Text Analysis Pipeline Technical Documentation

Prithvi Singh Dangas

May 21, 2025

1 Metric Calculations

1.1 Core Metrics and Formulas

Metric	Calculation Method
POSITIVE SCORE	Count of words present in `positive-words.txt` dictionary
NEGATIVE SCORE	Count of words present in `negative-words.txt` dictionary
POLARITY SCORE	Dictionary-based: $\frac{\text{Positive Score}-\text{Negative Score}}{\text{Positive Score}+\text{Negative Score}+\varepsilon}$
SUBJECTIVITY SCORE	$\frac{\text{Positive Score}+\text{Negative Score}}{\text{Total Words}}$
AVG SENTENCE LENGTH	$\frac{\text{Total Words}}{\text{Number of Sentences}}$
PERCENTAGE OF COMPLEX WORDS	$\left(\frac{\text{Words with >2 Syllables}}{\text{Total Words}}\right) \times 100$
FOG INDEX	$0.4 \times (\text{Avg Sentence Length} + \%\text{Complex Words})$
AVG WORDS PER SENTENCE	Same as Average Sentence Length
COMPLEX WORD COUNT	Count of words with more than 2 syllables
WORD COUNT	Total number of cleaned words after stopword and punctuation removal
SYLLABLE PER WORD	$\frac{\text{Total Syllables}}{\text{Total Words}}$
PERSONAL PRONOUNS	Count of “I,” “we,” “my,” “ours,” “us” using regex `\b(I|we|my|ours|us)\b`
AVG WORD LENGTH	$\frac{\text{Total Characters in Words}}{\text{Total Words}}$

2 Processing Pipeline

2.1 Step-by-Step Workflow

1. **Initialization Phase**
   - Load stopwords from all files in `StopWords/`
   - Load positive/negative words from `MasterDictionary/`
   - Initialize NLTK’s Punkt tokenizer
2. **Input Handling**
   - Read Excel file using pandas
   - Ensure columns: `URL_ID`, `URL` are present
3. **URL Processing (per row)**
   (a) HTTP GET request with 15s timeout
   (b) Main content extraction using Readability

   (c) HTML cleaning via BeautifulSoup:

```
Remove: <script>, <style>, <header>, <footer>, <nav>
Keep: <p>, <h1>, <h2>, <h3>
```

   (d) Save result to `articles/{URL_ID}.txt`

4. **Text Analysis Phase**
   (a) **Preprocessing**
      - Sentence tokenization using `nltk.sent_tokenize()`
      - Word tokenization using `nltk.word_tokenize()`
      - Clean non-alphabetic tokens, lowercase, and remove stopwords

   (b) **Metric Computation**
      - Sentiment from positive/negative dictionary
      - Readability using syllable counts
      - Regex-based pronoun counting

   (c) **Aggregation**
      - Combine original data with computed metrics
      - Gracefully handle division by zero for empty articles

5. **Output Generation**
   - Save combined DataFrame to Excel using `openpyxl`

3 Error Handling

- **Network Errors**: handled with try/except during HTTP requests
- **Empty Texts**: raise warning if no content extracted
- **Encoding Problems**: use `charset-normalizer` for detection
- **File Errors**: all I/O wrapped with exception handling

4 Solution Architecture

4.1 1. Text Extraction Module

Library	Version	Purpose
requests	≥2.26.0	Web request with timeout handling
readability-lxml	≥0.8.1	Extract main content from HTML
BeautifulSoup4	≥4.10.0	Remove unwanted tags
charset-normalizer	≥2.0.0	Auto-detect and normalize encoding

4.2 2. Linguistic Analysis Engine

Library	Version	Purpose
nltk	≥3.6.0	Tokenization, segmentation
textstat	≥0.7.0	Readability and syllable stats
TextBlob	≥0.15.3	Sentiment polarity and subjectivity
re	built-in	Pattern matching for pronouns

4.3 3. Data Management

Library	Version	Purpose
pandas	≥1.3.0	DataFrame manipulation and Excel I/O
openpyxl	≥3.0.9	Backend for writing Excel files
tqdm	≥4.62.0	Progress bars for processing loop

5 Execution Guide

5.1 Directory Structure

- Directory structure:

```
Root Folder
├── StopWords/
│   ├── StopWords_Auditor.txt
│   ├── StopWords_Currencies.txt
│   ├── StopWords_DatasandNumbers.txt
│   ├── StopWords_Generic.txt
│   ├── StopWords_GenericLong.txt
│   ├── StopWords_Geographic.txt
│   └── StopWords_Names.txt
├── MasterDictionary/
│   ├── positive-words.txt
│   └── negative-words.txt
├── Input.xlsx
├── articles/ (auto-created)
├── text_analysis.py
└── requirements.txt
```

- Required Python version: 3.8+

5.2 Run Commands

```
# Install dependencies
pip install -r requirements.txt

# Run main script
python text_analysis.py --input Input.xlsx --output final_output.xlsx
```

6 Dependencies

```
# requirements.txt
pandas==1.3.5
requests==2.26.0
beautifulsoup4==4.10.0
readability-lxml==0.8.1
nltk==3.6.7
textstat==0.7.0
textblob==0.15.3
tqdm==4.62.3
charset-normalizer==2.0.12
openpyxl==3.0.10
python-docx==0.8.11
```