Introduction to Text Mining

Techniques, Applications, and Practical Approaches

Dr. Maryam Movahedifar

4-6 December 2024

University of Bremen, Germany
movahedm@uni-bremen.de

Outline

Goal of the Course

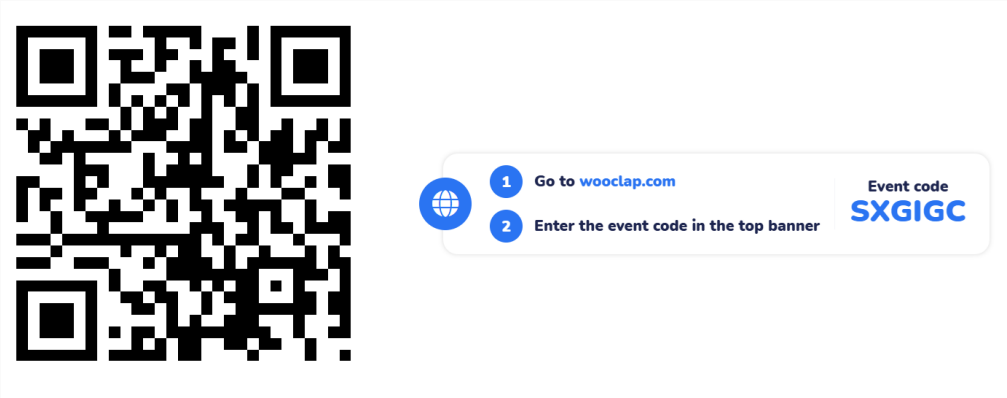
Goal of the Course

- Introduce key concepts and techniques in text mining.
- Provide practical skills for applying text mining techniques using Python.
- Enable participants to analyze and derive insights from unstructured text data.

How Familiar Are You with Python?

How Familiar Are You with Python?

- Please scan the QR code below to respond to the question.



Latest Python Version and Resources

- **Latest Python Version:** Python 3.11 (as of November 2024).
- **Resources for Python Learning:**
  - Python for Beginners
  - Python Language Reference
  - Python Documentation

Introduction to Text Mining

What is Text Mining?

- Text mining is the process of extracting meaningful information and patterns from large volumes of unstructured text data.
- It is commonly used for sentiment analysis (determining the sentiment behind a text), topic discovery (identifying themes in a corpus of documents), and document categorization (classifying text into predefined categories).
- Applications range across domains like social media (e.g., analyzing user sentiments), healthcare (e.g., extracting information from medical texts), finance (e.g., analyzing news for stock predictions), and legal fields (e.g., reviewing contracts or legal documents).

Text Mining in an Example

Let's see an example of how text mining can be used in practice:

- **Garry** works at Bol.com (a webshop in the Netherlands).
- He works in the **Customer Relationship Management** department.
- He uses **Excel** to read and search customer reviews, extract aspects they wrote their reviews on, and identify their sentiments.

Garry's task involves analyzing customer feedback, like these examples:

Example 1: Customer Review

> *This is a nice book for both young and old. It gives beautiful life lessons in a fun way. Definitely worth the money!*

- **Aspects:** Educational, Funny, Price
- **Sentiment:** Positive

Garry uses text mining to automatically detect aspects (e.g., *Educational, Funny*) and sentiments (e.g., *Positive*) from customer reviews.

Example 2: Customer Review

Nice story for older children. Funny, easy to read, and great for bedtime stories.

- **Aspects:** Funny, Readability
- **Sentiment:** Positive

By processing the text, Garry can quickly extract useful insights that can help his team improve customer experience.

Challenges in Text Mining

One of the biggest challenges in text mining is that language is complex and hard.

- Different things can mean more or less the same (e.g., *"data science"* vs. *"statistics"*).
- Context dependency (e.g., *"You have very nice shoes"*).
- Same words with different meanings (e.g., *"to sanction"*, *"bank"*).
- Irony and sarcasm (e.g., *"That's just what I needed today!"*).
- Figurative language (e.g., *"He has a heart of stone"*).
- Negation and spelling variations ((e.g., "not good" vs. "good") and (e.g., "color" vs. "colour")).

Text Mining Definition

- **Definition (Hearst, 1999):** *"The discovery by computer of new, previously unknown information, by automatically extracting information from different written resources."*
- Text mining is about looking for patterns in text, similar to how data mining is used to find patterns in data.
- It involves a set of linguistic, statistical, and machine learning techniques that model and structure the information content of textual sources (Wikipedia).

Text Mining Can Be Quite Effective!

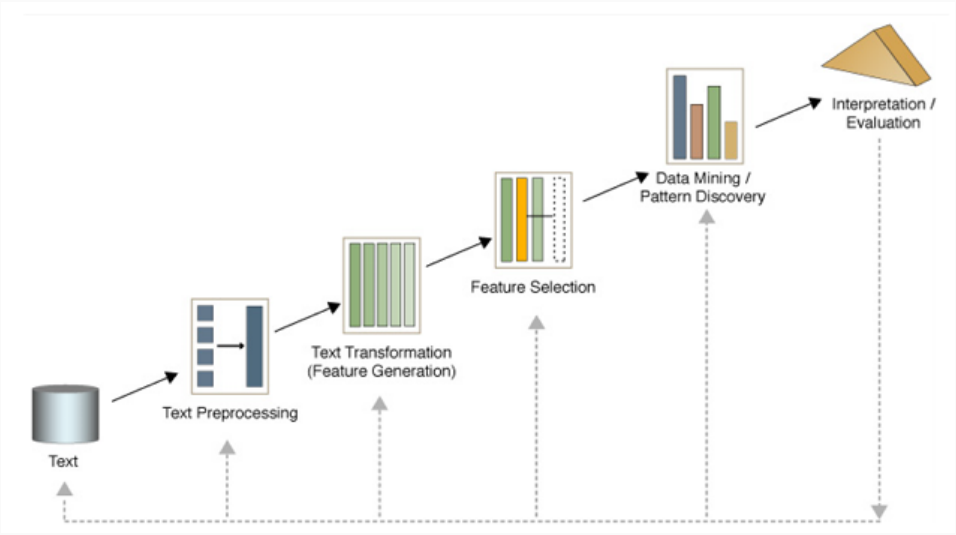
- We won't solve all linguistic problems...
- Despite the challenges, text mining can still provide valuable insights.

Text Mining Process

Key Steps in Text Mining

- Data Collection: Gathering raw text from various sources (web scraping, APIs, files).
- Preprocessing: Cleaning and normalizing text (e.g., removing stopwords, punctuation, converting to lowercase).
- Feature Extraction: Converting text into a numerical format (e.g., using TF-IDF, Bag of Words).
- Modeling: Applying machine learning algorithms to analyze or classify text.
- Evaluation: Assessing model performance with metrics such as accuracy, precision, and recall.

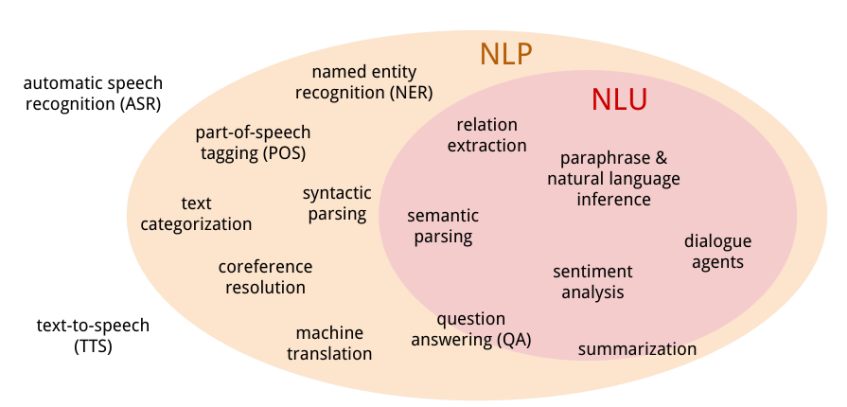
Text Mining Process



Text Mining Tasks

- Text Classification
- Text Clustering
- Sentiment Analysis
- Feature Selection
- Topic Modeling
- Responsible Text Mining
- Text Summarization

Terminology: NLU vs. NLP vs. ASR



Source: Stanford NLU Paper

Text Preprocessing Techniques

Text Preprocessing

- Text preprocessing is an approach for cleaning and noise removal of text data.
- It brings your text into a form that is analyzable for your task.
- It transforms text into a more digestible form so that machine learning algorithms can perform better.

Typical Steps in Text Preprocessing

Typical Steps:

- **Tokenization:** Split text into words or tokens (e.g., “text”, “ming”, “is”, “the”, “best”, “!”)
- **Stemming:** (“lungs” $\rightarrow$ “lung”) or **Lemmatization:** (“were” $\rightarrow$ “is”)
- **Lowercasing:** (“Disease” $\rightarrow$ “disease”)
- **Stopword Removal:** (“text ming is best!” becomes “text ming best”)

More Steps:

- **Punctuation Removal:** (“text ming is the best!” becomes “text ming is the best”)
- **Number Removal:** (“I42” $\rightarrow$ “I”)
- **Spell Correction:** (“hart” $\rightarrow$ “heart”)

Note: Not all of these are appropriate at all times!

Tokenization

- Tokenization: Splits text into words, phrases, or other meaningful elements.
- Example:
  - Text: “Text mining is a useful technique.”
  - Tokens: [“Text”, “mining”, “is”, “a”, “useful”, “technique”]

Stemming and Lemmatization

- Stemming: Reduces words to their root form by trimming suffixes.
  - Example: "running" $\rightarrow$ "run", "bikes" $\rightarrow$ "bike"
- Lemmatization: More advanced; converts words to their dictionary forms based on context.
  - Example: "better" $\rightarrow$ "good" (Lemmatization considers grammatical structure).

Removing Stopwords and Punctuation

- Stopwords: Commonly used words (e.g., "the", "is") that are often removed to reduce noise.
- Punctuation Removal: Helps to clean up text, although in some cases (e.g., sentiment analysis), punctuation can carry meaning.

Definitions: N-grams and POS Tagging

N-grams: A contiguous sequence of N tokens from a given piece of text.

- Example: *"Text mining is to identify useful information."*
- **Bigrams:** *"text_mining"*, *"mining_is"*, *"is_to"*, *"to_identify"*, *"identify_useful"*, *"useful_information"*, *"information_"*.
- **Pros:** Capture local dependency and order.
- **Cons:** Increase vocabulary size, leading to sparsity.

Part of Speech (POS) Tagging:

- Annotates each word in a sentence with a part-of-speech (e.g., noun, verb, adjective).
- Helps with understanding sentence structure and clarifying the meaning of words.

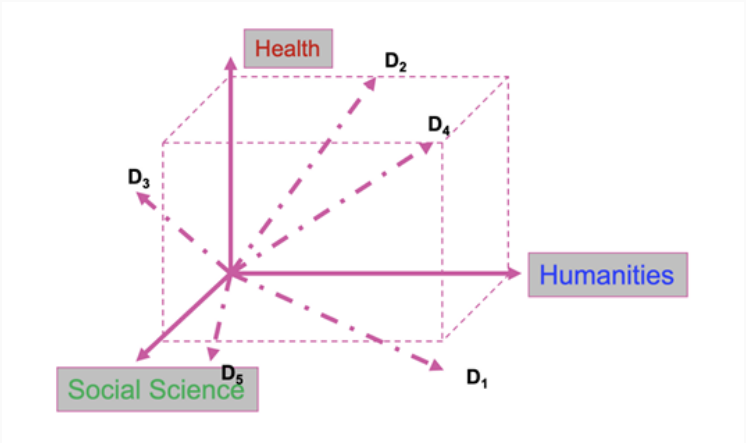
Feature Extraction and Vector Space Model

Vector Space Model

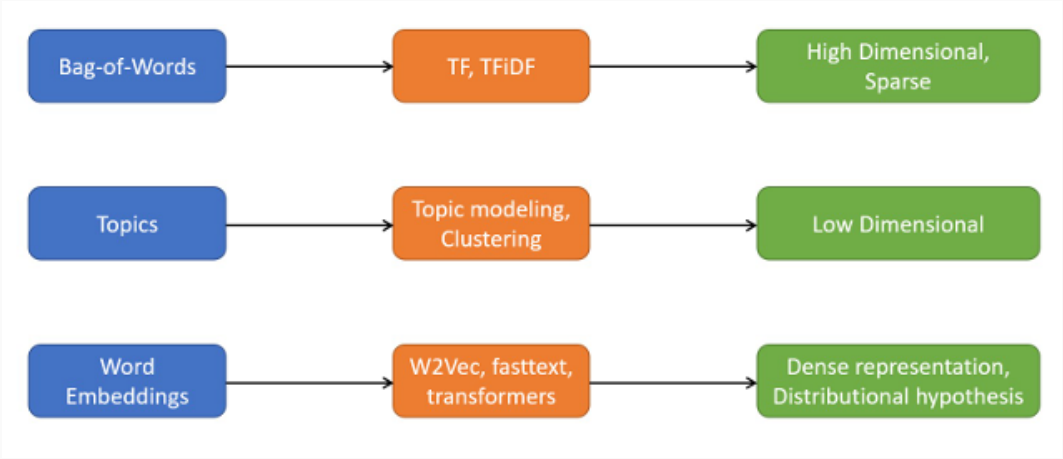
- Basic Idea:
  - Text is unstructured data that must be converted into a structured format for analysis.
  - A document can be represented in a way that computers can process.
- Representation Methods:
  - String representation: Lacks semantic meaning.
  - Sentence list: Treats each sentence as a short document.
  - Vector representation: Documents represented as ordered lists of numbers.

An Illustration of VS Model

All documents are projected into this concept space:



VSM: How do we represent vectors?



Bag of Words (BoW)

- BoW Model:
  - Represents documents by counting term occurrences.
  - Terms can include single words or n-grams (e.g., "text mining").
- Weighting Schemes:
  - Binary: Marks term presence or absence.
  - Term Frequency (TF): Counts term frequency within a document.
  - Term Frequency-Inverse Document Frequency (TF-IDF): Weighs terms based on document-level significance.

Bag of Words (BoW) - Binary Example

- Shows term presence (1) or absence (0) in each document.
- Example Documents:
  - Doc1: "Text mining is to identify useful information."
  - Doc2: "Useful information is mined from text."
  - Doc3: "Apple is delicious."

	text	information	identify	mining	mined	is	useful	to	from	apple	delicious
Doc1	1	1	1	1	0	1	1	1	0	0	0
Doc2	1	1	0	0	1	1	1	0	1	0	0
Doc3	0	0	0	0	0	1	0	0	0	1	1

TF-IDF and Document-Term Matrix (DTM)

- Term Frequency-Inverse Document Frequency (TF-IDF):
  - Formula:

$$\text{TF-IDF} = \text{TF} \times \log\left(\frac{\text{Total Documents}}{\text{Documents with Term}}\right)$$

- Document-Term Matrix (DTM):
  - Each document is represented as a vector of term weights.
  - Provides a basis for similarity calculations and further processing.

Bag of Words (BoW) - TF Example

- In TF,a term is more important if it occurs more frequently in a document
- Example Documents:
  - Doc1: "And God said, Let there be light: and ther was light."
  - Doc2: "And God saw the light, that it was good: and God devided the light from the darkness.
  - Doc3: "And God called the light Day, and the darkness he called Night, And the evening and the morning were the first day."

"Document - Term matrix" (DTM) (raw word counts)

	light	god	darkness	called	day	let	said	divided	good	saw	evening	first	morning	night
d1	2	1	0	0	0	1	1	0	0	0	0	0	0	0
d2	2	2	1	0	0	0	0	1	1	1	0	0	0	0
d3	1	1	1	2	2	0	0	0	0	0	1	1	1	1

Bag of Words (BoW) - TF-IDF Example

- In TF-IDF, a term is more discriminative if it occurs a lot but only in fewer documents:
- Example Documents: Same as last one

"Document - Term matrix" (DTM) (tf-idf)

	light	god	darkness	called	day	let	said	divided	good	saw	evening	first	morning	night
d1	0	0	0.000	0.0	0.0	1.1	1.1	0.0	0.0	0.0	0.0	0.0	0.0	0.0
d2	0	0	0.405	0.0	0.0	0.0	0.0	1.1	1.1	1.1	0.0	0.0	0.0	0.0
d3	0	0	0.405	2.2	2.2	0.0	0.0	0.0	0.0	0.0	1.1	1.1	1.1	1.1

Definition: Similarity Metrics in Vector Space Model

- Euclidean Distance:
  - Measures straight-line distance between document vectors.
  - Penalizes longer documents.
  - Formula:

$$\text{dist}(d_i, d_j) = \sqrt{\sum_{t \in V} (\text{tf}(t, d_i) \cdot \text{idf}(t) - \text{tf}(t, d_j) \cdot \text{idf}(t))^2}$$

- Cosine Similarity:
  - Measures the cosine of the angle between vectors, focusing on their overlap.
  - Formula:

$$\cos(d_i, d_j) = \frac{\vec{d_i} \cdot \vec{d_j}}{|\vec{d_i}| \times |\vec{d_j}|}$$

Applications of Text Mining

Sentiment Analysis

- Sentiment Analysis: Determines if a text conveys positive, negative, or neutral sentiment.
- Applications: Customer reviews, social media analysis, brand monitoring.

Topic Modeling

- Topic Modeling: Discovers themes or topics within document collections.
- Common Algorithms: Latent Dirichlet Allocation (LDA), Non-negative Matrix Factorization (NMF).

Document Clustering and Classification

- Document Clustering: Groups similar documents without predefined labels.
- Document Classification: Categorizes documents based on content.

Conclusion

Conclusion

- Text mining turns unstructured data into meaningful insights.
- Vital for social media, customer insights, healthcare, and more.
- Involves natural language processing (NLP) and machine learning techniques.

Practical 1