Data Science & AI for Economists

Lecture 11: Text Analysis(I)

Zhaopeng Qu
Business School, Nanjing University

December 04 2025

Roadmap

Today's Agenda

Part I: Basic Concepts

- What is Text Analysis?
- Why Text Analysis for Economists?
- Applications in Economics
- Basic Terminology
- Text Preprocessing
- Chinese Text Analysis Challenges
- Bag of Words & DTM
- TF-IDF

Part II: Advanced Topics

- Sentiment Analysis
- Topic Modeling (LDA)
- Text Classification
- Word Embeddings
- Information Extraction
- Python Tools for Text Analysis

Introduction to Text Analysis

What is Text Analysis?

Text Data is Everywhere

Text data is one of the richest sources of unstructured information:

- **Product Reviews**: Taobao, JD, Amazon user reviews
- **Social Media**: Weibo, Twitter, WeChat public accounts
- **News Media**: Financial news, policy announcements
- **Government Documents**: Government work reports, policy documents, laws and regulations
- **Academic Literature**: Paper abstracts, research reports
- **Corporate Data**: Annual reports, earnings call transcripts

What is Text Analysis?

Characteristics of Text Data

- Text data is far more complex and far from perfect than traditional data.
    - High-dimensional
    - Sparse with many zeros
    - unstructucted
    - noisy
    - sample selection bias

What is Text Analysis?

Definition of Text Analysis

Text Analysis is a set of methods and techniques to process and analyze text data to extract useful information.

Also known as:

- **Text Mining** (文本挖掘)
- **Natural Language Processing, NLP** (自然语言处理)

- **Computational Text Analysis** (计算文本分析)

- Text analysis is a powerful tool for economists to analyze and understand the world.

Why Text Analysis for Economists?

Limitations of Traditional Economic Data

Traditional Data	Limitations
Macro indicators (GDP, CPI)	Publication lag, low frequency
Survey data	High cost, limited sample
Administrative data	Difficult to obtain, privacy restrictions

Advantages of Text Data

- **Real-time**: Can obtain near real-time information
- **Large scale**: Massive text data available on the internet
- **Rich dimensions**: Contains soft information such as sentiment, opinions, expectations
- **Low cost**: Much lower acquisition cost compared to traditional surveys

What Can Text Analysis Do?

Task	Description
Descriptive Analysis	Word frequency statistics, word clouds, keyword extraction
Information Extraction	Extract structured information (names, places, events)
Sentiment Analysis	Determine emotional tone (positive/negative/neutral)
Topic Modeling （Summarization）	Discover latent topics in document collections
Text Classification （Classification）	Categorize texts into predefined categories

Applications:Economic Policy Uncertainty Index

Baker, Bloom, and Davis (QJE 2016)

- **Research Question**: How to quantify economic policy uncertainty?

1. For each newspaper on each day since 1985, submit the following query:
    - Article contains `uncertain` or `uncertainty`, AND
    - Article contains `economic` or `economy`, AND
    - Article contains `congress` or `deficit` or `federal reserve` or `legislation` or `regulation` or `white house`
2. Normalize resulting article counts by total newspaper articles that month

Result: Constructed monthly economic policy uncertainty index dating back to 1985

Other Applications in Economics

Corporate Annual Report Text Analysis

- Analyze risk disclosure language in annual reports
- Detect linguistic features of financial fraud
- Predict future company performance

Consumer Sentiment Analysis

Data Sources: Product reviews, social media posts

Applications:

- Real-time tracking of consumer confidence
- Predict product sales
- Monitor brand reputation

Basic Terminology for Text Analysis

Core Concepts

- **Token (词元)**: The smallest unit of text in a document (a word or character)
- **Type (词型)**: The set of all unique tokens in a document
- **Document**: A collection of tokens
- **Corpus (语料库)**: A collection of documents
- **Metadata**: Information about the document (author, date, source)

Example

Text: "A rose is a rose is a rose."

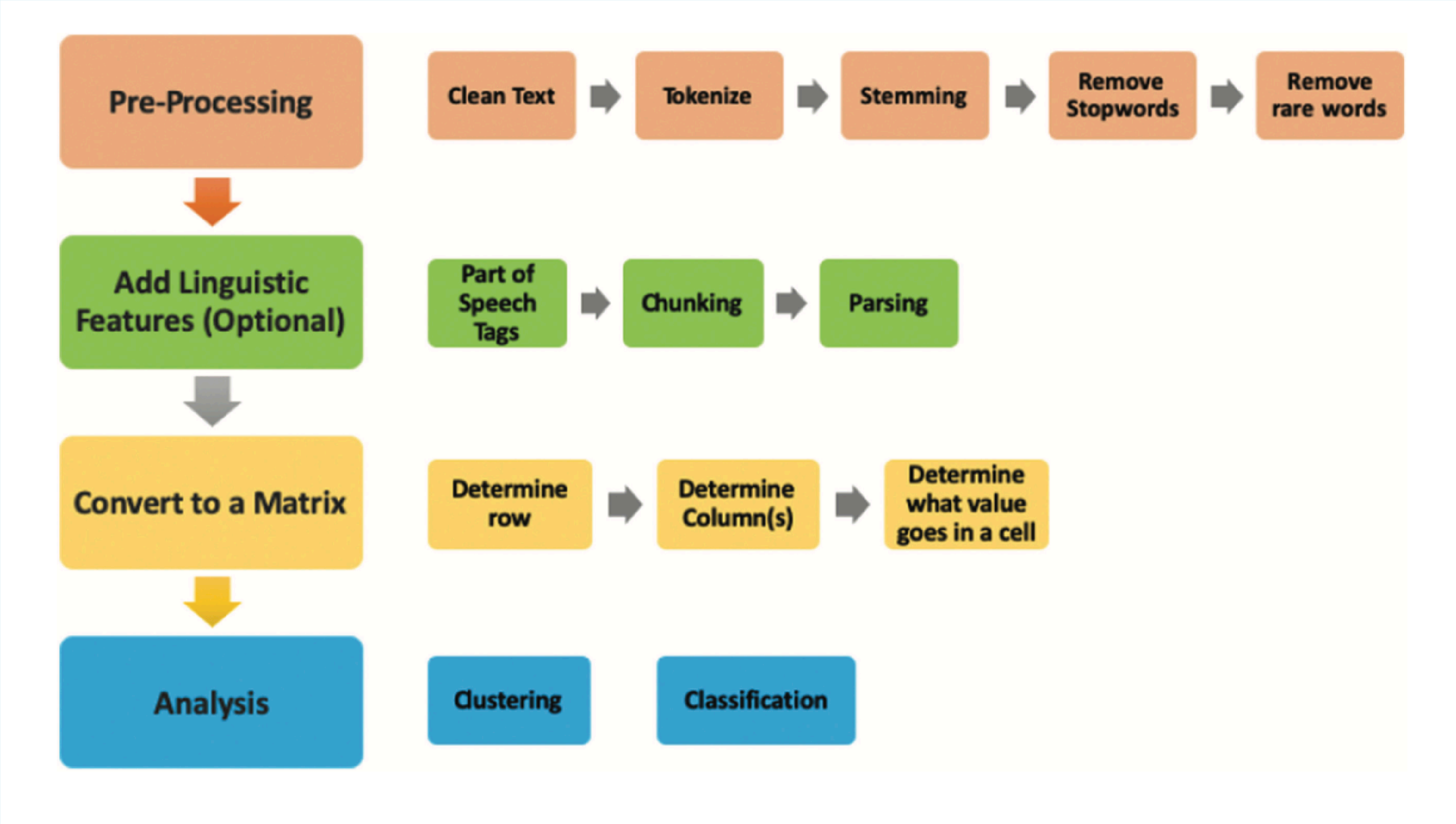
- **Tokens**: 9 $\rightarrow$ [A, rose, is, a, rose, is, a, rose, .]
- **Types**: 4 $\rightarrow$ {A, rose, is, a}
- So `tokens` is a vector variable with 9 elements
- `types` is a factor variable with 4 levels for tokens.

Core Concepts

File Formats

- **XML/JSON**: Two most common file formats for text data
- More structured than `.txt`
- Include metadata
  - includes the metadata of the document
  - author, date, source, etc.

Text Preprocessing Pipeline



Preprocessing Steps

- `raw text` → `tokenization` → `lowercasing` → `remove punctuation` → `remove stopwords` → `stemming/lemmatization` → `clean text`

(1) Tokenization (分词)

- **English**: Relatively simple, split by spaces and punctuation
- **Chinese**: No natural separators, requires specialized segmentation tools

(2) Lowercasing (小写化)

- Convert all letters to lowercase to reduce vocabulary size

Text Preprocessing Pipeline

Preprocessing Steps

(3) Remove Punctuation

- Delete punctuation marks

```
"I love economics!" → "I love economics"
```

(4) Remove Stopwords (去除停用词)

- **English**: the, is, at, which, on, a, an, ...
- **Chinese**: 的, 是, 在, 了, 和, 与, ...

Text Preprocessing Pipeline

Preprocessing Steps

(5) Stemming vs. Lemmatization

Stemming (词干提取): Crude chopping of word endings

```
running, runs, ran → run
studies → studi (may not be a real word)
```

Lemmatization (词形还原): Reduce to dictionary form

```
better → good
running → run
```

Text Preprocessing Pipeline

Preprocessing Steps

(6) N-grams: A sequence of n consecutive words

- **Unigram (1-gram)**: Single words → ["I", "love", "economics"]
- **Bigram (2-gram)**: Two consecutive words → ["I love", "love economics"]
- **Trigram (3-gram)**: Three consecutive words → ["I love economics"]

Why N-grams?

- Preserve word order information
- Capture phrases and collocations
- Example: `New York` is more meaningful than `New` + `York` separately

Special Challenges for Chinese Text

(1) Encoding Issues (编码问题)

Chinese characters may be encoded in different formats:

- **GB2312/GBK**: Commonly used in Windows systems
- **UTF-8**: International standard, **recommended**
- **Suggestion**: Always convert text to `UTF-8` encoding first

Special Challenges for Chinese Text

(2) Tokenization Challenges

Chinese has no natural word boundaries, requiring specialized segmentation algorithms.

Example: "南京市长江大桥"

- Option 1: "南京市" / "长江大桥"
- Option 2: "南京" / "市长" / "江大桥"

Common Segmentation Tools in Python:

- **jieba**: Most popular Chinese segmentation library
- **pkuseg**: Developed by Peking University
- **THULAC**: Developed by Tsinghua University

Special Challenges for Chinese Text

(3) Chinese Stopwords

Need to use specialized Chinese stopword lists. Common stopwords include:

- 的、了、是、在、和、与、或
- 这、那、它、他、她
- 因为、所以、但是、而且

Python Library: `stopwords-zh` or `jieba.analyse`

Special Challenges for Chinese Text

(4) Sentiment Analysis for Chinese

Chinese sentiment analysis models are different from English models.

Python Libraries: `snownlp`, `textblob`, or `transformers` (Chinese models)

From Text to Data: Bag of Words

Bag of Words (BoW) Model

Core Idea: Represent text as a collection of words, ignoring word order and grammar.

Two Common Data Structures:

1. **Hash table/Key-value pairs**: {word: count}
    - Keys: words
    - Values: counts in the document
    - Easy to add new entries

1. **Document-Term Matrix (DTM)**: Rows = documents, Columns = words
    - Each cell is the count of the word in the document
    - Each column is a word in the corpus

From Text to Data: Bag of Words

Document-Term Matrix (DTM)

- **i-th document:N**: Number of documents (rows)

- **j-th word:D**: Size of dictionary/vocabulary (columns)

- **X_{ij}**: Number of times term/word j appears in document i

Characteristics:

- Matrix is usually very sparse (mostly zeros)
- Dimensions can be very high (huge vocabulary)

Document-Term Matrix (DTM)

Example 5.1. Two example sentences that include the word manufacturing from the SOTU corpus.

> Doc 1: It is undoubtedly in the power of Congress seriously to affect the agricultural and manufacturing interests of France by the passage of laws relating to her trade with the United States. (President Jackson, 1831)
>
> Doc 2: And this Congress should make sure that no foreign company has an advantage over American manufacturing when it comes to accessing financing or new markets like Russia. (President Obama, 2012)

	undoubtedly	power	congress	seriously	affect	agricultural	manufacturing	interests	france	passage	laws	relating	trade	united	states	make	sure	foreign	company	advantage	american	comes	accessing	financing	new	markets	like	russia
Doc 1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Doc 2	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1

Measures of Word Importance: TF-IDF

Problem: Word Counts Are Not Directly Comparable

- Longer documents have more words than shorter documents
- Common words (e.g., "the", "is") appear frequently in all documents

Solution: We need to normalize word counts.

Measures of Word Importance: TF-IDF

Term Frequency

Term frequency（词频） is a measure of how frequently a term occurs in a document.

$$TF_{ij} = \frac{X_{ij}}{\sum_{j=1}^{D} X_{ij}}$$

- Word j's count in document i, divided by total words in document i
- Normalizes for document length

Measures of Word Importance: TF-IDF

Inverse Document Frequency (IDF)

Inverse document frequency（逆文档频率） measures *how important a term is across documents*.

$$\mathrm{IDF}_j = \log\left(1 + \frac{N}{M_j}\right)$$

- N: Total number of documents
- M_j: Number of documents containing term j
  - If a word appears in many documents $\rightarrow$ low IDF
  - If a word appears in few documents $\rightarrow$ high IDF

Measures of Word Importance: TF-IDF

TF-IDF(词频-逆文档频率)

Term frequency-inverse document frequency (TF-IDF)

$$
\begin{aligned}
\mathrm{TF}-\mathrm{IDF}_{ij} &= \mathrm{TF}_{ij} \cdot \mathrm{IDF}_j \\
&= \frac{X_{ij}}{\sum_{j=1}^{D} X_{ij}} \cdot \log\left(1 + \frac{N}{M_j}\right)
\end{aligned}
$$

Intuition:

- A word appears frequently in a document → high TF
- But this word rarely appears in other documents → high IDF
- High TF-IDF → This word is important for this document

Measures of Word Importance: TF-IDF

Applications of TF-IDF

1. 文档聚类 (Document Clustering)

- 使用 TF-IDF 向量计算文档相似度
- 将相似文档分组，发现文档集合中的主题结构

2. 文本分类 (Text Classification)

- 新闻分类（体育、商业、科技等）
- 垃圾邮件检测
- 情感分类（正面/负面）

典型经济学应用

- 招聘广告分类：将招聘广告的职位描述分类为不同的职位类别

Python Tools for Text Analysis

Essential Python Libraries

Library	Purpose	Features
pandas	Data manipulation	Easy to use, integrates with other libraries
nltk	Natural language processing	Comprehensive, widely used, educational
spaCy	Industrial NLP	Fast, production-ready, supports multiple languages
jieba	Chinese word segmentation	Simple and effective
stopwords-zh	Chinese stopwords	Chinese stopword lists
snownlp	Chinese sentiment analysis	Chinese sentiment models
transformers	Pre-trained models	BERT, GPT, state-of-the-art models
gensim	Topic modeling	LDA, Word2Vec implementation
scikit-learn	Machine learning	TF-IDF, classification, clustering
wordcloud	Word cloud visualization	Beautiful word clouds

Text Analysis in Python