said above, for a few tags hardcoding is required through which we can achieve matching of existing tags i.e., I am trying to implement based on category (will include category as input along with other inputs).
- **Example:** If the category is partner-content/press-release we get mostly promotion as a tag so when we do hardcoding it'll check for category and if there is no match of existing tag we'll print hardcoded keywords.
- I am considering the unmatched tags from the NER table as the tags that can be suggested/recommended, but again sometimes there are inappropriate keywords. So, as of now I have one comparison table which is printing tags using different libraries which I will eventually remove and try getting 10 suggestable tags.

Snapshots of Gradio Output:



NER Tags	Matched Tags	Unmatched Tags - Suggestable Tags
iran	iran	
the october 7 2023		the october 7 2023
hamas	hamas	
zionist		zionist
ismail haniyehs		ismail haniyehs
irans	irans	
israel	israel	
ismail haniyeh killed		ismail haniyeh killed

NER Tags	Matched Tags	Unmatched Tags - Suggestable Tags
iranian	iranian	
masoud pezeshkian		masoud pezeshkian
hamasrun		hamasrun
palestinian	palestinian	
ismail haniyeh	ismail haniyeh	
israels	israels	
tehran	tehran	

Existing Tags	Found in NER Tags Table	Found in Title/Content
gaza war	No	gaza war
ismail haniyeh	ismail haniyeh	ismail haniyeh
hamas	hamas	hamas

Extracted Tags	Matched Text Tags
gaza war	Title: gaza war
ismail haniyeh	Content: ismail haniyeh, Title: ismail haniyeh, Summary: ismail haniyeh
hamas	Content: hamas, Title: hamas, Summary: hamas

TextBlob Noun Phrases	Spacy Noun Phrases	YAKE Keywords
hamas	Israels health ministry	Irans president Masoud
october	the Palestinian group	Ismail Haniyehs killing
ismail haniyeh killed	the group	Ismail Haniyeh Killed
iranian	the deaths	killing happened hours
palestinian	Wednesday	Hamas chief Ismail
hamas says according	the statement	Brother leader mujahid

Additional Code : XML Content Analysis

Summary: This Python script performs comprehensive analysis on XML content fetched from a specified URL(XML Link). It processes the content using various text processing techniques, including HTML/XML parsing, named entity recognition(NER) with SpaCy, keyword extraction with TextBlob and YAKE, and cleaning of text. The script extracts and highlights named entities (such as persons, organizations, and locations), filters and cleans noun phrases, and identifies keywords. It also reads existing tags from a file, compares them with the extracted tags, and highlights matches and mismatches. Additionally, it evaluates the presence of hardcoded keywords and finds similar phrases within the text. The results are presented through tables and highlighted text to provide a detailed analysis of the XML content.

Below is the short and compressed output of XML Analysis Code:

URL:(https://rss1.oneindia.com/xml4apps/www.oneindia.com/bengaluru/3893859.xml)

Category: bengaluru

NER Matched and Unmatched Tags:

NER Tags	Matched Tags	Unmatched Tags - Suggestable Tags
alok kumar		alok kumar
bengalurumysuru		bengalurumysuru
karnataka	karnataka	

Extracted Tags	Matched Text Tags
fir	fir, fir
drivers	drivers, drivers
violations	Not Found

Fuzzy Matched Tags (With Tags File):

Tag	Fuzzy Matched
fir	fir
drivers	drivers
violations	violations

Comparison of TextBlob Noun Phrases, Spacy Noun Phrases, and YAKE Keywords:

| km speed limit | reckless driving |
|

| bengalurumysuru | road discipline |
|

| fatal accidents | 50 vehicles |
|

| alok kumar adgp | speeding |
|

| concern district police | traffic road safety department |
|

Similar Phrases that are found in Title & Content:

Tag	Similar Phrases Found
fir	firs, fir
violations	violating
drivers	drivers

No hardcoded keywords found.