香港浸會大學
HONG KONG BAPTIST UNIVERSITY

Dive into Text Analytics with Python

-An Introductory Hands-on Workshop

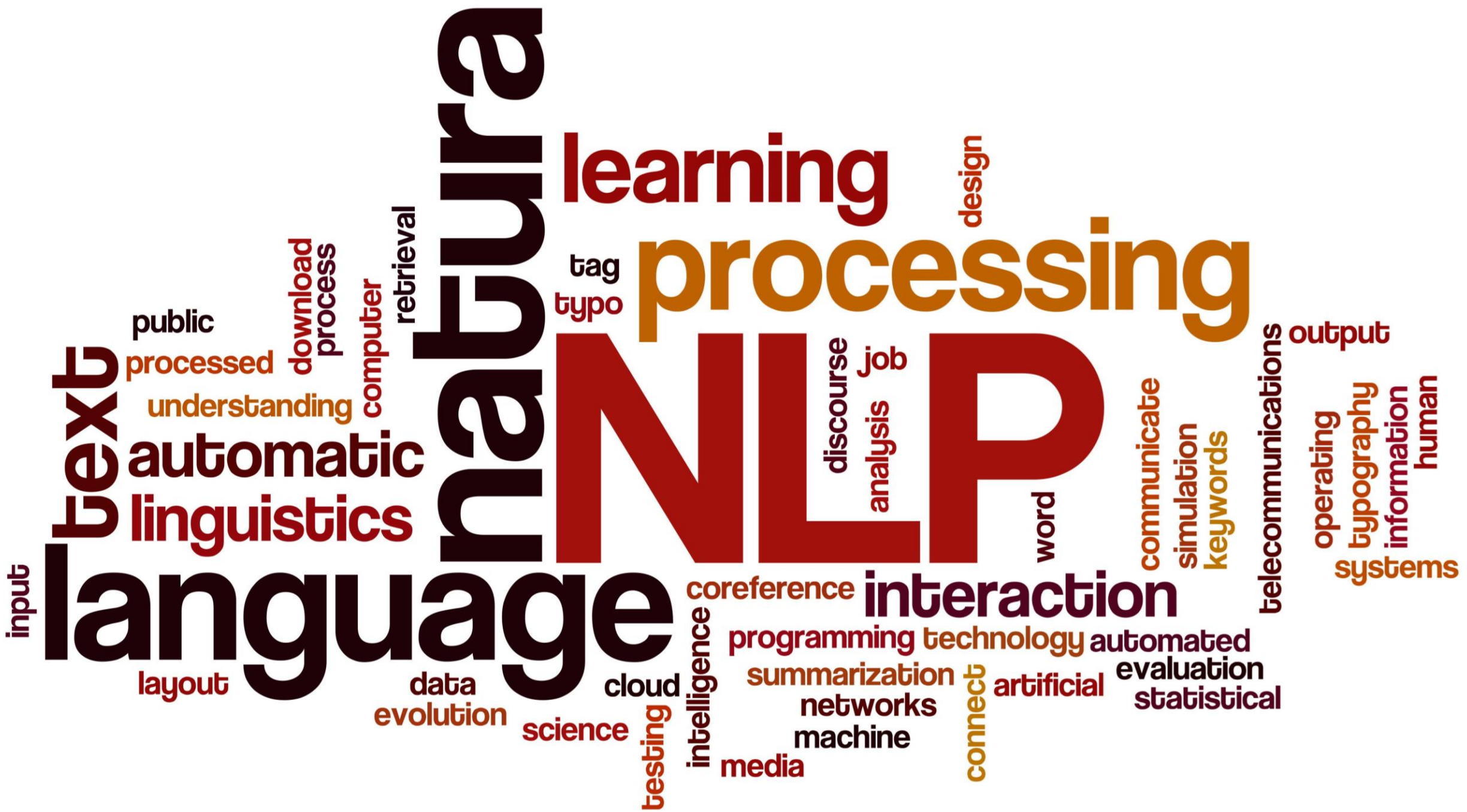
Dr. Xiaoyi Fu

Intended Learning Outcomes

- Understand basic concepts of text analytics
- Preprocessing techniques
- Apply methods to real-world applications

What is Text Analytics?

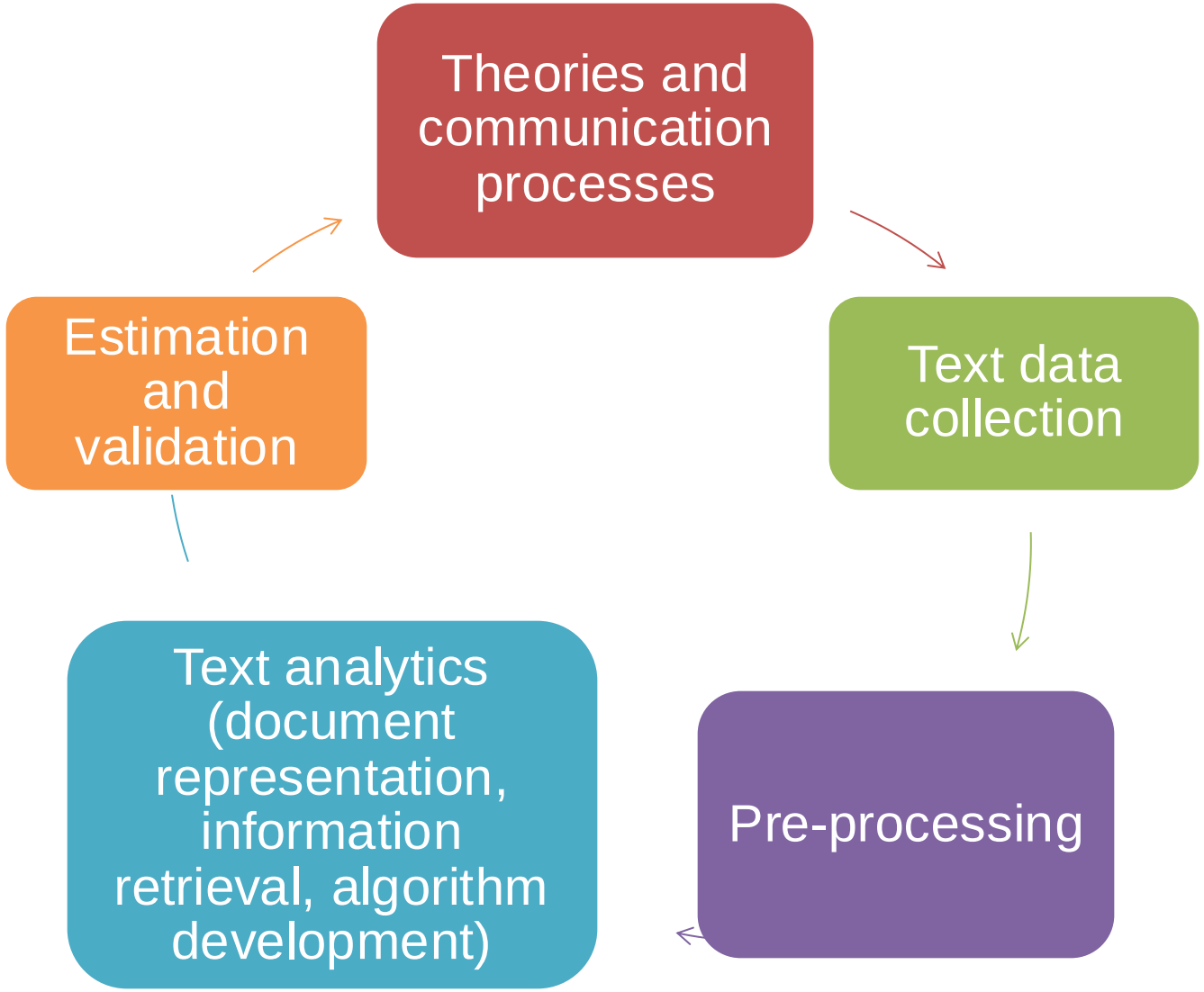
- What is text analytics?
  - An automated (or semi-automated) analysis (processing, analysis, visualization) of large-scale of text data
  - An area of data mining and machine learning
- Natural language processing (NLP)
  - Using machine learning to understand and analyze human language
  - Machine translation
  - Speech recognition
  - Sentiment analysis
  - Text generation (robot writer, automated journalism)


natural
learning
processing
NLP
language
text
automatic
linguistics
interaction
input
layout
data
evolution
science
cloud
testing
intelligence
programming
technology
automated
evaluation
statistical
summarization
networks
machine
media
artificial
connect
coreference
public
processed
understanding
download
process
computer
retrieval
tag
typo
design
discourse
analysis
job
word
communicate
simulation
keywords
telecommunications
output
operating
typography
information
human
systems

“Why”

- The advantages of digital data (“big data”) also hold (large-scale, always-on, unobtrusive, computational)
- It helps to retrieve the hidden information in (a large amount of) text data, and
- It reveals communication practices, perceptions, behaviors, and cultural values

The text analytics pipeline



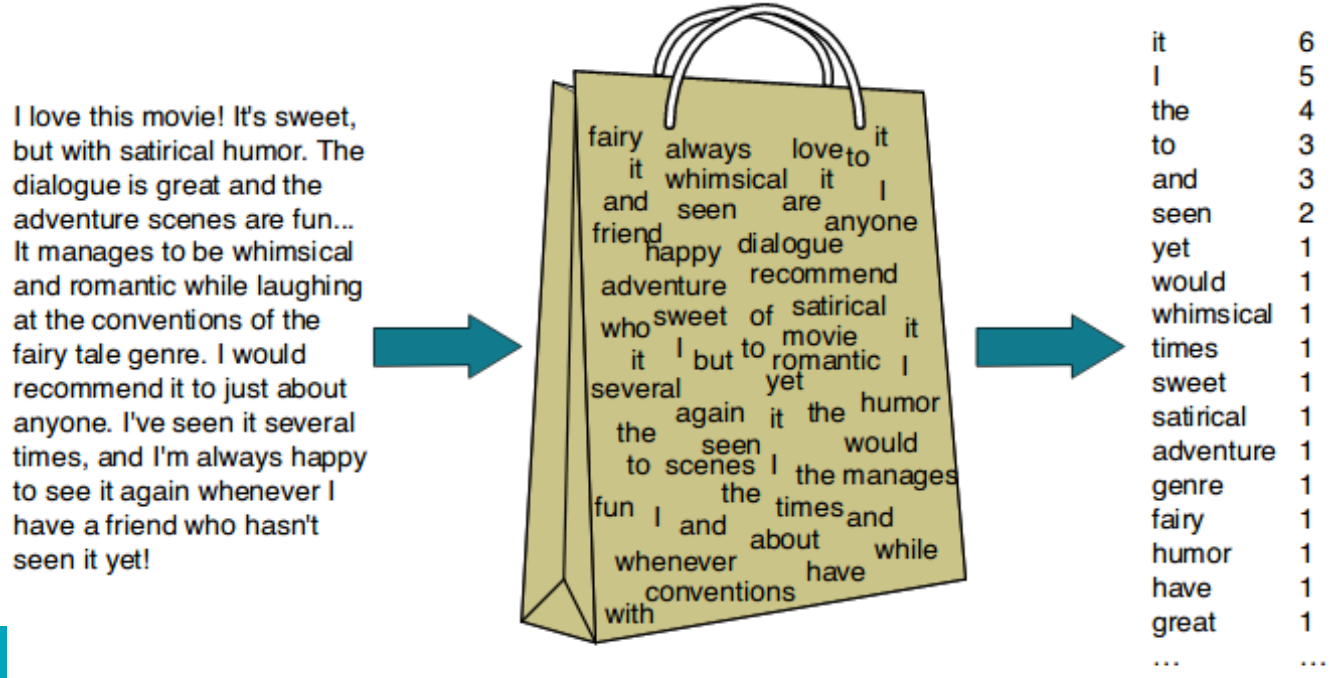
Working with Text Data

Analyzing text: Document representation

- *“Representation” – “turning texts into numbers”*
- There are several document representation approaches.
- The bag-of-words model is a common one.
- The bag-of-words model is used to form a vector representing a document using the frequency count of each term in the document.
- This method of document representation is called as a Vector Space Model (VSM).

The bag-of-words model

- Each document is encoded as the **list** of words (“terms”) it contains.
- “Case” = document
- “Variable” = the vocabulary of a document set
- “Value” = the number of times the term appears in the document



“DTM”

- Document 1 (D1): The boy hits the dog.
- Document 2 (D2): The boy hits the dog and the cat.
- Document 3 (D3): The boy likes the cat cat.

	The	boy	hits	dog	and	cat	likes
D1	2	1	1	1	0	0	0
D2	3	1	1	1	1	1	0
D3	2	1	0	0	0	2	1

- This “data frame” = document-terms matrix (DTM)
- Each row = document vector
- Individual cells = “term frequency” (TF): the number of times a term occurs in a document

The word-counting method: TF-IDF

- There are some “common” words (high frequencies)
- We want to identify the words that can distinguish the document from others.
  - Stopwords: “the” “a” “and”…
  - Common words: if all the documents contain the word “dog” then we cannot tell the differences among these documents based on the word “dog.”

The word-counting method: TF-IDF

- Solution: “de-weight” the “common” words
- Common = appearing in many documents
- DF = “document frequency” = fraction of documents that containing the term
- IDF = “Inverse document frequency” = invert DF and logged
- TF-IDF = term frequency–inverse document frequency
- TF-IDF is a numerical statistic to reflect how important a word is to a document, among N documents

Preprocessing

Tokenization

- Tokenization is the process of splitting the text up into individual units (tokens), such as words or characters.
- Two ways of tokenization: word or character

Word Tokenization

- Pros
  - Lowercase Conversion:
    - Ensures consistent tokenization (e.g., "The" and "the" are treated the same).
    - Exception: Proper nouns (e.g., names, places) may benefit from remaining capitalized.
  - Handling Sparse Words:
    - Replace rare words with an unknown token (e.g., <UNK>).
    - Reduces vocabulary size and model complexity.
  - Stemming:
    - Reduces words to their root form (e.g., "browse," "browsing" → "brows").
    - Helps group related words together.
  - Punctuation:
    - Either tokenize punctuation or remove it entirely.

Word Tokenization

- Cons
- Limited Vocabulary:
  - Model cannot predict words outside the training vocabulary.
  - Requires a large vocabulary to capture all possible words.

Character Tokenization

- Pros
- Out-of-Vocabulary Words:
  - Model can generate new words not seen in training.
  - Useful for creative text generation or handling rare words.
- Smaller Vocabulary:
  - Vocabulary size is much smaller (e.g., ~26 letters + punctuation).
  - Faster training due to fewer weights in the output layer.
- Case Sensitivity:
  - Option to treat uppercase and lowercase letters as separate tokens or convert to lowercase.

Character Tokenization

- Cons
- Longer Sequences:
  - Text is split into many more tokens, increasing sequence length.
  - May require more computational resources.
- Less Semantic Meaning:
  - Characters alone lack the semantic meaning of words.
  - Model must learn to combine characters into meaningful words.

Choosing Between Word and Character Tokenization

- When to Use Word Tokenization
  - Tasks requiring semantic understanding (e.g., sentiment analysis, machine translation).
  - When the training vocabulary is well-defined and large enough to cover most words.
- When to Use Character Tokenization
  - Tasks requiring flexibility (e.g., creative text generation, handling rare words).
  - When the training vocabulary is small or unknown words are expected.

Stop Word Removal

- Eliminate common, low-meaning words (e.g., "the," "and," "it") to reduce noise and focus on sentiment-rich terms.
- Why It Matters:
  - Boosts analysis accuracy by retaining meaningful words.
  - NLTK offers prebuilt stop word lists for multiple languages.
- Example:

  Before: "The movie was not good."
  After: "movie not good." → Sentiment (negative) becomes clearer.

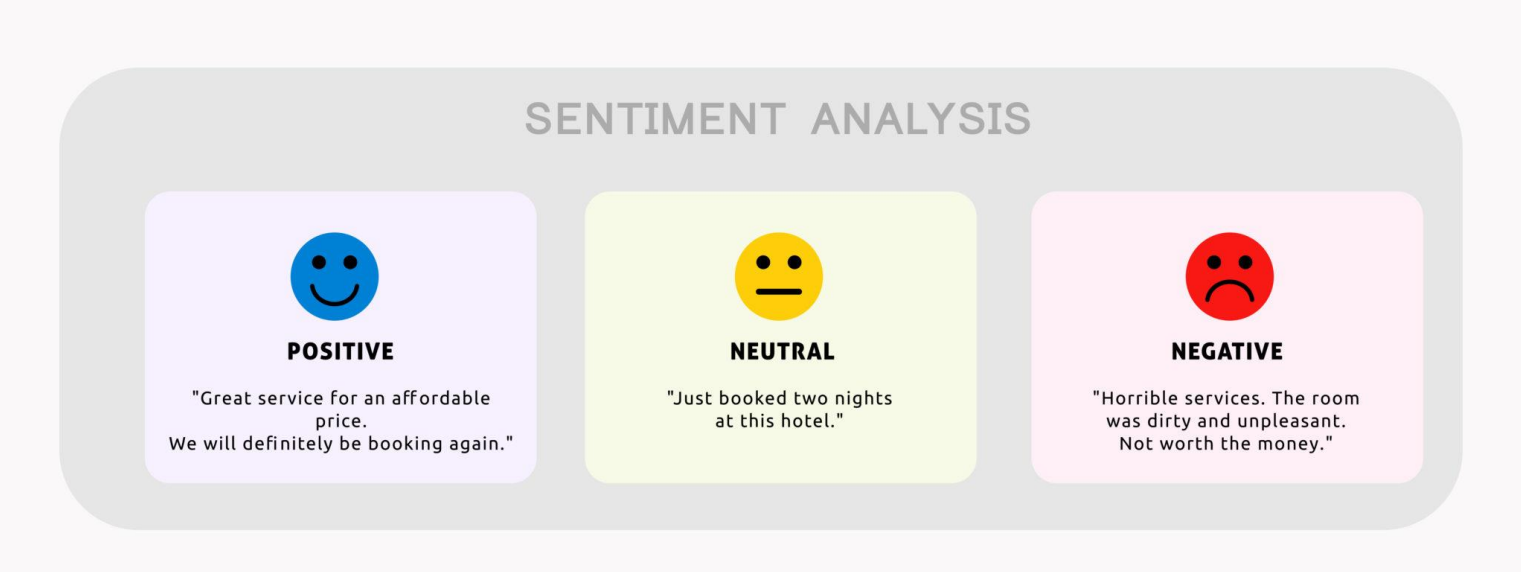
Stemming and Lemmatization

- Stemming:
  - Method: Chops off suffixes (e.g., "running" → "run").
  - Example: "jumping" → "jump" (may produce non-words like "happi" from "happy").
- Lemmatization:
  - The goal is to reduce a word to its root form, also called a lemma.
  - Example: Running → Run. Better → Good. Mice → Mouse
- Key Difference:
  - Stemming: Faster but less accurate.
  - Lemmatization: Slower but precise.

Application

Sentiment analysis

- Automatically detecting emotional tone (Positive/Negative/Neutral) in unstructured text.
- Key Uses:
  - ✔ Brand monitoring
  - ✔ Customer feedback analysis
  - ✔ Market trend prediction



Topic modeling

- Topic modeling is used to discover the topics that occur in a document's body or a text corpus.
- Unsupervised learning
- Performing real-time analysis on unstructured textual data
- Learn from unstructured data at scale
- Build a consistent understanding of data, regardless of its format.
- What are topics
  - Topics are the latent descriptions of a corpus (large group) of text.

LDA

- Latent Dirichlet Allocation (LDA) is an approach used in topic modeling based on probabilistic vectors of words, which indicate their relevance to the text corpus.
- Refs: https://noduslabs.com/cases/tutorial-lda-text-mining-network-analysis/

Applications of topic modeling in digital research

- A convenient method for “communication researchers to employ topic modeling to describe themes in a massive amount of text data without having much prior knowledge”.

- Bian, J., Yoshigoe, K., Hicks, A., Yuan, J., He, Z., Xie, M., ... & Modave, F. (2016). Mining Twitter to assess the public perception of the “Internet of Things”. PloS one, 11(7), e0158450.
- Ho, J. C., & Zhang, X. (2020). Strategies for marketing really new products to the mass market: A text mining-based case study of virtual reality games. Journal of Open Innovation: Technology, Market, and Complexity, 6(1), 1, p. 7
- Lu, S. (2022). News technology innovation as a field: A structural topic modeling analysis of patent data in mainland China. Communication and Society, 59, 147–175.

Fig 7. Topics learned from Twitter regarding the "Internet of Things".

doi:10.1371/journal.pone.0158450.g007

- Smart technologies—Smart home systems are already on the market, and production and advancement of smart cars have led the automobile industry. Many highly dense cities are undertaking smart city projects. Based on the top words used in this topic, public opinions seem positive for smart technology, but there is a concern apparent by the term "hacking".
- Connected device—The scale and connectivity of the IoT is particularly expressed in this topic (e.g., "connected", "billion", and "people"). Nevertheless, security seems to be a concerns as expressed by "Symantec" (a technology company who provides security products and services), possibly because of the "open"-ness of "connected" "device(s)".
- Emerging security—As seen in this topic (and in previous topics already), "security" and "privacy" seems to be a great concern for the public. At the same time, the topic term "emerging security" also implies that some discussions were about positive opinions about the IoT to fix existing security concerns.

Bian, J., Yoshigoe, K., Hicks, A., Yuan, J., He, Z., Xie, M., ... & Modave, F. (2016). Mining Twitter to assess the public perception of the "Internet of Things". PloS one, 11(7), e0158450, page 10.

Table 2. Topics discovered in text presentations of VR and non-VR games.

Topic Categories	VR Games	Non-VR Games
VR	Topic 5: experience, virtual, reality, world, first, real, immersive, life, characters, explore Topic 9: game, vive, htc, content, reality, virtual, may, oculus, full, rift	
General	Topic 4: get, just, make, like, dont, youll, time, youre, one, even	Topic 7: game, just, one, like, get, make, time, dont, want, youll
Gameplay/ Game Mechanics	Topic 1: game, mode, play, players, player, score, friends, challenge, arcade, fun Topic 2: game, new, now, games, early, access, please, hands, coming, also Topic 10: time, game, use, level, move, controller, right, levels, movement, around	Topic 1: game, games, gameplay, music, new, unique, experience, graphics, original, soundtrack Topic 5: build, different, game, new, make, world, many, city, get, create Topic 8: game, players, play, mode, player, friends, multiplayer, new, online, team
Game Content	Topic 3: music, play, like, create, different, amp, virtual, enjoy, choose, using Topic 6: weapons, enemies, different, fight, enemy, battle, weapon, combat, use, action Topic 7: world, find, story, puzzles, escape, explore, adventure, magic, room, solve Topic 8: space, city, take, way, control, around, mission, new, fly, planet	Topic 2: enemies, weapons, different, action, fight, use, enemy, unique, special, weapon Topic 3: battle, new, combat, game, strategy, war, battles, different, system, take Topic 4: world, adventure, monsters, new, find, explore, evil, journey, save, characters Topic 6: content, story, may, find, mature, appropriate, work, life, game, adventure Topic 9: space, ship, planet, system, control, must, ships, explore, one, survival Topic 10: game, levels, level, puzzle, time, simple, puzzles, move, achievements, different

Ho, J. C., & Zhang, X. (2020). Strategies for marketing really new products to the mass market: A text mining-based case study of virtual reality games. Journal of Open Innovation: Technology, Market, and Complexity, 6(1), 1, p. 6

Reflections

- Pros: easy to use and good for initial exploration (or when reliable training data is not available)
- Cons: unsupervised, merely descriptive, cannot establish causal analysis

References

- https://www.datacamp.com/tutorial/text-analytics-beginners-nltk
- https://www.datacamp.com/tutorial/what-is-topic-modeling

Thank You