

Introduction to Text Mining

Techniques, Applications, and Practical Approaches

Dr. Maryam Movahedifar

14-17 July 2025

University of Bremen, Germany

movahedm@uni-bremen.de



Universität
Bremen



DATA SCIENCE
CENTER

Goal of the Course

Introduction to Text Mining

Text Mining Process

Text Preprocessing Techniques

Feature Extraction and Vector Space Model

Summary

Goal of the Course

Understand Core Concepts

Learn the fundamental ideas behind text mining and natural language processing.

Extract Insights from Text

Learn how to turn unstructured text into useful information and analysis.

Gain Practical Python Skills

Apply text mining techniques hands-on using real-world Python tools.

Latest Python Version and Resources

- **Latest Python Version:** Python 3.13.5
- **Resources for Python Learning:**
 - [Python for Beginners](#)
 - [Python Language Reference](#)
 - [Python Documentation](#)

Introduction to Text Mining

What is Text Mining?

Extract Meaning from Text

Turn unstructured text into meaningful patterns and insights.

Common Tasks




Sentiment analysis, topic discovery, document classification.

Real-World Applications

- Social media: analyze user opinions
- Healthcare: mine medical records
- Finance: detect trends in news
- Legal: scan and sort legal documents





Let's look at how text mining works in a real-world setting:

-  Garry works at [Bol.com](https://www.bol.com), a major online store in the Netherlands.
-  He's part of the Customer Relationship Management (CRM) team.
-  He uses Excel to analyze customer reviews — finding out what people talk about and how they feel.

Garry's job is to turn free-text feedback into useful business insights.

“ 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

 Text mining helps Garry automatically detect these aspects and sentiments.

“ Example 2: Customer Review

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

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

💡 Garry uses these insights to improve customer experience and spot trends.



Challenges in Text Mining

Language is powerful — but also tricky!

- 📖 Synonyms confuse (e.g., *“data science”* vs. *“statistics”*).
- ” Meaning depends on context (e.g., *“You have very nice shoes”*).
- ? Words can mean different things (e.g., *“bank”*, *“sanction”*).
- 😏 Irony and sarcasm flip the meaning (e.g., *“That’s just what I needed today!”*).
- ✂ Figurative speech is not literal (e.g., *“heart of stone”*).
- ⊘ Negation and spelling variations (e.g., *“not good”*, *“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.”

What does this mean in practice?


- Text mining is about looking for patterns in text, just like data mining finds patterns in structured data.
- It combines linguistic, statistical, and machine learning methods to model and interpret textual information.


Text Mining Can Be Quite Effective!


- It won't solve every linguistic nuance...
- But it still uncovers valuable insights from text data.


Text Mining Process


Key Steps in Text Mining

 **Data Collection:** Gathering raw text from various sources (web scraping, APIs, files).

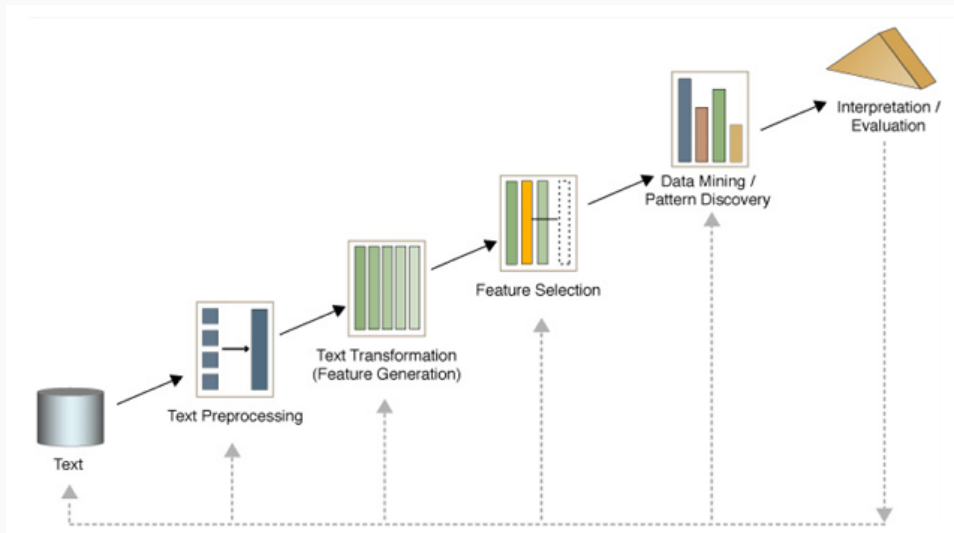
 **Preprocessing:** Cleaning and normalizing text (e.g., removing stopwords, punctuation, converting to lowercase).

 **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:** Assigning categories to text
- 📁 **Text Clustering:** Grouping similar texts
- 😊 **Sentiment Analysis:** Detecting emotions/opinions
- 🔧 **Feature Selection:** Picking important text features
- 💡 **Topic Modeling:** Discovering main themes
- ⚖️ **Responsible Text Mining:** Ethical and fair use
- 📄 **Text Summarization:** Creating short summaries

Text Preprocessing Techniques

- 👉 **Cleaning and Noise Removal:** Removes irrelevant and noisy parts of the text.
- ✕ **Normalization:** Converts text into a consistent and analyzable form.
- ✓ **Improves Machine Learning:** Helps algorithms perform better on processed text.

Typical Steps in Text Preprocessing

Typical Steps:

- **Tokenization:** Split text into words or tokens (e.g., “text”, “ming”, “is”, “the”, “best”, “!”)
- **Stemming:** (“lungs” → “lung”) or **Lemmatization:** (“were” → “is”)
- **Lowercasing:** (“Disease” → “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:** (“l42” → “l”)
- **Spell Correction:** (“hart” → “heart”)

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

Definitions: N-grams and POS Tagging

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

Example Sentence: *"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.

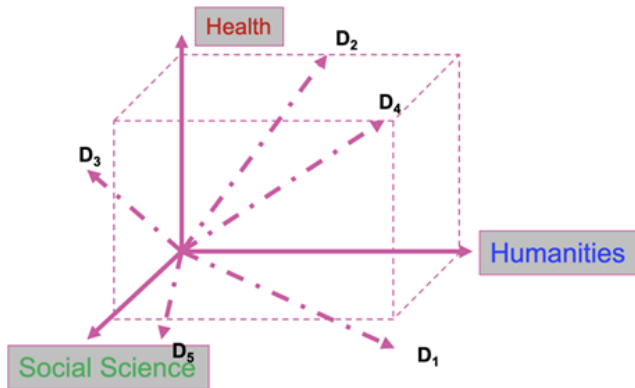
Feature Extraction and Vector Space Model

Vector Space Model

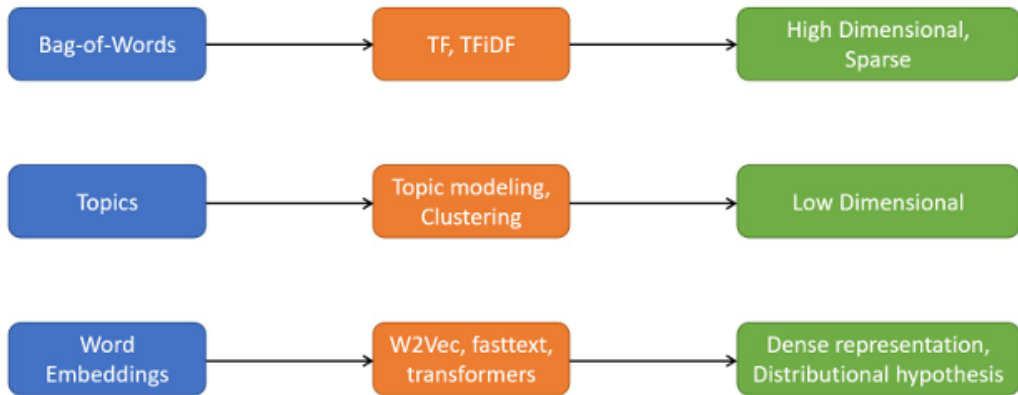
- Basic Idea:
 - Text is unstructured data that must be converted into a structured format for analysis.
 - A document can be represented in a way that computers can process.
- Representation Methods:
 - String representation: Lacks semantic meaning.
 - Sentence list: Treats each sentence as a short document.
 - Vector representation: Documents are represented as numeric vectors capturing word importance for analysis (this is a form of Feature Extraction that transforms raw text into measurable inputs for algorithms).

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.

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) - 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

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:

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

Summary

- **Unlock the Power of Text:** Transform unstructured data into actionable insights.
- **Everywhere You Look:** Essential for social media, customer analysis, healthcare, and beyond.
- **Cutting-Edge Tools:** Combines NLP and machine learning to reveal hidden patterns.

Practical 1