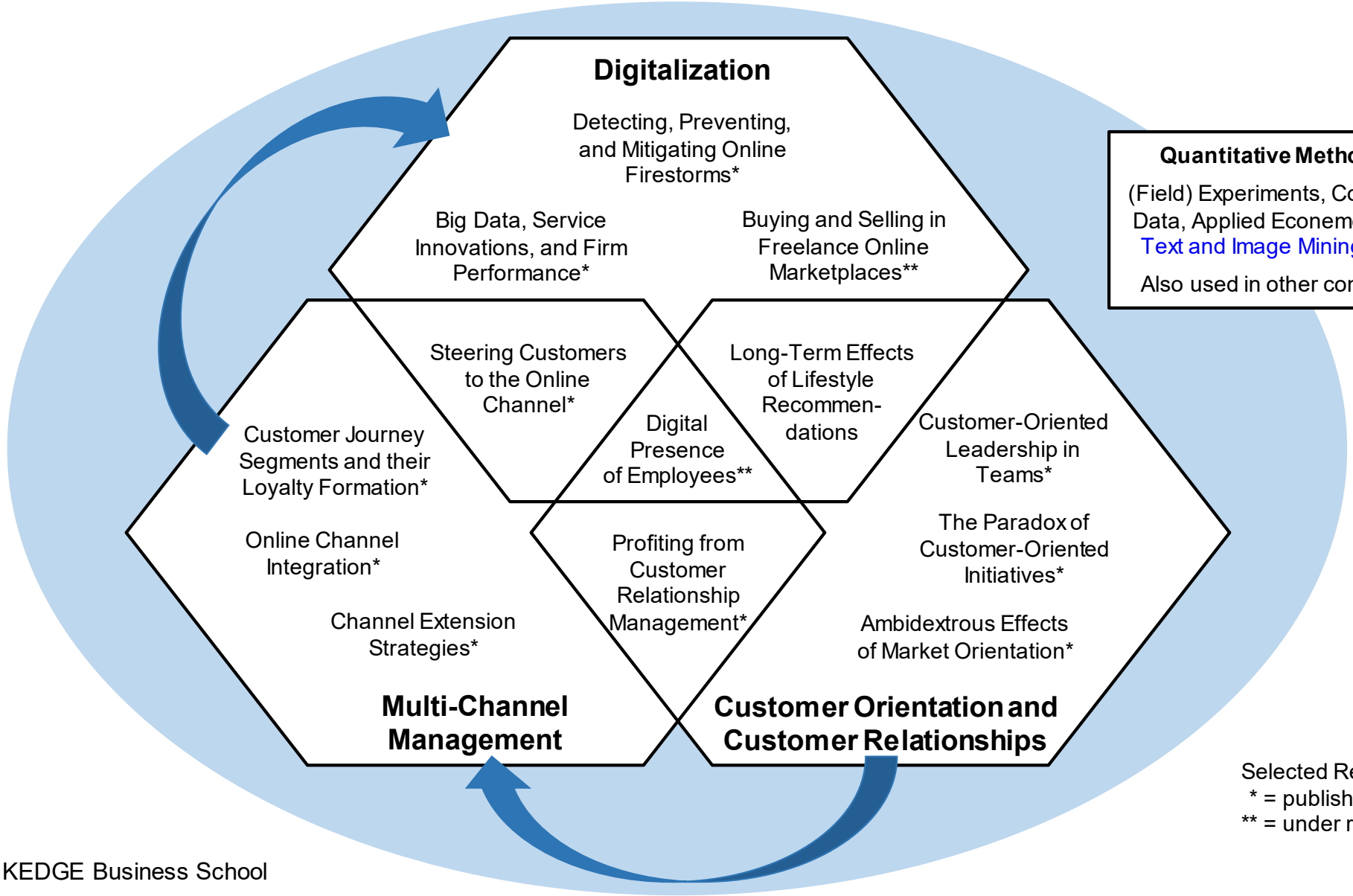
KEDGE Faculty Training

Automated Text Analysis – DAY 1

Prof. Dr. Dennis Herhausen, KEDGE Business School

Dennis Herhausen # My Research Journey



Quantitative Methods

(Field) Experiments, Complex Data, Applied Econemetrics, Text and Image Mining etc.

Also used in other contexts

Selected Research Projects
* = published
** = under review

Why Automated Text Analysis?

Analyzing a straw of hay:

Understanding the meaning of a sentence

- Humans are great
- But computer struggle

Qualitative Text Analysis

Organizing the haystack:

Describing, classifying, scaling texts

- Humans struggle
- But computers are great

What this course is about

Goals of this Training

- Understand the wide scope of linguistic theories to investigate text data
- Become knowledgeable about different text analysis methods
- Be aware of the consequences of analytic choices
- Employ different text analysis methods in R
- Build up the necessary skills to conduct and write up a text analysis study

Getting Started

What type of textual data have you worked with?

What data would you be interested in using/collecting?

Agenda: Automated Text Analysis

Day 1

1) What is Automated Text Analysis?
2) A Roadmap for Automated Text Analysis
   - ➢ *Getting Started with R and RStudio*
3) Data Preparation and Data Visualization

Day 2

4) Classification with Dictionaries
5) Classification with Supervised Machine Learning
6) Clustering and Topic Discovery

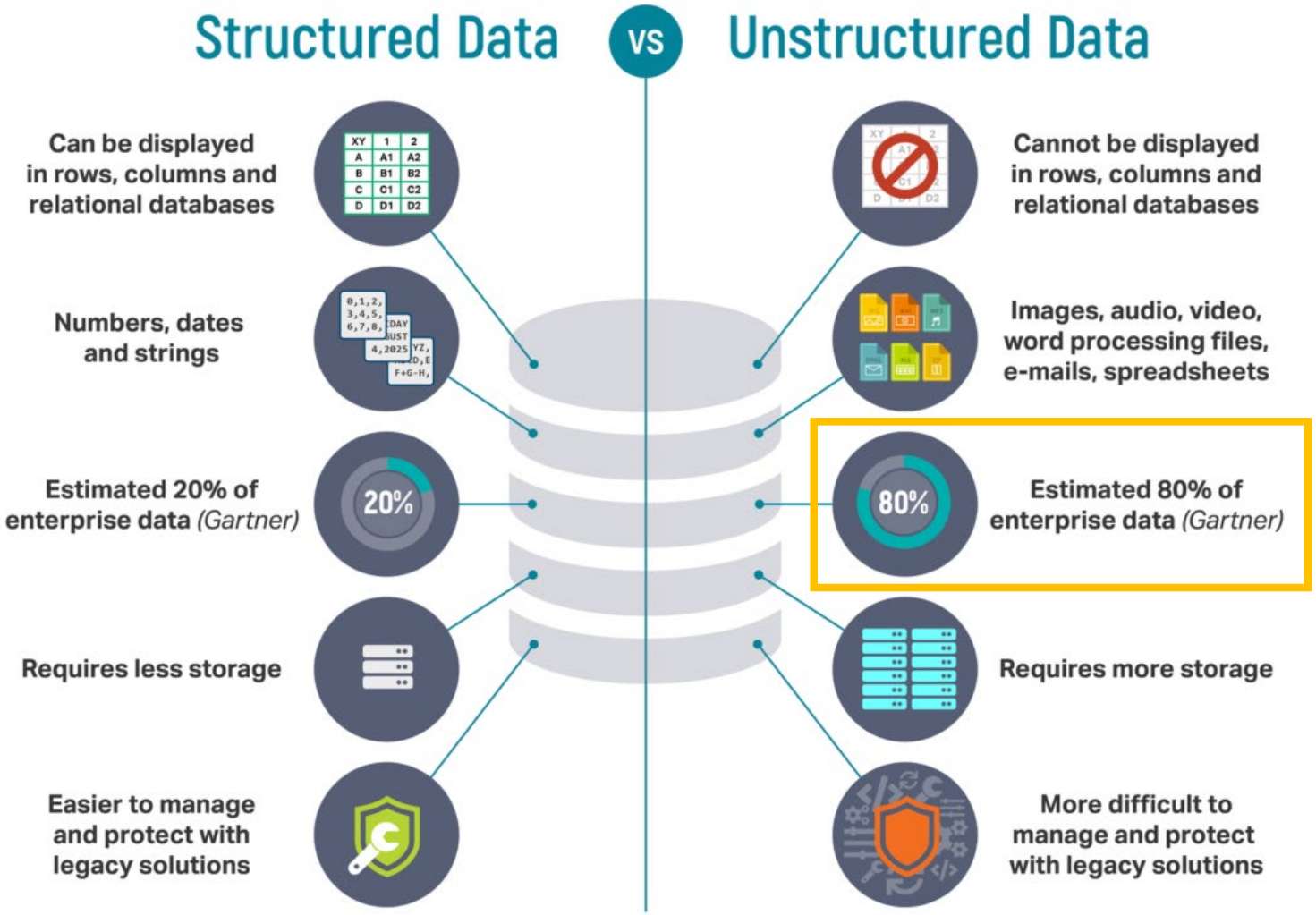
Disclaimer

This is a non-technical introduction that should enable you to understand and use the most common methods in text mining.

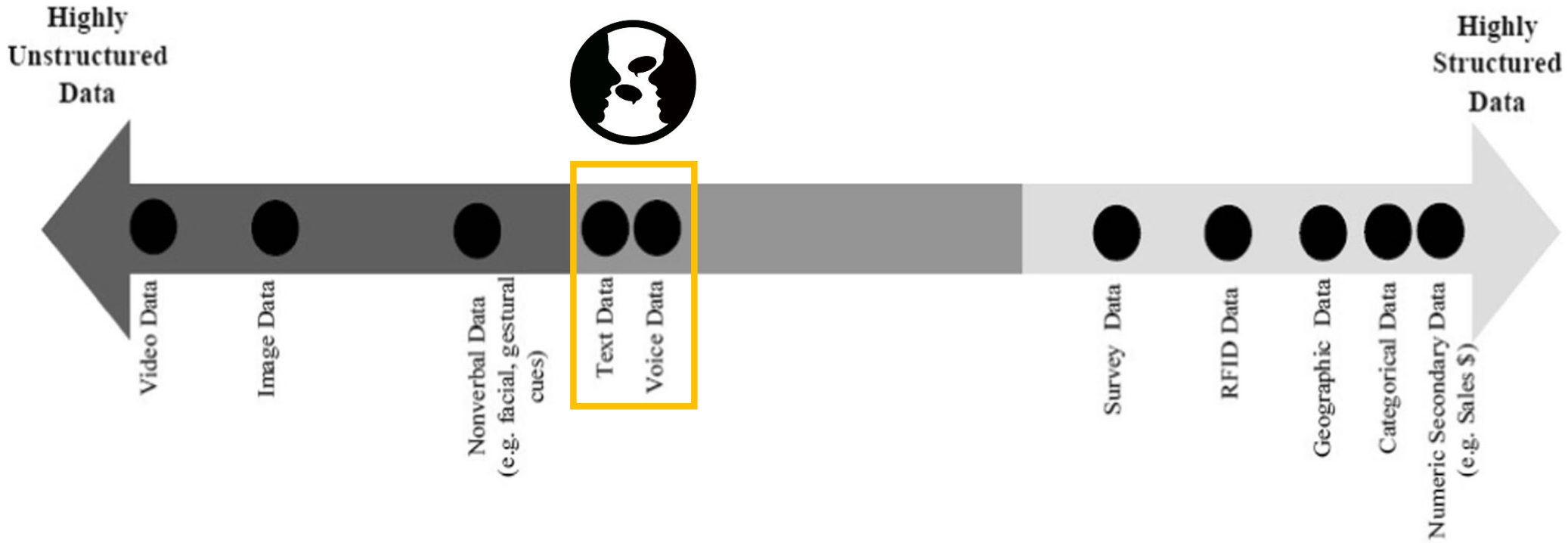
Thus, the training oversimplifies concepts, does not address all relevant aspects, and does not appropriately explain most technical aspects. For more details and technical aspects please consult the "Further Reading on Automated Text Analysis".

1) What is Automated Text Analysis?

“80% of Big Data is Unstructured Data”



What is Unstructured Data?



"A single data unit in which the information offers a relatively **concurrent representation** of its **multifaceted nature** without predefined **organization or numeric values**."

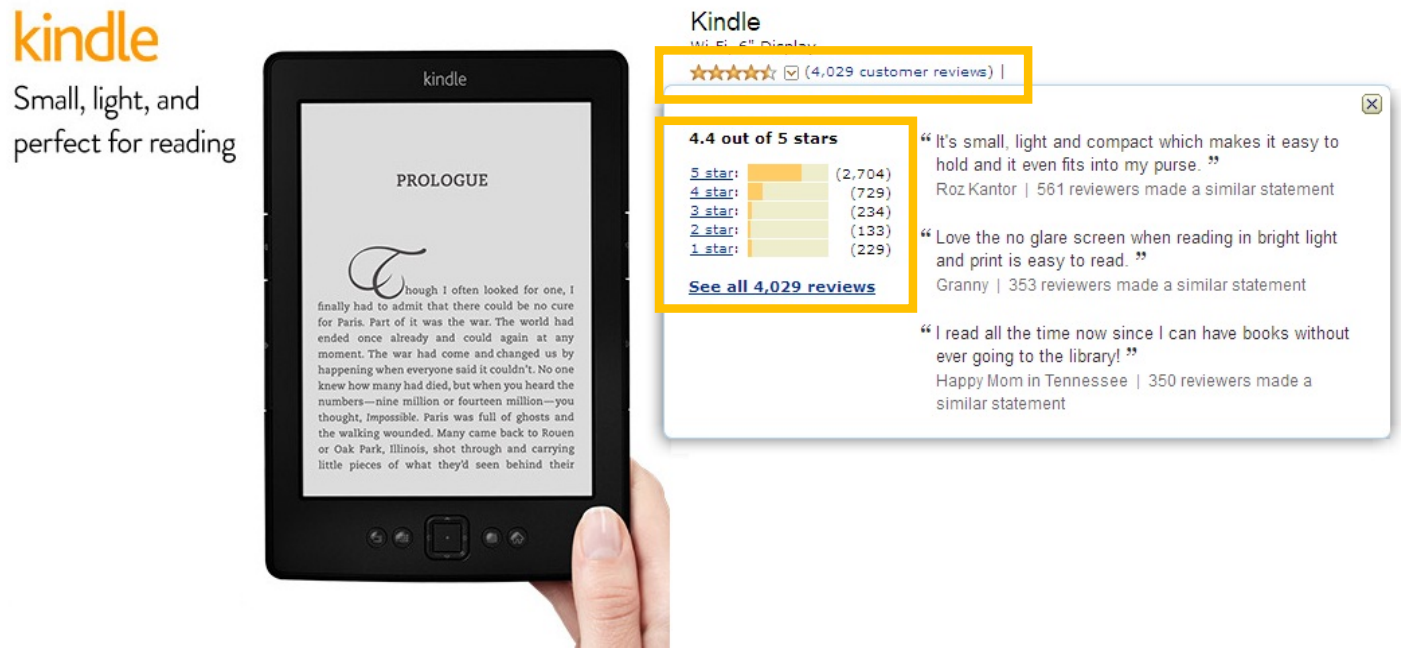
Balducci and Marinova 2018

Structured Data is “Ready” for Data Analysis

But Unstructured Data is Everywhere...

Even on More Structured Sites, Unstructured Data Matters

Numerical data (valence, volume, variance of “stars”) impacts sales



But: The **why** and **how** of the customer experience is not captured by stars...

Why is Text Data so Useful?

Among all data types, text is:

- the most **natural way of encoding human knowledge** (e.g., scientific knowledge, manuals, industry reports)
- by far the most **common type of information** encountered by people (produced and consumed)
- the **most expressive form of information** (used to describe other media such as video or images)

Compared to audio, image, and video data, text data is **relatively easy to analyze and to interpret**.

Text data is **already available** or can be relatively easily obtained.

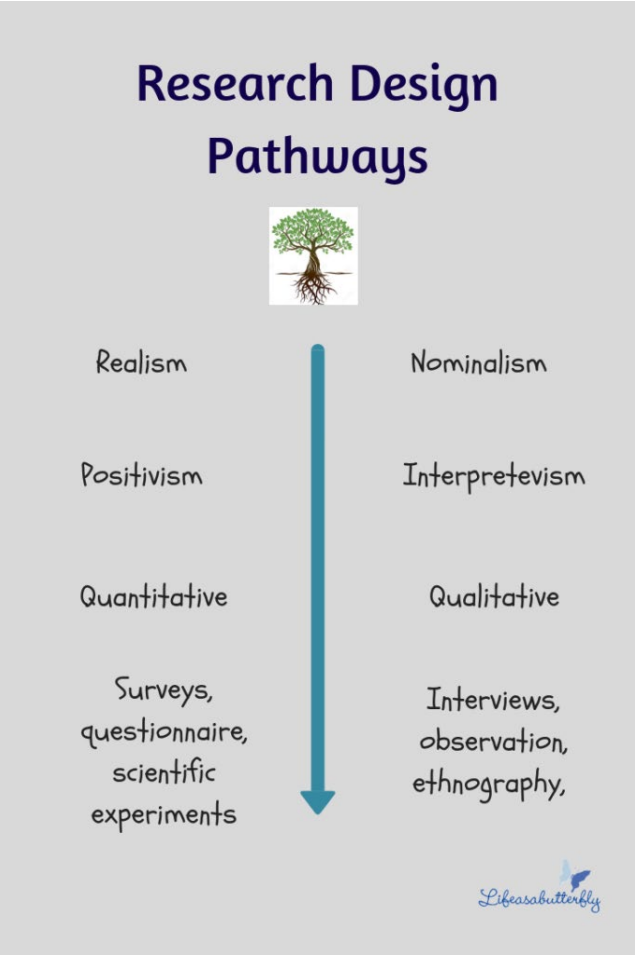
Zhai and Massung 2016

Text Data is Informative for Every Researcher

- For every **research approach** (e.g., quantitative, qualitative, modelling)
  - *Can be used for hypotheses testing, sense-making, or exploration*
  - *Can be combined with numerical or survey data (and any other data)*

- For every **research field** (marketing, strategy, management, IS, finance, …)
  - *Consumers write online reviews, answer open-ended survey questions, and call customer service representatives (the content of which can be transcribed)*
  - *Firms write ads, email frequently, publish annual reports, and issue press releases*
  - *Newspapers write articles, movies have scripts, and songs have lyrics...*

Text Mining across Different Types of Researchers



"The benefits of text analysis are best realized if we include both quantitative, positivists analysis of content (Cause), and qualitative, interpretative analysis of discourses (Meaning)"

Behavioral Researchers: Text data to increase external validity helps to address the why question and provides real-world applications.

Quantitative Modelers: Text data provides a rich set of predictors to be combined with structured variables to explain and predict relevant outcome.

Strategy Researchers: Text data contains valid measures of firm's assets resulting from press releases, patents, employee reviews, etc.

Qualitative and CCT Researchers: Text data quantifies qualitative information to measure meanings, norms, and values of consumers or humans more generally and in a natural context.

The Underlying Theory: “To Speak is to Act”

You're fired!

This product sucks.

I hereby appoint you as chairman.

DT is a notorious liar.

The central premise of **Speech Act Theory** is that language construction (in speech or writing), through words, sentences and interactional exchanges, conveys a speaker’s underlying intention.

More colloquially, “**to say something is to do something**”.

Speech acts can be analyzed on three levels:

- **A locutionary act:** the actual utterance and its apparent meaning, comprising any and all of its verbal, social, and rhetorical meanings, all of which correspond to the verbal, syntactic and semantic aspects of any meaningful utterance
- **An illocutionary act:** the active result of the implied request or meaning presented by the locutionary act.

  For example, if the locutionary act in an interaction is the question “Is there any salt?” the implied illocutionary request is “Can someone pass the salt to me?”
- **A further perlocutionary act:** the actual effect of the locutionary and illocutionary acts, such as persuading, convincing, scaring, enlightening, inspiring, or otherwise getting someone to do or realize something, whether intended or not.

Austin 1962, Searle 1976

Fields Involved in Automated Text Analysis

Linguistics: The study of language and its different branches such as syntax, semantics, pragmatics, rhetoric's, speech acts, etc.

Natural Language Processing: Concerned with developing computational techniques to enable a computer (i.e., AI) to understand the meaning of natural language text.

Pattern Recognition: Branch of machine learning that focuses on the recognitions of patterns and regularities in the data, although its in some cases considered synonym of Machine Learning.

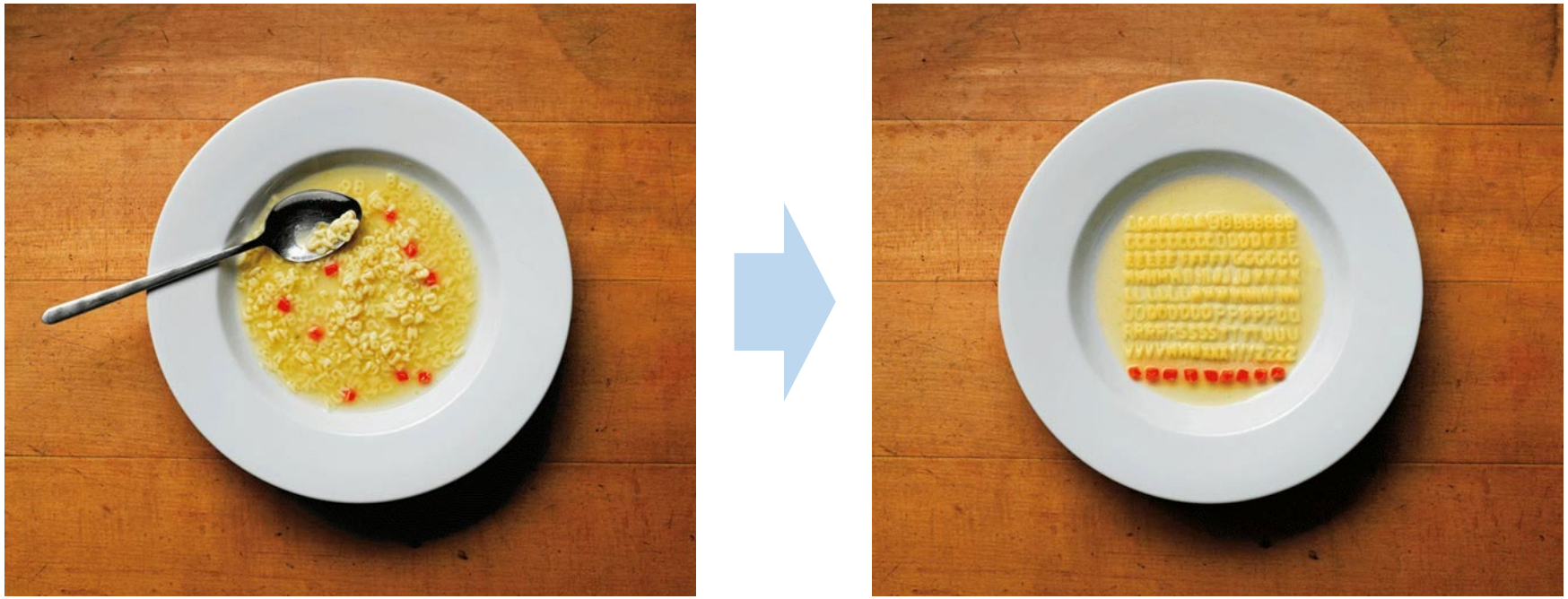
Content Analysis: Research method of gathering, analyzing, and categorizing the content associated with psychological constructs without preconceptions.

Statistics: Plays an important role in text mining algorithms. Measurements such as term frequency are based on the probability that a term is contained in a text. Also k-means clustering uses probability values to determine groups of words.

Villarroel Ordenes 2020

What is Automated Text Analysis?

In the Most Intuitive Way, Automated Text Analysis is...

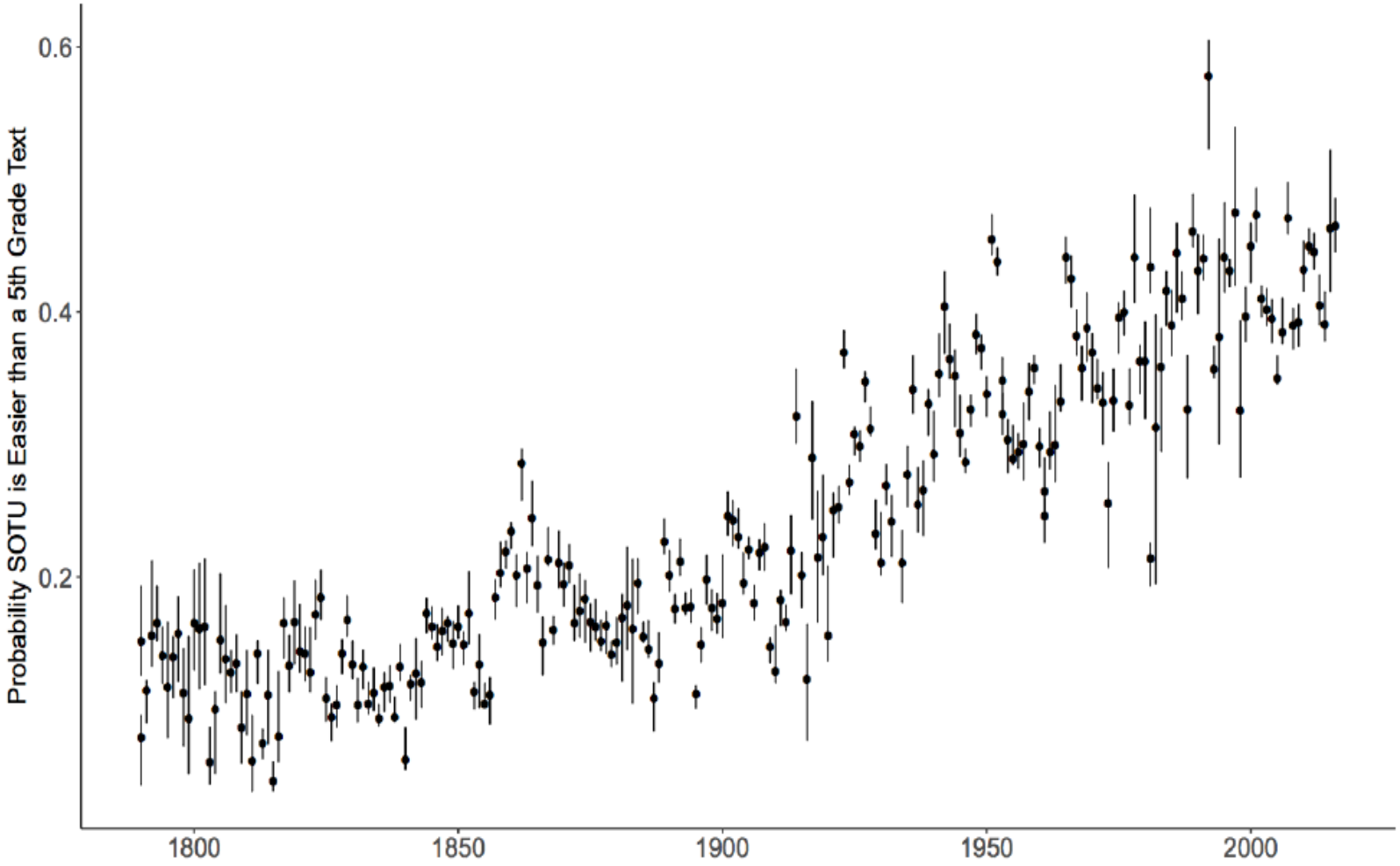


“Structuring the Unstructured”

Goal: Convert text into numbers to be analyzed with statistical and machine learning techniques

Villarroel Ordenes 2020

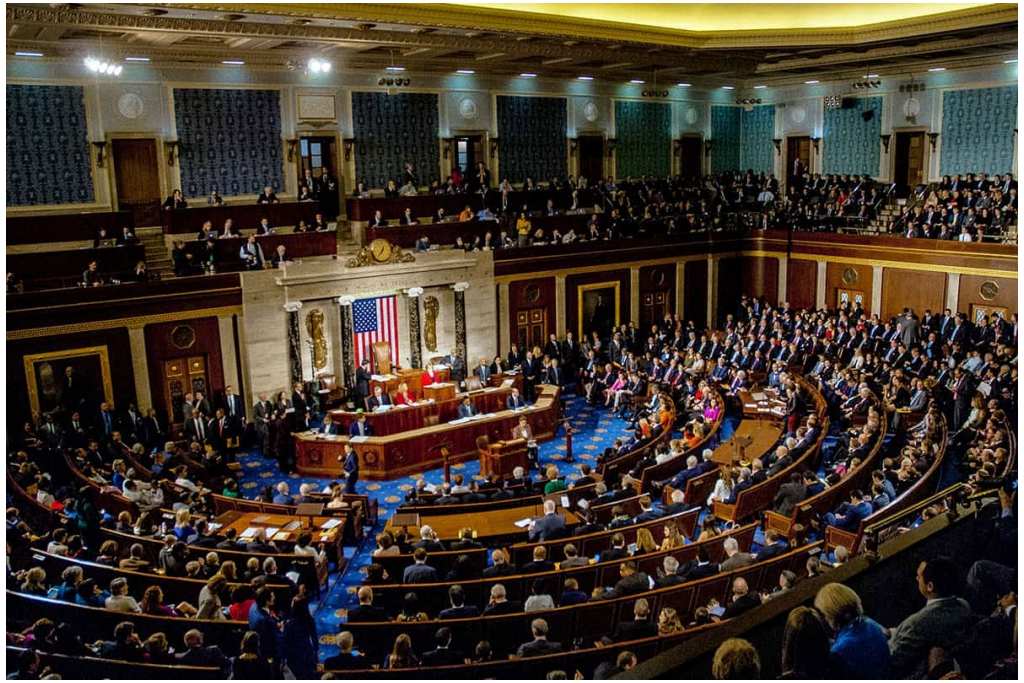
Example: Descriptive Text Analysis



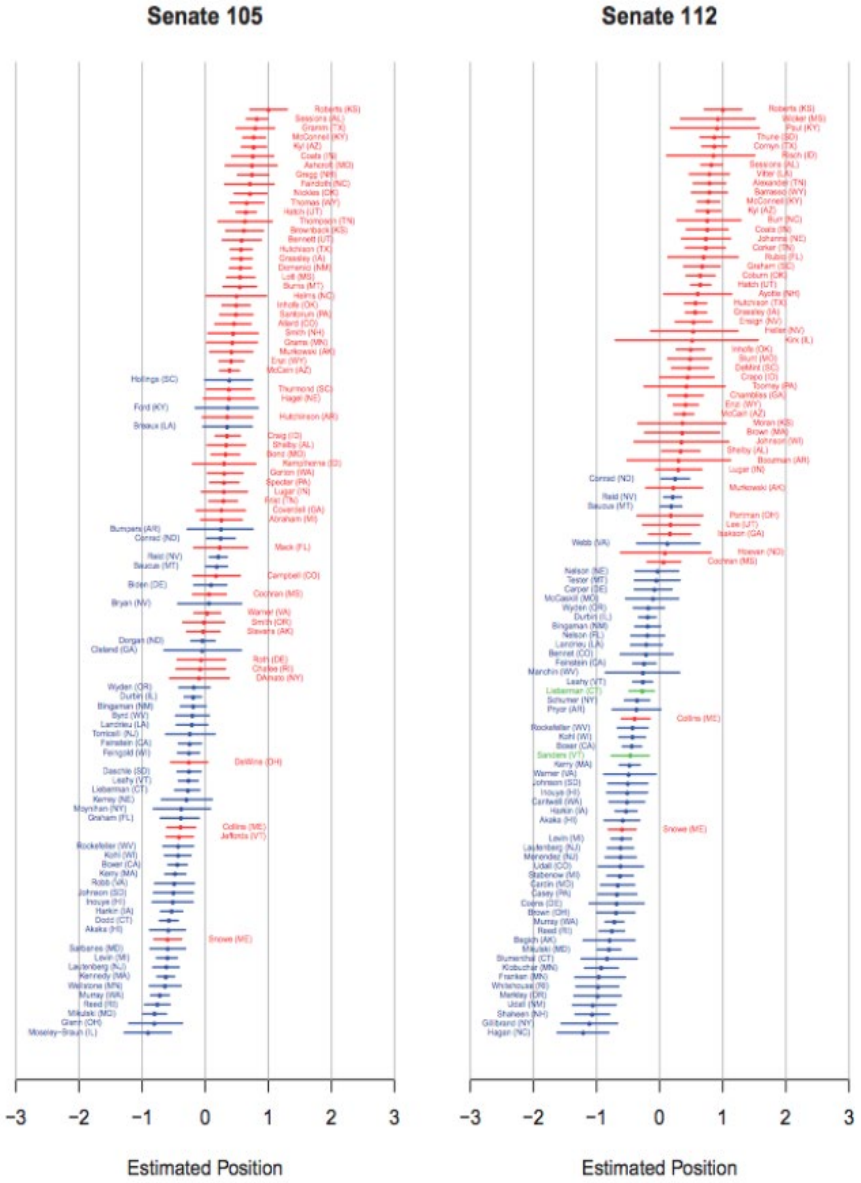
Dumbing Down?
Trends in the Complexity of Political Communication

Benoit, Munger, and Spirling 2017

Example: Classification into Known Categories

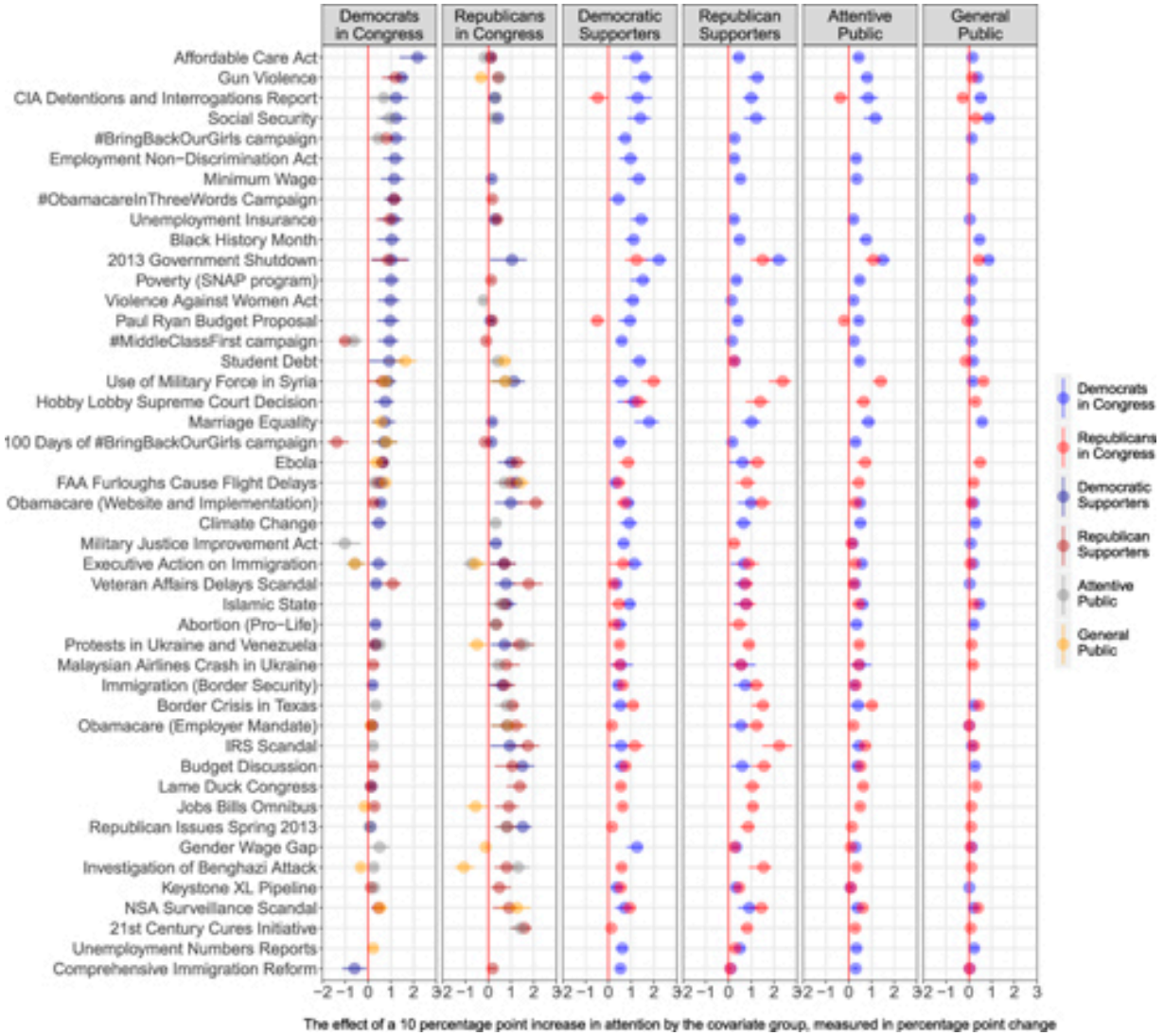


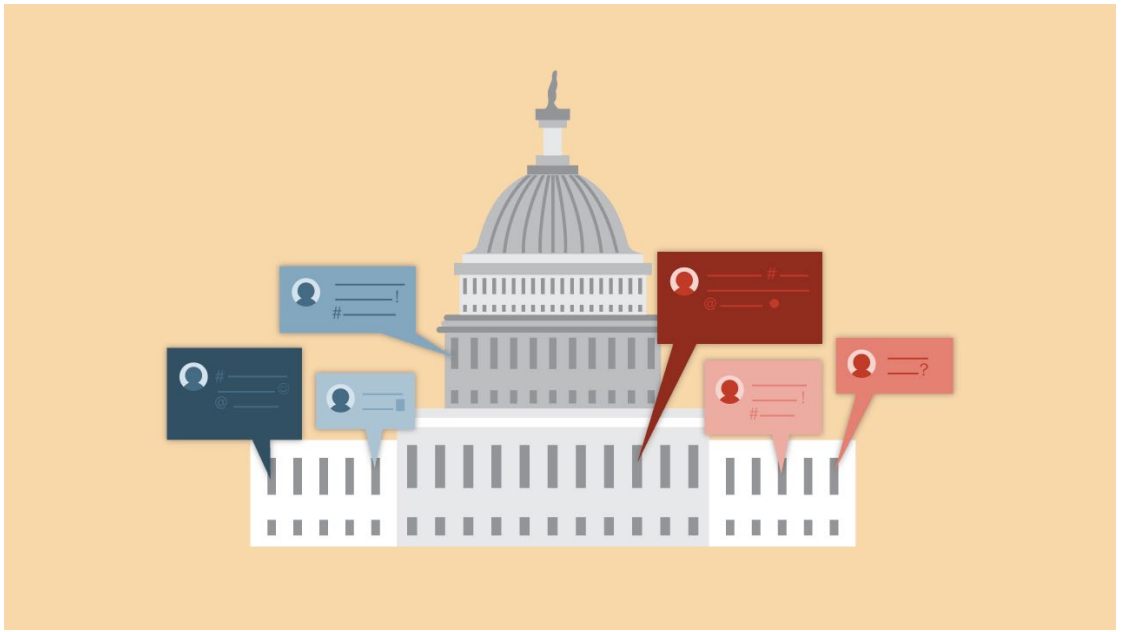
Measuring Political Positions from Legislative Speech



Lauderdale and Herzog 2016

Example: Classification into Unknown Categories





Leaders or Followers?
Measuring Political Responsiveness
in the U.S. Congress Using Social Media Data

Barberá, Bonneau, Egan, Jost, Nagler, and Tucker 2014

Automated Text Analysis requires Assumptions

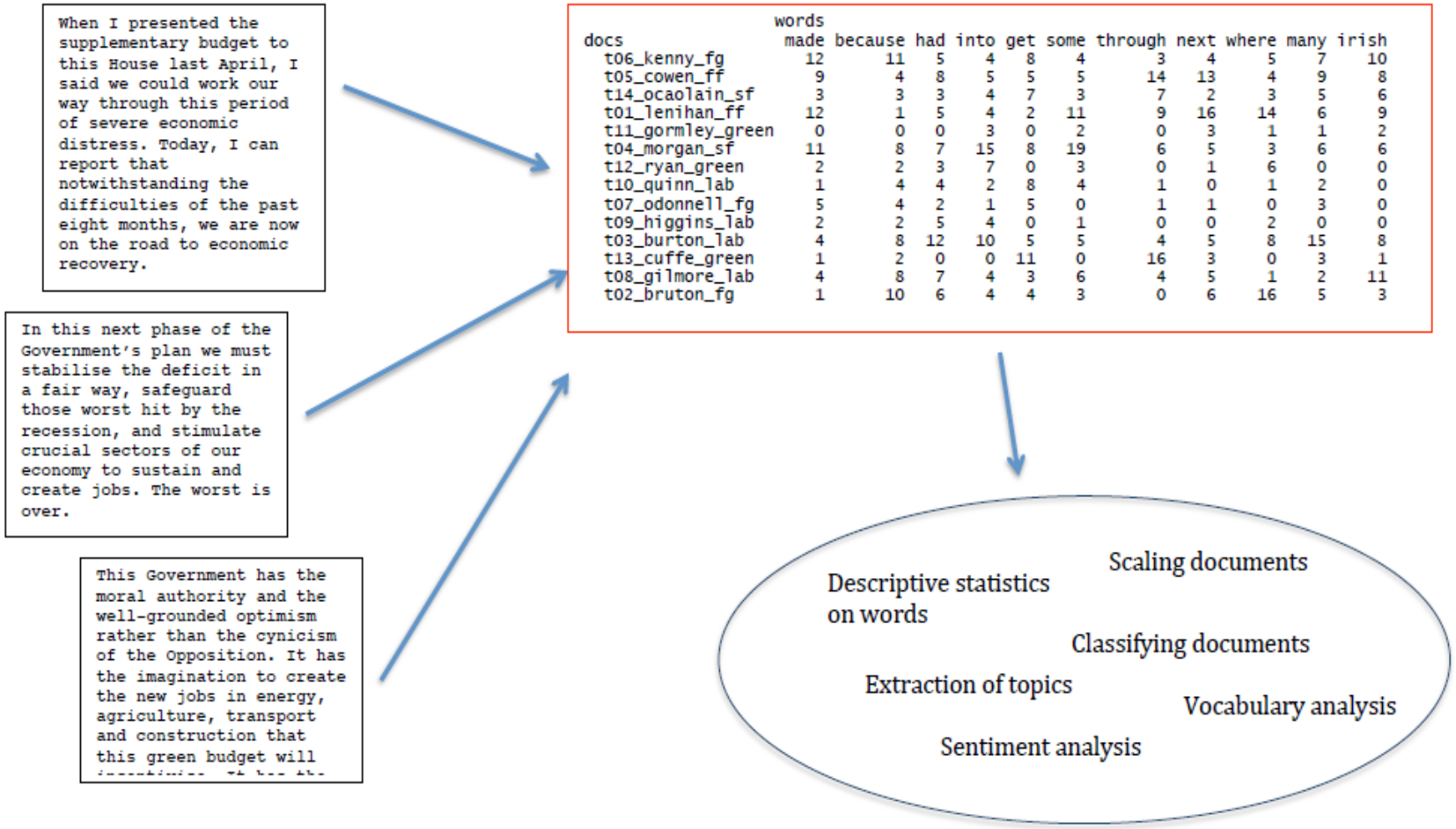
1. Texts represent an **observable implication** of some underlying **characteristic of interest**
   - An attribute of the author
   - A sentiment or emotion
   - Salience of a business issue
   - ...

2. Texts can be represented through **extracting their features**
   - Most common is the **bag of words** assumption
   - Many other possible definitions of “features” (e.g., word embeddings)

3. A **document-feature matrix** can be analyzed using **quantitative methods** to produce meaningful and valid estimates of the underlying characteristic of interest
   - E.g., classification, clustering, or and topic discovery

https://quanteda.io/

Texts → Document-Feature Matrix → Analysis



	words										
docs	made	because	had	into	get	some	through	next	where	many	irish
t06_kenny_fg	12	11	5	4	8	4	3	4	5	7	10
t05_cowen_ff	9	4	8	5	5	5	14	13	4	9	8
t14_ocaolain_sf	3	3	3	4	7	3	7	2	3	5	6
t01_lenihan_ff	12	1	5	4	2	11	9	16	14	6	9
t11_gormley_green	0	0	0	3	0	2	0	3	1	1	2
t04_morgan_sf	11	8	7	15	8	19	6	5	3	6	6
t12_ryan_green	2	2	3	7	0	3	0	1	6	0	0
t10_quinn_lab	1	4	4	2	8	4	1	0	1	2	0
t07_odonnell_fg	5	4	2	1	5	0	1	1	0	3	0
t09_higgins_lab	2	2	5	4	0	1	0	0	2	0	0
t03_burton_lab	4	8	12	10	5	5	4	5	8	15	8
t13_cuffe_green	1	2	0	0	11	0	16	3	0	3	1
t08_gilmore_lab	4	8	7	4	3	6	4	5	1	2	11
t02_bruton_fg	1	10	6	4	4	3	0	6	16	5	3

https://quanteda.io/

Simplified Overview of "Text-as-Data" Method

Acquire Documents

Existing Corpora,
Electronic Sources,
Digitized Text,
Undigitized Text,
(Books, Speech)

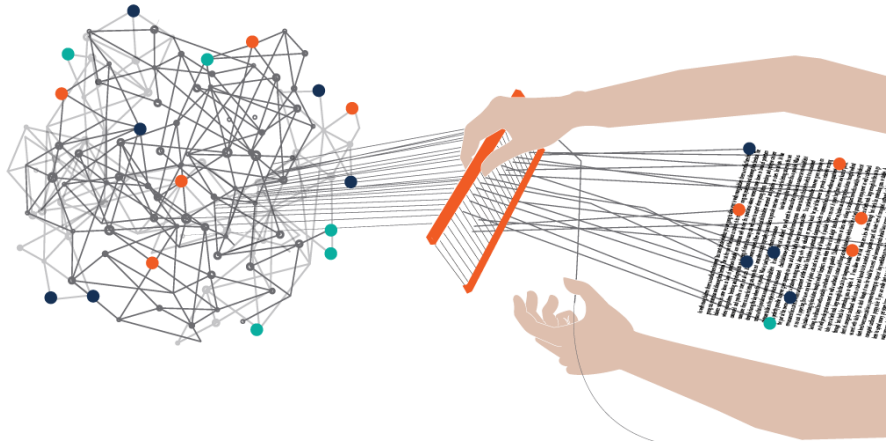
See "Further Reading on Web Scraping" and "Packages for Data Collection in R"

Preprocess Text

Data Preparation and
Data Visualization

Analyze Text

- Classification with Dictionaries
- Classification with Supervised Machine Learning
- Clustering and Topic Discovery

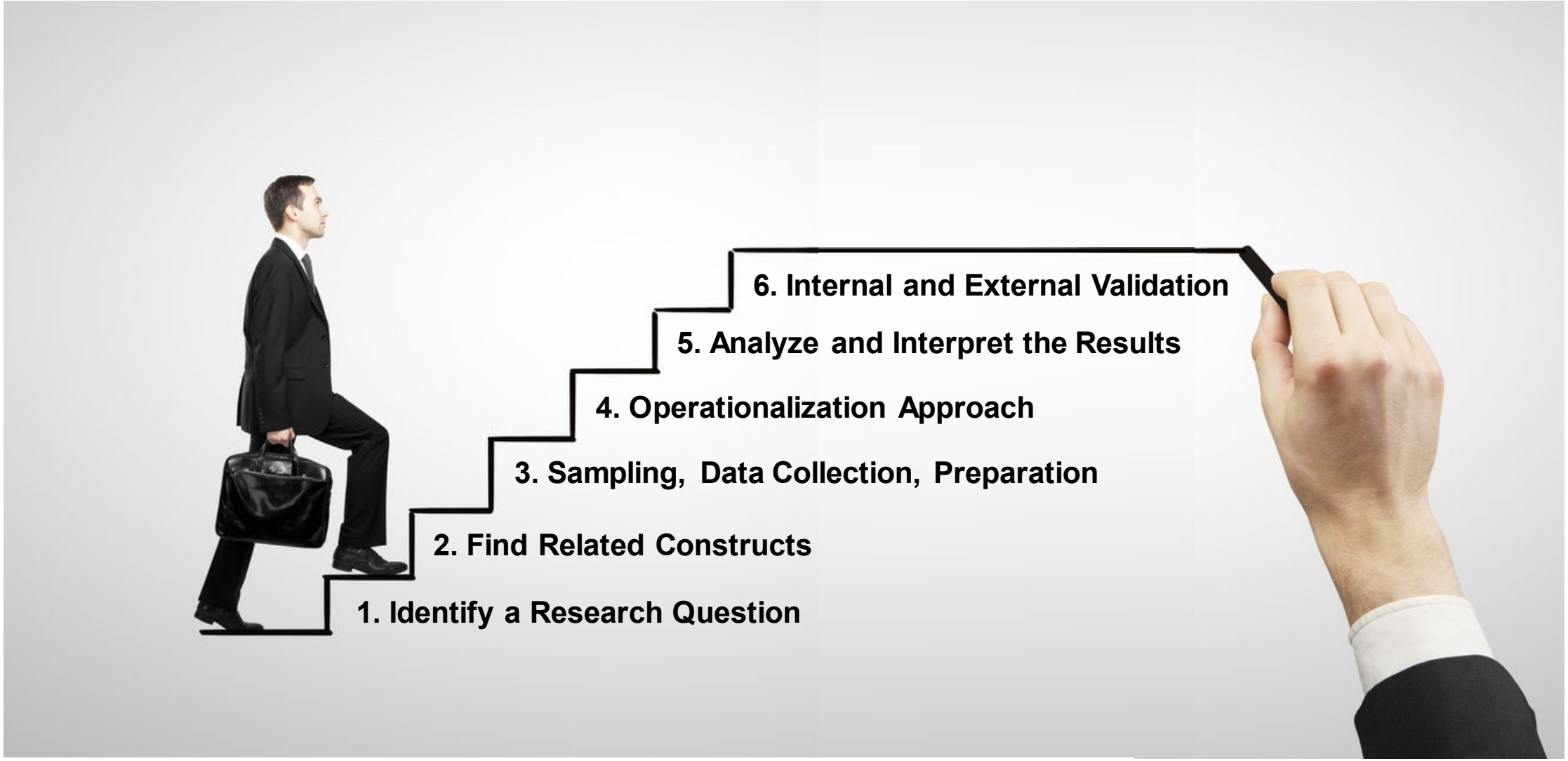


ct. Grimmer and Stewart 2013

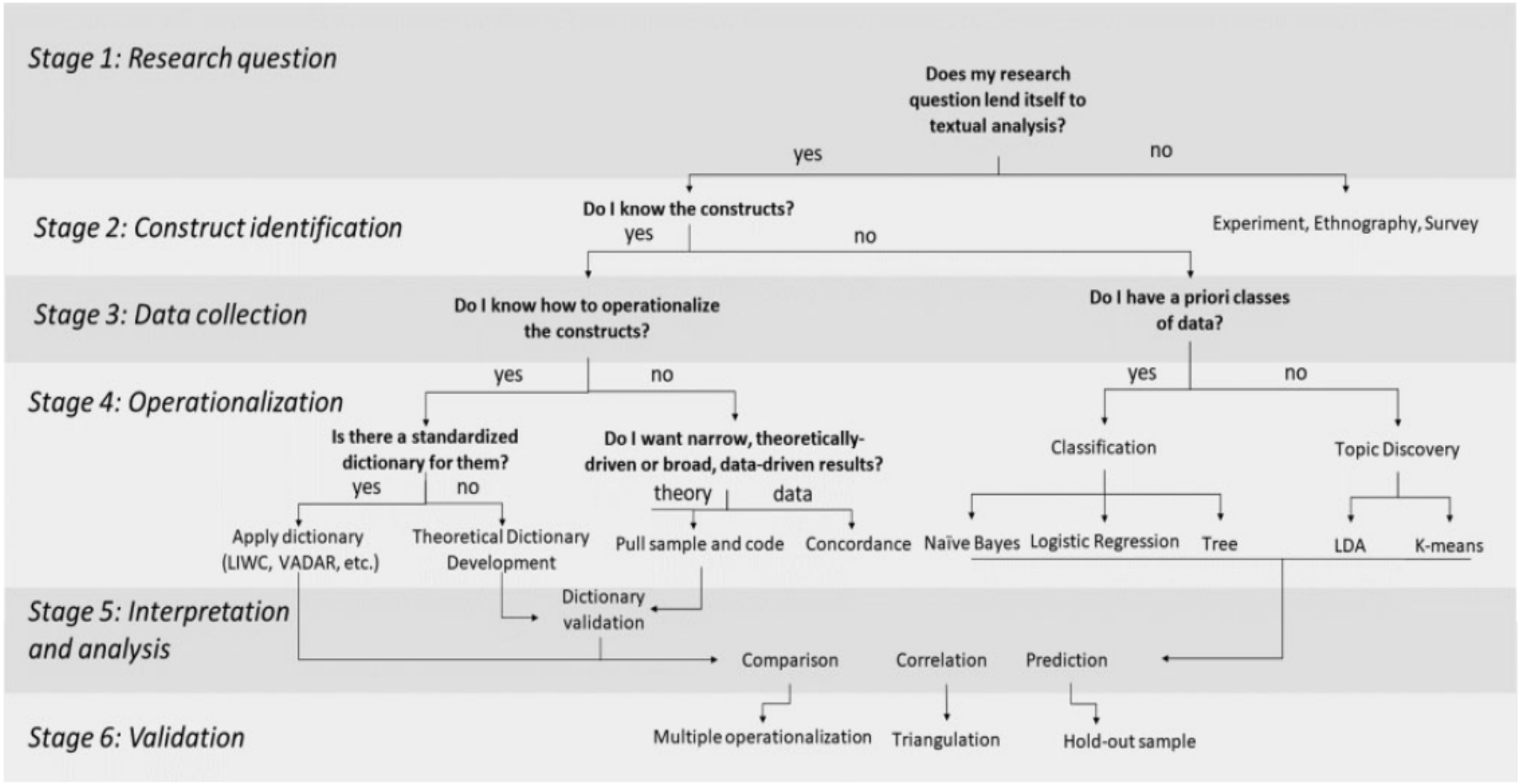
"Research teams must have the **interpretative skills** to understand the meaning of words, the **behavioral skills** to link them to underlying psychological processes, the **quantitative skills** to build the right statistical models, and the **strategy skills** to understand what these findings mean for firm actions and outcomes"

Berger et al. 2020

2) A Roadmap for Automated Text Analysis



Stages are based on Humphreys and Wang (2017)



1. Identify a Research Question

Description ("model free evidence")

- Application of statistical models for the purpose of **capturing the association between observations**

Prediction ("modeling")

- Application of statistical models or data mining algorithm for the purpose of **predicting new or future observations**
- Results in more modelling papers where the purpose is using advanced algorithms aiming at prediction. Cares less about the individual feature and more about set of predictive features.

Understanding ("theorizing")

- Application of statistical models for testing **causal hypotheses about theoretical constructs**
- Results in more behavioral papers where the purpose is using simpler text mining methods to operationalize a variable (or a limited set) and explain a relationship with a behavioral/attitudinal outcome.

The “Universe of Text”

1. Text as a Reflection of the Producer

Language as a “fingerprint” (*Pennebaker 2011*). Text can be used to predict writers deceitfulness (*Ludwig et al. 2016*), severity of complaining (*Herhausen et al. 2019*), or product quality and valence perceptions (*Tirunillai and Tellis 2014*).

2. Impact of Text on the Receiver

Text can change brand attitudes (*Humphreys and Latour 2013*), purchase behavior (*Ludwig et al. 2013*), cultural success (*Berger and Packard 2019; Eliashberg et al. 2014*), e-word of mouth transmission (*Herhausen et al. 2019*), advertising investments (*Wies et al. 2019*)

3. Contextual Influences on Text

The context always influences text, e.g., in the form of the platform (*Facebook vs. Twitter*), of the device (*phone vs. PC*) or in the context of the situation where the message is produced (*private customer service chat vs. public customer service tweet*)

- Text almost always “reflects” and “impacts”. Option 1 and 2 are very related
- In my perspective, 95% of research has used text as an IV but more exploratory studies might use text also as the DV

When is Automated Text Analysis Appropriate?

In general, it is good for analyzing text data in a context where humans may be limited or partial:

- Computers can sometimes **see patterns in language** that humans cannot detect, and they are impartial in the sense that they **measure textual data evenly and precisely** over time or in comparisons between groups without preconception.
- Further, by quantifying constructs in text, computers provide new ways of **aggregating and displaying information to uncover patterns** that may not be obvious at the granular level.

There are **at least four types of problems** where these advantages can be leveraged:

1) Automated text analysis can lead to discoveries of systematic relationships in text and hence amongst constructs that may be overlooked by researchers or consumers themselves (*relationships amongst three or more textual elements are simply hard for a human reader to see*)
2) Researchers can use computers to execute rules in order to measure changes in language over time, compare between groups, or aggregate large amounts of text (e.g., *"seeing" the text through conceptual maps or timelines networks*)
3) For some relationships observational data is the most natural way to study the phenomenon (e.g., *consumer-to-consumer interactions in online brand communities*)
4) Text analysis can be a companion to experimental research by adding ecological validity to lab results

Humphreys and Wang 2017

Not so Ideal Areas for Automated Text Analysis only…

Sometimes other (primarily) methods are more appropriate:

- When inferring **causation using a psychological mechanism**. E.g., precise control to compare groups, introduce manipulations, or rule out alternative hypothesis through random assignment protocols.
  - ➢ *Automated text analysis can be a companion, but not the only method (with experiment)*
- When you need a **behavioral dependent variable** then text analysis would not be appropriate to measure it. Not all constructs lend themselves to examination through text, and these constructs tend to be behavioral oriented.
  - ➢ *Automated text analysis can be a companion, but not the only method (with other DV)*
- To identify **finer shades of meaning** such as sarcasm, rhetoric, mindfulness, or other complex concepts. E.g., areas where deeper insights need to be extracted using discourse analysis or hermeneutic analysis.
  - ➢ *Automated text analysis can be a companion, but not the only method (with human content analysis)*

Automated text analysis is a distinct component of the research design that need to be **executed and then incorporated into the overall research design!**

Humphreys and Wang 2017

2. Find Related Constructs

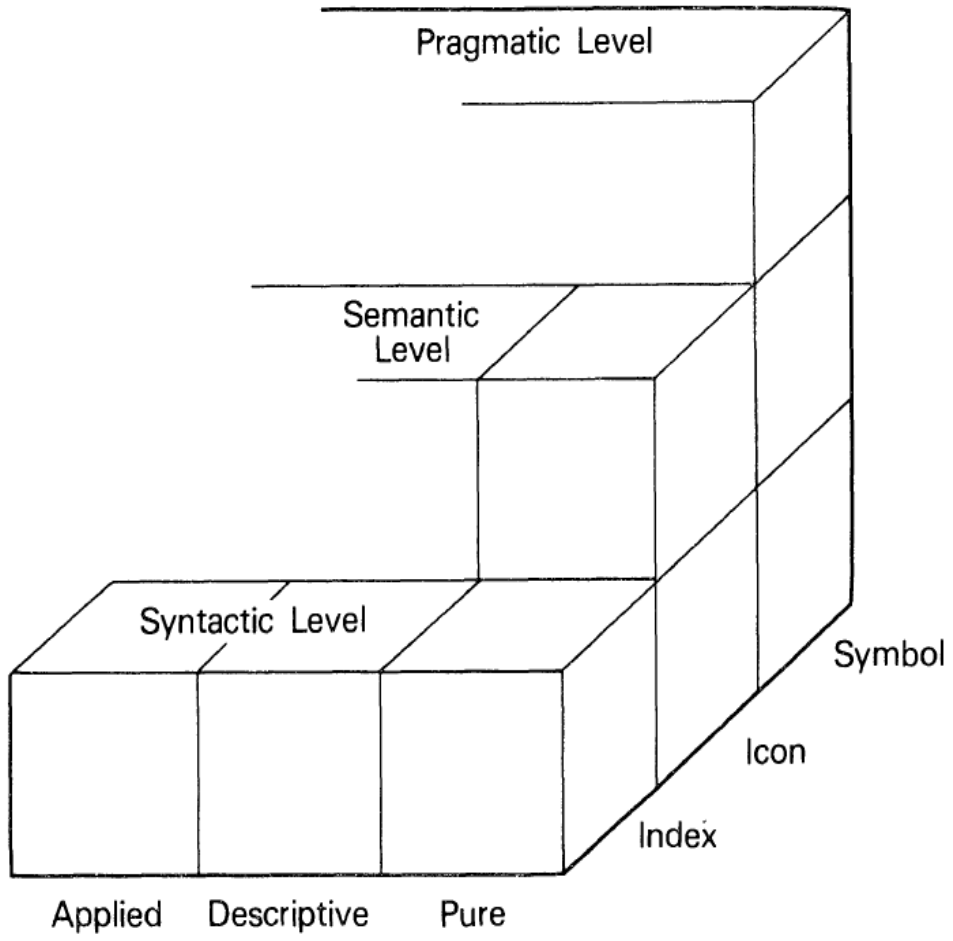
If Automated text analysis might be appropriate for the research question, the next step is to **identify the construct(s)**.

Doing so, however, entails recognizing that text is ultimately **based on language**.

As a sign system, language has three aspect that provide a unique window into humans thought, interaction, or culture:

- **Syntax** focuses on grammar, the order in which linguistic elements are presented
- **Semantics** concerns word meaning that is explicit in linguistic content → dictionary approach (e.g., LIWC)
- **Pragmatics** addresses the interaction between linguistic content and extra-linguistic factors like context or the relationship between speaker and hearer

By understanding and appreciating these linguistic underpinnings, researchers can develop **sounder operationalizations** of the constructs and more **insightful, novel hypotheses**.



Humphreys and Wang 2017, Mick 1986

Examples for Constructs in Marketing and Management

Four theoretical areas of consumer research that link with linguistics:

- **Attention – Semantics:** Study of emotions *("explicit vs. implicit" emotions)* and self-construal theory *("I vs. we")*
- **Processing – Syntax:** Study of language complexity *("but and without")* and narrativity (use of verbs)
- **Interpersonal Dynamics – Pragmatics:** Study of language in action *("speech acts")* and as a means of power
- **Group Characteristics – Semantics and Pragmatics:** Study of cultural compatibility in movie reviews and language style matching in product reviews

But not always a linguistic construct is necessary, many times the construct (latent or predefined) might come from:

- **A theory different from Linguistics:** The constructs comes from the psychology or strategy literature
- **The business field itself:** Extracting latent constructs and matching them with a framework

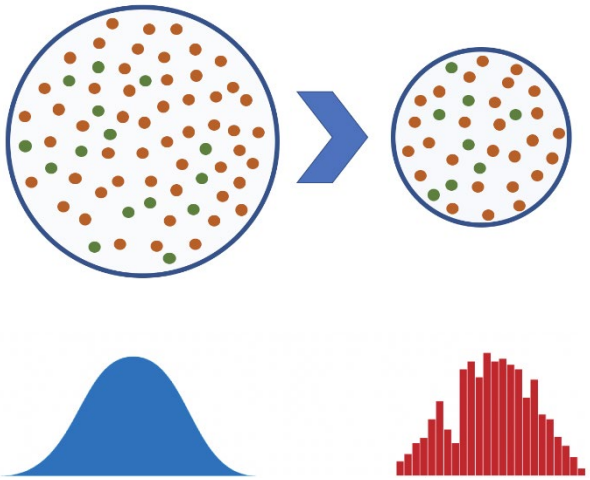
Method-oriented papers: Might not require a linguistic construct, for example focus on machine learning accuracy

➢ Construct operationalization must be aligned with construct definitions and the theoretical framework

➢ Reviewers often question the "fit" between the theory / conceptual framework and the operationalization

Humphreys and Wang 2017

3a. Sampling



Important considerations for sample selection in automated text analysis:

- **Selection Bias of Documents:** one over another social network
  - E.g., Twitter reflects a younger population than Facebook
- **Systematic Bias in Keyword Search:** carefully choose keywords
  - Researchers miss important data because they have the wrong phrasing or keyword
- **Sampling Diversity:** Ensure to capture different categories of documents
  - E.g., stratified sampling because online reviews are biased towards 4 and 5 stars
- **Sample Size**. Two issues that are highly contextual:
  - ***Number of words per document***, e.g. Facebook posts vs. Tweets in terms of length. Documents with too few words might suffer from low frequency (too many 0's)
  - ***Number of Documents***, e.g. statistical tests as well as overfitting or underfitting in machine learning (the statistical model cannot adequately capture the underlying structure of the data)

As a starting rule of thumb, having at least 30 units is usually needed to make statistical inferences!

3b. Data Collection

- See "Further Reading on Web Scraping"
- See "Packages for Data Collection in R"

Where can I find "Text" to Answer my Research Question?

If you are considering automated text analysis as a research method then you must have text data of some kind to be analyzed. Your text data may be:

- **created as a part of your research**, e.g. survey responses, interview transcripts, audio/video taped observations, industry collaboration
- **collated as part of your research**, e.g. journal articles for literature review
- **collated by a third party**, e.g. Senate enquiry transcripts, British National Corpus, Trump Twitter Archive
- **collated via web scraping**, e.g. news feeds, social media posts and comments, website content

<u>Non-Proprietary Sources of text data include:</u>

Library databases, Social media, Open sources, Web scraping, Language corpora, Discourses, Transcription of audio/video data...

- Data integrity and acknowledging the source of your data in important!

3c. Data Preparation

Also called **data-preprocessing**:

1. Document, paragraph, or sentence analysis
2. Language filtering
3. Spelling mistakes
4. Case standardization
5. Emojis, Emoticons, Hashtags, @, question marks
6. Stop words, negations, stemming, lemmatizing, etc.
7. Word Filtering

- Link all summary and background information by one ID variable
- Sometimes stop words are important ("Linguistic Style Matching")
- When mining for writing style, stemming can mask the tense used

4. Choose an Operationalization Approach

Three overall approaches and seven main methods associated:

Top-Down Approach (Dictionary- and Rule-based)

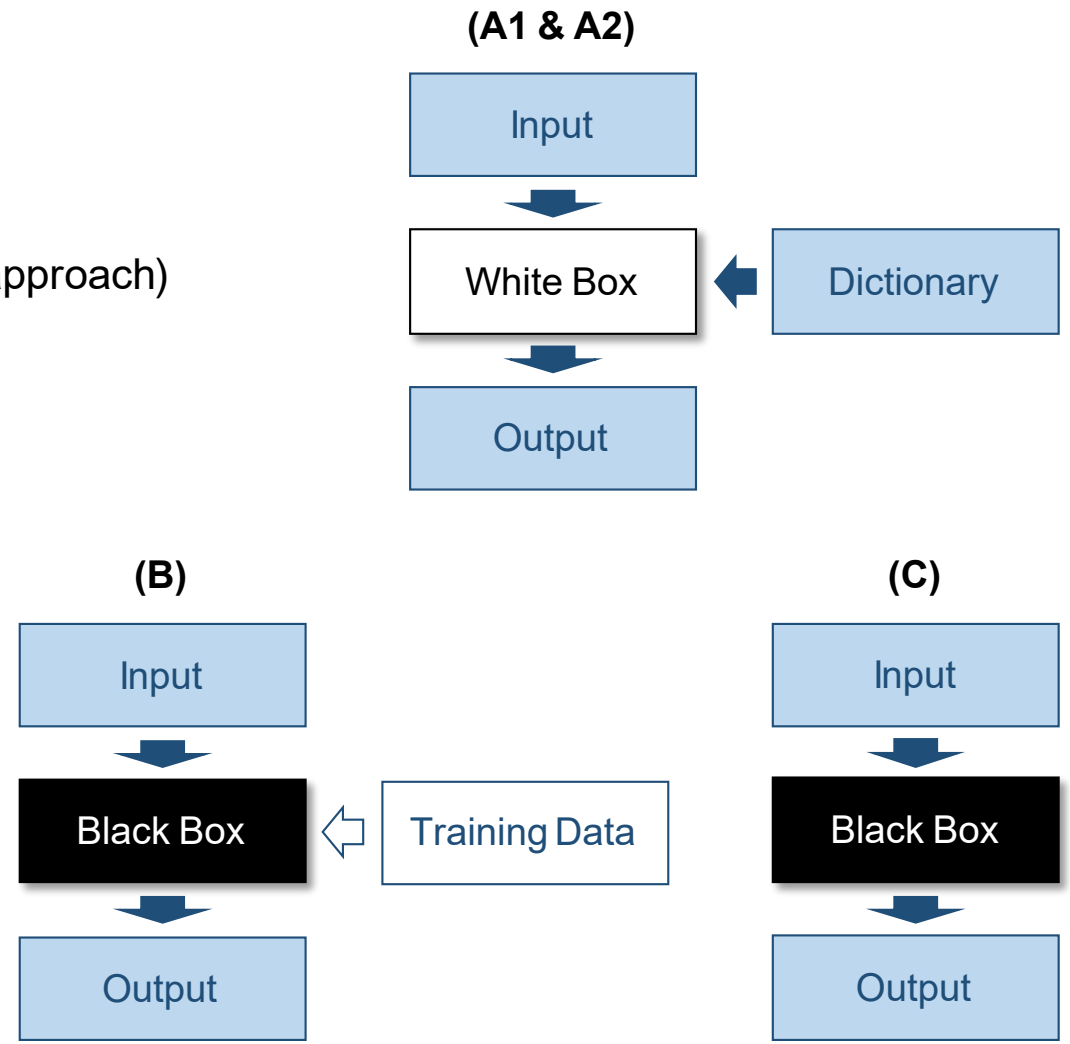
- (A1) Existing Dictionary (e.g., LIWC)
- (A2) Creating a Dictionary (e.g., grounded theory or concordance approach)
- Rule-Based Approaches (e.g., taxonomies)

Bottom-Up Approach (Classification and Topic Discovery)

- (B) Supervised ML (i.e., classification of labelled documents)
- (C) Unsupervised ML (i.e., patterns of unlabeled documents)
- Semantic Networks (i.e., word association patterns)
- Word Embeddings and Deep Learning

Mixed Approaches

- Combinations of the previous approaches

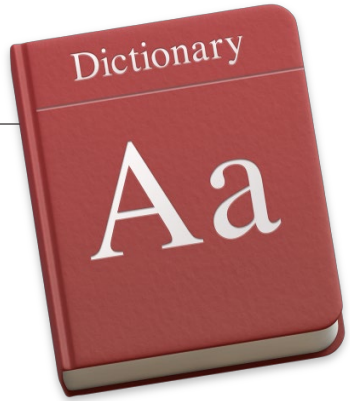


Top Down Approach: Dictionaries

White Box

- Also called “lexicon” or “linguistic approaches”
- Certain words are used to measure a certain category
- Using **existing dictionaries** from a determined field.
  - *For example GI (General Inquirer), LIWC (Linguistic Inquiry and Word Count), MPQA (Multi-Perspective Question Answering) Opinion Corpus, etc.*
- **Developing a dictionary** and/or rules for a specific domain.
  - *For example Customer Experience Feedback, Online Review Genre, Customer Service Interaction, Online Review Cultures, etc.*
- Dictionaries can have a hard coding (word “1” or “0”) or a soft coding (using word weights)

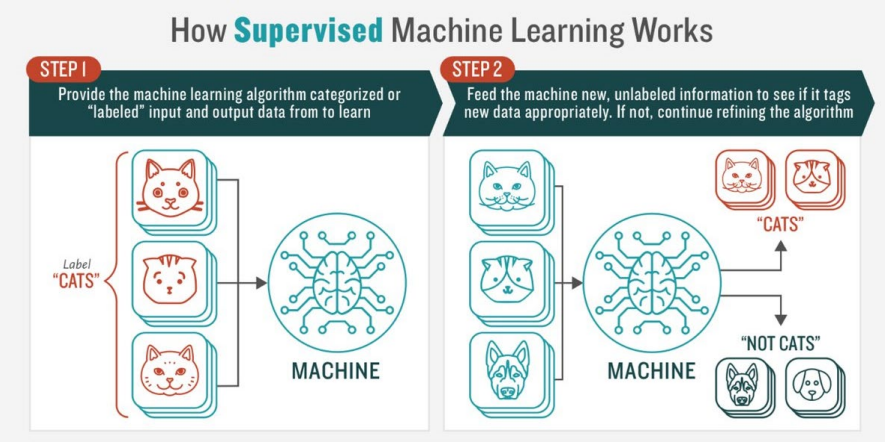
Category	Examples
Positive emotion	Love, nice, sweet
Negative emotion	Hurt, ugly, nasty
Anxiety	Worried, nervous
Anger	Hate, kill, annoyed
Sadness	Crying, grief, sad
Cognitive processes	Cause, know, ought
Insight	Think, know, consider
Causation	Because, effect, hence
Discrepancy	Should, would, could
Tentative	Maybe, perhaps, guess
Certainty	Always, never
Inhibition	Block, constrain, stop
Inclusive	And, with, include
Exclusive	But, without, exclude

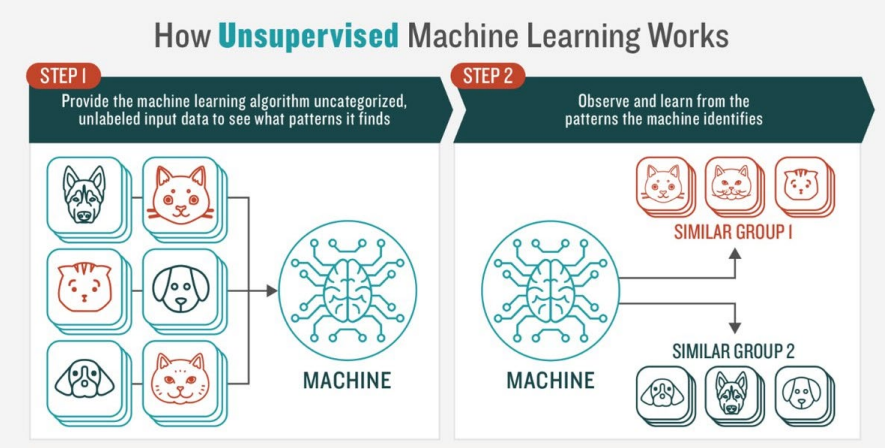


Bottom-Up Approach: Machine Learning

Black Box

- Involves building a text classifier, which is able to **automatically classify text** into different categories.
- There are two general approaches:
  - In **supervised learning**, the output datasets are provided and then used to train the machine and get the desired outputs
  - In **unsupervised learning** no output datasets are provided, instead the data is clustered into different classes. Other sources also call this process clustering



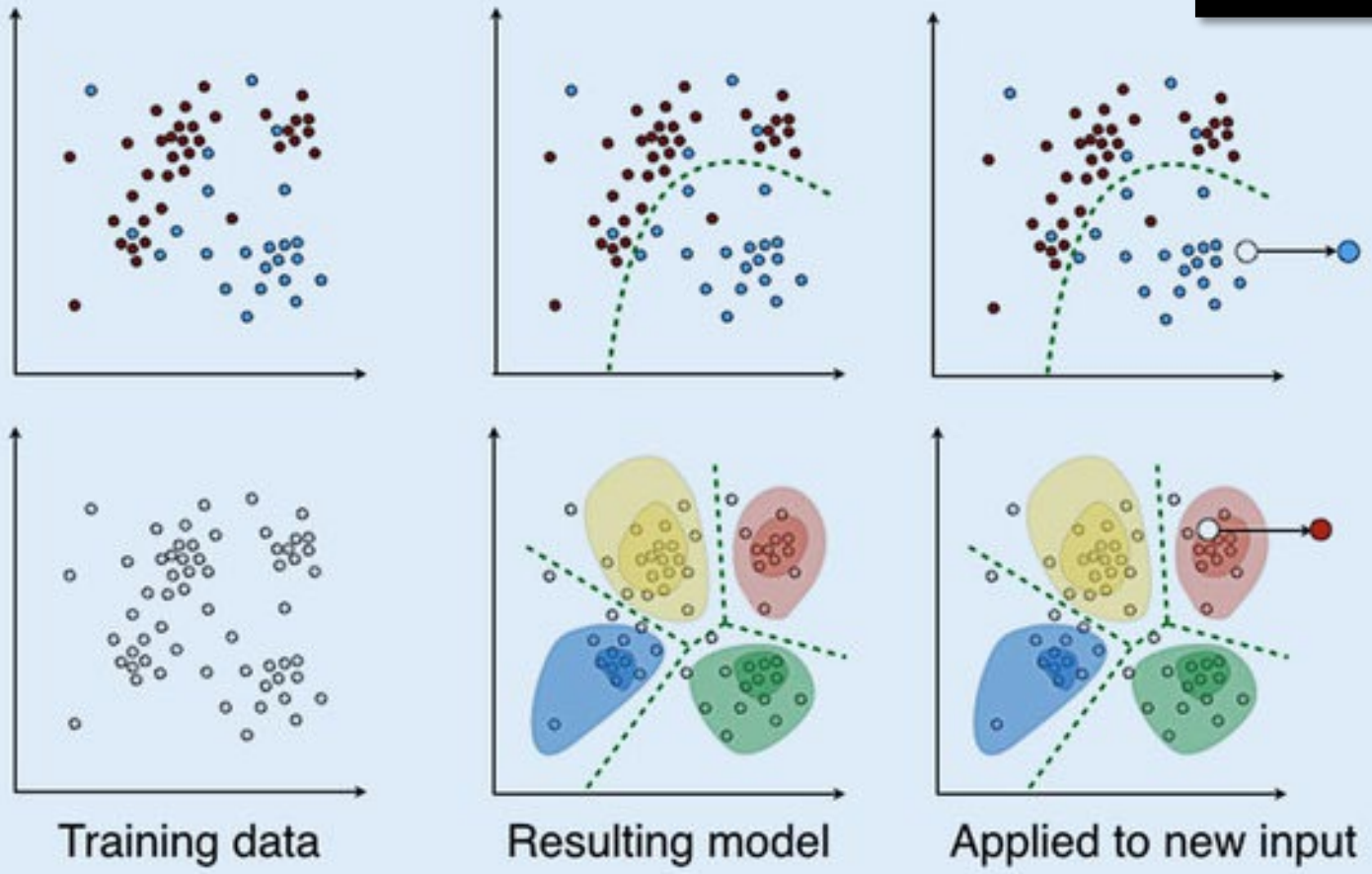


Supervised Learning vs. Unsupervised Learning

Black Box

Supervised learning: each training example has a ground truth label. The model learns a decision boundary and replicates the labeling on new data.

Unsupervised learning: training examples do not have ground truth labels. The model identifies structure such as clusters. New data can be assigned to clusters.

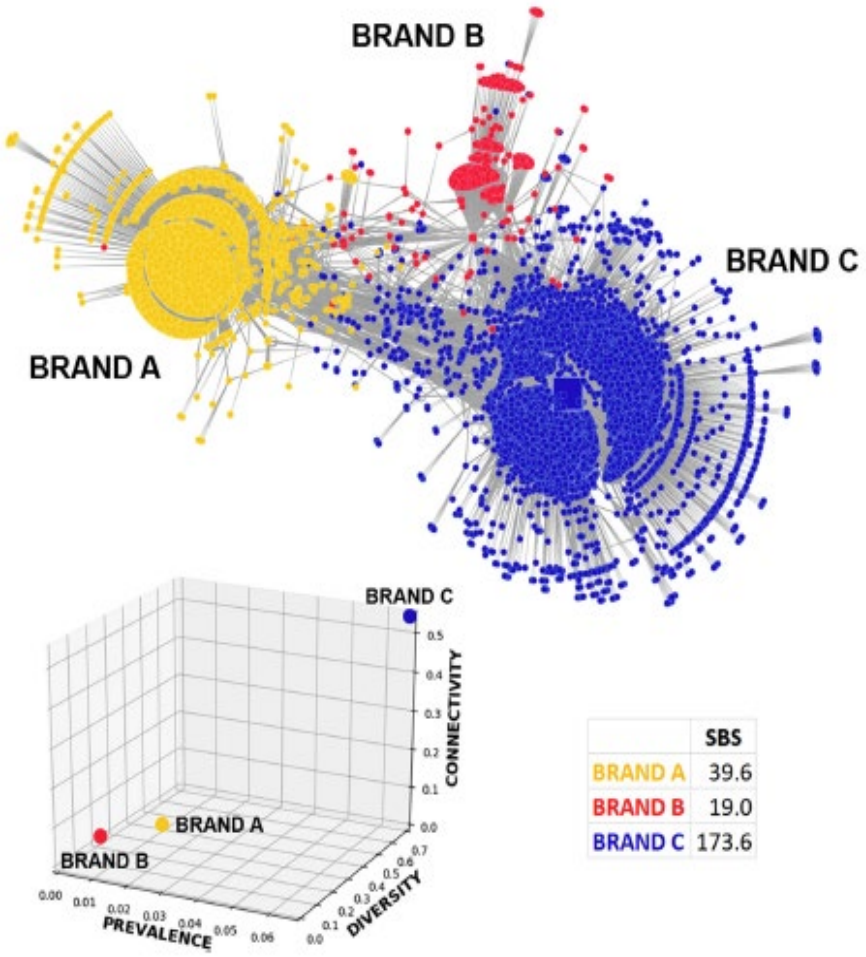


Mixed Approaches

White Box & Black Box

Examples:

- Dictionaries can be used as a **prediction feature for supervised machine learning** and classify brand messages into different types (*How Speech and Image Acts Drive Consumer Sharing, Villarroel Ordenes et al. 2019*).
- Exploring latent topics in the data (LDA), and then using the words that describe each topic as **seeds to develop a dictionary** (*Topic Detection from Microblogs, Huang et al. 2017*).
- Although generally used for exploring latent relationships, the use of **semantic network analysis** can be used to operationalize defined constructs such as brand connectivity (*The Semantic Brand Score, Fronzetti Colladon 2018*).
- ...



	SBS
BRAND A	39.6
BRAND B	19.0
BRAND C	173.6

Top-Down and Bottom-Up Classification

Dictionary-based Top-Down Approach

- *Easy to* ***implement and comprehend***, especially for researchers that have limited programming or coding experience
- Combined with the fundamentals of linguistics, they allow ***intuitive operationalization*** of constructs and theories directly from sociology or psychology.
- The ***validation process*** of dictionary-based approaches is relatively straight forward for non-specialists, and ***findings are relatively transparent*** to reviewers and readers.
- Note that when a dictionary-based approach is used, ***tests will be conservative***. That is, by predetermining a word list, one may not pick up all instances of what one wants to measure, but if meaningful patterns emerge, one can argue that there is an effect, despite the omissions.

Supervised ML-based Bottom-Up Approach

- Reduces the ***amount of human coding required and dependence*** yet produces clear distinctions between texts.
- While dictionary-based approaches provide information related to magnitude, supervised ML approaches provide information about ***type and likelihood of being of a type***, and researchers can go a step further by understanding what words or patterns lead to being classified as a type.
- The classification model itself can reveal insights or test hypotheses that may be otherwise buried in the data. Latent elements, such as ***surprising combinations of words or patterns***, may be revealed.

5. Analyze and Interpret the Results

Comparison between groups

Text used to represent the construct and comparisons are made to assess statistical differences between the groups. Comparisons can be regarding groups, space, and time

Robustness checks: Using non-parametric tests (e.g., Kruskal-Wallis test), using TF-IDF rather than TF, ...

Correlation between textual and non-textual elements

Patterns of association between textual elements or between textual and non-textual elements such as ratings.

Robustness checks: Random data subset and repeating analysis, using varying distance measures, ...

If causal hypotheses: Augment field data with experiments to allow key independent variables to be manipulated

Prediction of variables outside the text

Uses textual data to predict a different non-textual variable. For example predicting whether a review is false or not, predicting message type or credit default.

Robustness checks: Endogeneity using a data subsample, cross-validation, ...

- Check robustness tests with a GOOD paper using the same type of automated text analysis in your field
- Always check the Web Appendix – often robustness tests are reported there...

6. Internal and External Validation

Internal Validity

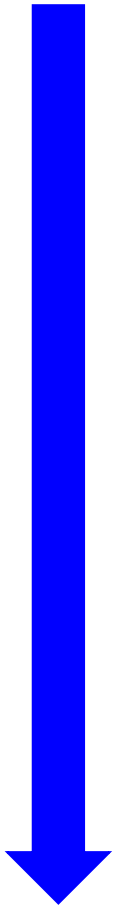
- ***Construct Validity:*** Consistent construct operationalization, human coders to validate the text mining dictionary
- ***Concurrent Validity***
  - *Use existing Dictionaries:* Compare to inferences of previous studies
  - *Multiple Dictionaries:* Calculate and compare multiple textual measures of the same construct
  - *Comparison of Topics:* Compare with other topic models of similar data sets in other research
- ***Convergent Validity***
  - *Triangulation:* Look for converging patterns (e.g., positive/e emotion correlates with known-positive attributes)
  - *Multiple Operationalizations:* Operationalize constructs with textual and non-textual data (e.g., sentiment, star rating)
- ***Causal Validity:*** Use control variables, IV or LIV, replicate focal relationships in a laboratory setting

External Validity

- ***Generalizability:*** Replication with different data sets or (randomized) subsamples
- ***Predictive validity:*** Hold out sample and cross validation (applied in Machine Learning)
- ***Robustness:*** Use different, but comparable, statistical measures or algorithms

- The review team will remind you of the importance of internal and external validation...

Now it is Up to You...



1. Identify a Research Question
2. Find Related Constructs
3. Sampling, Data Collection, Preparation
4. Choose an Operationalization Approach
5. Analyze and Interpret the Results
6. Internal and External Validation

Agenda: Automated Text Analysis

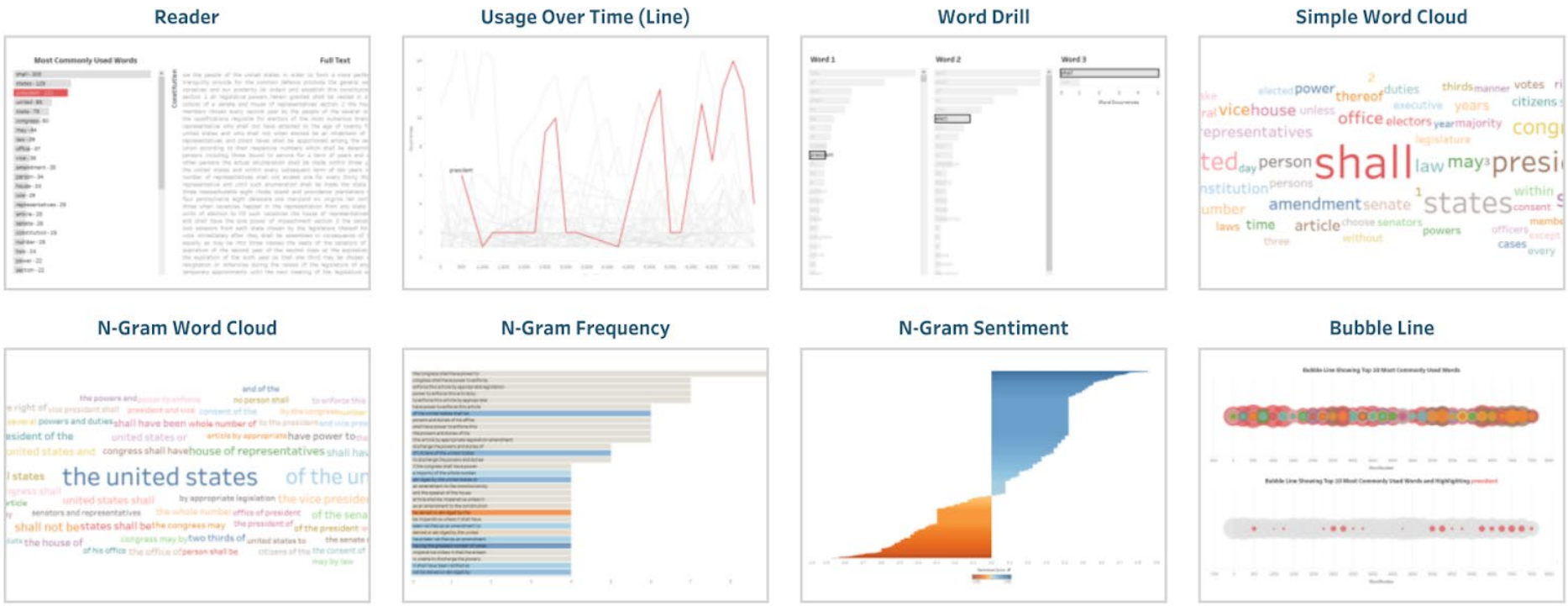
1) What is Automated Text Analysis?
2) A Roadmap for Automated Text Analysis
   - ➢ ***Getting Started with R and RStudio***
3) Data Preparation and Data Visualization
4) Classification with Dictionaries
5) Classification with Supervised Machine Learning
6) Clustering and Topic Discovery

Disclaimer

This is a non-technical introduction that should enable you to understand and use the most common methods in text mining.

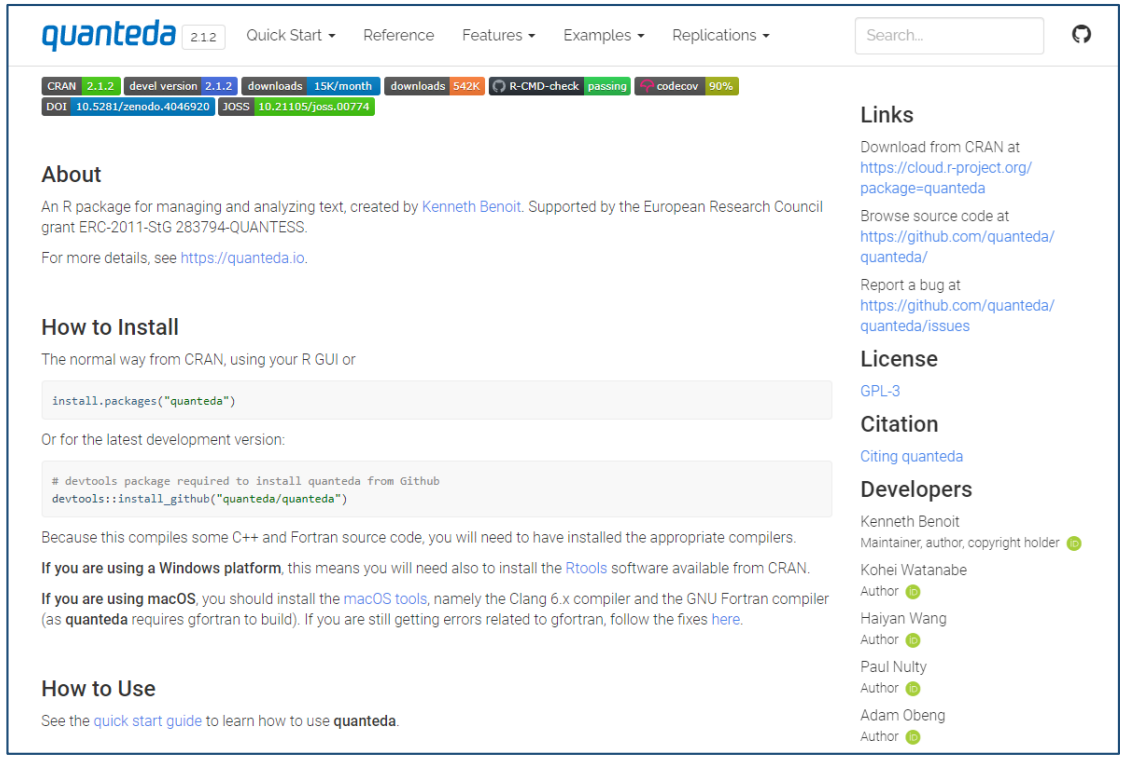
Thus, the training oversimplifies concepts, does not address all relevant aspects, and does not appropriately explain most technical aspects. For more details and technical aspects please consult the "Further Reading on Automated Text Analysis".

3) Data Preparation and Data Visualization



Text is often unstructured and "messy," so before any formal analyses can take place, researchers must first preprocess the text itself. This step provides structure and consistency so that the text can be used systematically in the scientific process.

Automated Text Analysis with R/RStudio and quanteda



➢ Other packages are available but more complicated (at least for me)

The **quanteda package** is an extensive text analysis suite for R.

It covers everything to perform a variety of automatic text analysis techniques, and features clear and extensive documentation.

https://quanteda.io/

The documentation for each function can be found here: https://quanteda.io/reference/index.html

For a more detailed explanation please see "Text Analysis in R" from Welbers, van Atteveldt & Benoit, 2017). http://vanatteveldt.com/p/welbers-text-r.pdf

Getting Started with quanteda

quanteda: Core package that we will use in the following
quanteda.textmodels: Scaling models and classifiers for textual data in the form of a document-feature matrix
topicmodels: Topic models allow the probabilistic modeling of term frequency occurrences in documents
readtext: An easy way to read text data into R, from almost any input format
openxlsx: An easy way to read, write and edit .xlsx files in R
ggplot2: An easy way for creating aesthetically graphics

You only need to install a package once, but **you need to reload it every time you start a new session**!

```
# Install the quanteda package and additional packages
install.packages("quanteda")
install.packages("quanteda.textmodels")
install.packages("topicmodels")
install.packages("readtext")
linstall.packages("openxlsx")
linstall.packages("ggplot2")

# Load the packages
library(quanteda)
library(quanteda.textmodels)
library(topicmodels)
library(readtext)
library(openxlsx)
library(ggplot2)
```

How quanteda works: The corpus Principle

(text) corpus a large and structured set of texts for analysis

A corpus is designed to be a "library" of original documents that have been converted to plain, UTF-8 encoded text, and stored along with meta-data at the *corpus level* and at the *document-level*.

A corpus is designed to be a static container of texts with respect to processing and analysis. This means that the texts in corpus are not designed to be changed internally through cleaning or pre-processing steps.

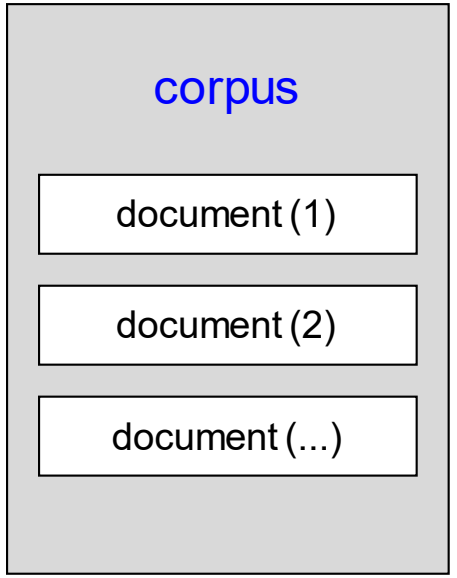
Texts can be extracted from the corpus as part of processing, and assigned to new objects, but the idea is that the corpus will remain as an original reference copy so that other analyses can be performed on the same corpus.

document each of the units of the corpus

The name for document-level meta-data is *docvars*. These variables describe attributes of each document.

types refer to a unique word for our purposes

tokens refer to any word → so token count is total words



A corpus is a set of documents.
This is the second document in the corpus.

- This is a corpus with 2 documents, where each document is a sentence. The first document has 6 types and 7 tokens. The second document has 7 types and 8 tokens (we ignore punctuation for now).

https://quanteda.io/

What can the document be?

Words, n-word sequences, sentences, pages, paragraphs, **natural units (a review, a post, a Tweet, a speech, a poem, a manifesto...)**, aggregation of units (e.g., all speeches by party and year), etc.

➢ Depends on the research design: Frequent trade-off between complexity and accuracy

Sampling strategies for selecting texts

- Difference between a sample and a population
  - May not be feasible to perform any sampling
  - May not be necessary to perform any sampling
- Be wary of sampling that is based on "non-random availability" (e.g., deleted Facebook posts)
- Different types of sampling vary from random to purposive
  - random sampling
  - non-random sampling

➢ Make sure that what is being analyzed is a valid representation of the phenomenon as a whole (→ research design)

https://quanteda.io/

Creating the Corpus

quanteda has a simple and powerful companion package for loading texts: ***readtext***. The main function in this package, *readtext()*, takes a file or a set of files from disk or a URL, and returns a type of *data.frame* that can be used directly with the *corpus()* constructor function, to create a **quanteda corpus object**.

readtext() works on multiple formats (https://cran.r-project.org/web/packages/readtext/vignettes/readtext_vignette.html):

- text (.txt) files, comma-separated-value (.csv) files, XML formatted data, data from the Facebook API in JSON format, data from the Twitter API in JSON format, generic JSON data, ...

The corpus constructor command ***corpus()*** works on the resulting *data.frame* containing a text column.

```
#Load example data from the readtext package (five inaugural speeches) into "dat_inaug"
path_data <- system.file("extdata/", package = "readtext")
dat_inaug <- readtext(paste0(path_data, "/csv/inaugCorpus.csv"), text_field = "texts")

# Construct a corpus of five inaugural speeches
corp_inaug <- corpus(dat_inaug)

# Summarize the information in the corpus
summary(corp_inaug)
```

➢ Automatically detects your path

https://quanteda.io/

Loading the Data can be a bit Tricky...

The most common problems related to loading data into R are wrong syntax and misspecifications of file locations!

Sometimes it is necessary to extensively clean data:
https://www.rdocumentation.org/packages/stringr/versions/1.4.0

stringr is a simple wrapper for common string operations.
Strings play a big role in many data cleaning and preparation tasks.

➢ **More advanced topic no covered today**

Example: Import text data from a .xls File ➢ **As an alternative, use the "Import Dataset" function in RStudio**

Save "Social_Media_Data.xlsx" as ***Text (Tab Delimited) (.txt)***

Rename "Social_Media_Data.txt" into "Social_Media_Data.tsv"

```
# read in "Social_Media_Data" as .tsv file
(rt_smd <- readtext("C:/01 Teaching/KEDGE # Text Mining/Social_Media_Data.tsv", text_field = "TEXT"))
###Use your own folder!!!###

# Construct corpus of "Social_Media_Data"
data_smd <- corpus(rt_smd)

# Summary of "Social_Media_Data", showing 10 documents
summary(data_smd, n = 10)
```

Exercise with Your Own Data

- **Loading your own text data into quanteda (including document variables)**
    1. Store data in an appropriate format
    2. Import data with “readtext” or “Import Dataset” into RStudio
    3. Import data with “corpus” into quanteda

Try to use and adapt the code from the examples!

Example “data_corpus_inaugural”

corpus = all 58 inaugural speeches from the US presidents

document = each inaugural speech (labeled with year, president, and party)

To summarize the texts from a corpus, we can call a *summary()*:

```
# Summarize the information in the corpus
summary(data_corpus_inaugural)

# Same as above but only 10 first entries
summary(data_corpus_inaugural, n = 10)
```

```
> # Same as above but only 10 entries
> summary(data_corpus_inaugural, n = 10)
Corpus consisting of 58 documents, showing 10 documents:

           Text Types Tokens Sentences Year  President   FirstName                Party
 1789-Washington   625   1537        23 1789 Washington      George                 none
 1793-Washington    96    147         4 1793 Washington      George                 none
      1797-Adams   826   2577        37 1797      Adams        John             Federalist
  1801-Jefferson   717   1923        41 1801  Jefferson      Thomas Democratic-Republican
  1805-Jefferson   804   2380        45 1805  Jefferson      Thomas Democratic-Republican
    1809-Madison   535   1261        21 1809    Madison       James Democratic-Republican
    1813-Madison   541   1302        33 1813    Madison       James Democratic-Republican
     1817-Monroe  1040   3677       121 1817     Monroe       James Democratic-Republican
     1821-Monroe  1259   4886       131 1821     Monroe       James Democratic-Republican
      1825-Adams  1003   3147        74 1825      Adams John Quincy Democratic-Republican
```

To extract texts from a corpus, we can use an extractor, *called texts()*:

```
# Extract the last text, the inaugural speech of Donald Trump
texts(data_corpus_inaugural)[58]
```

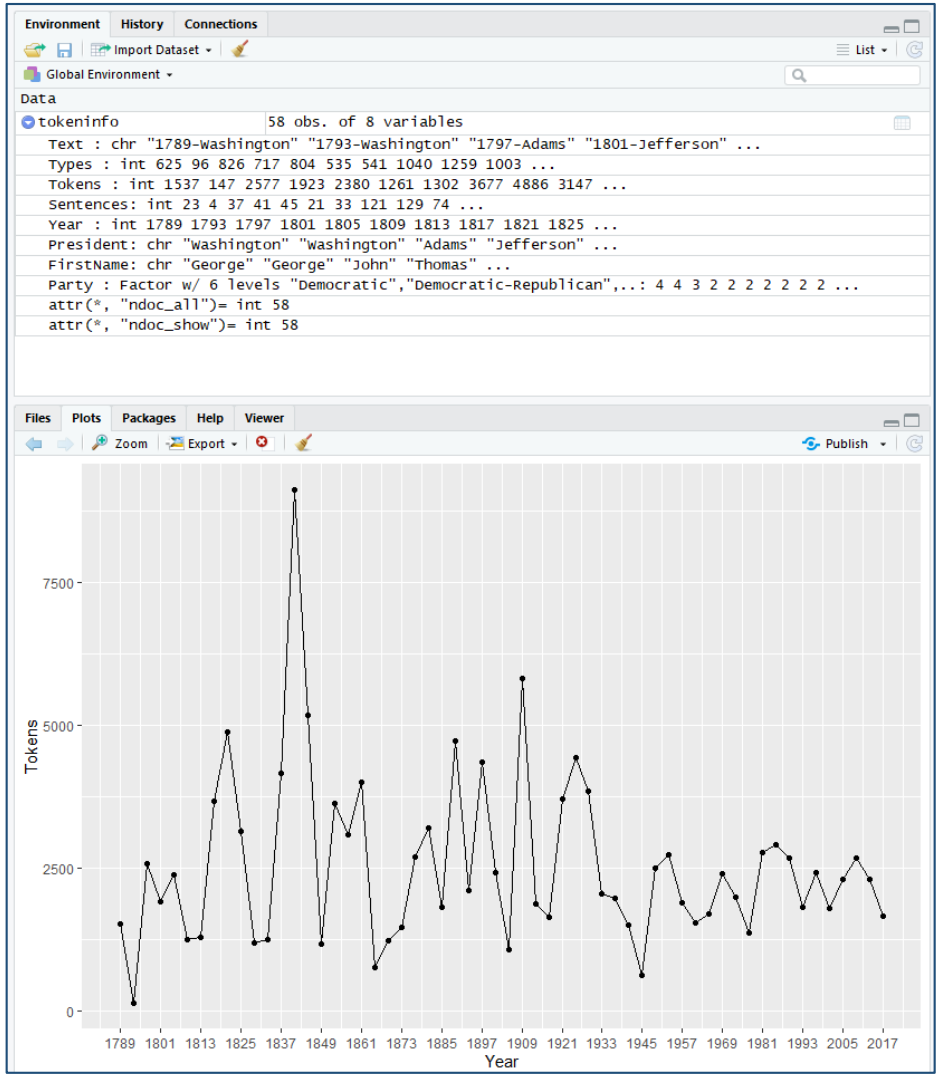
```
                                                                                                2017-Trump
"Chief Justice Roberts, President Carter, President Clinton, President Bush, President Obama, fellow Americans, and people of
 the world: thank you.\n\nWe, the citizens of America, are now joined in a great national effort to rebuild our country and r
estore its promise for all of our people.\n\nTogether, we will determine the course of America and the world for many, many y
ears to come.\n\nWe will face challenges. We will confront hardships. But we will get the job done.\n\nEvery four years, we g
ather on these steps to carry out the orderly and peaceful transfer of power, and we are grateful to President Obama and Firs
t Lady Michelle Obama for their gracious aid throughout this transition. They have been magnificent. Thank you.\n\nToday's ce
remony, however, has very special meaning. Because today we are not merely transferring power from one Administration to anot
her, or from one party to another - but we are transferring power from Washington DC and giving it back to you, the people.\n
\nFor too long, a small group in our nation's Capital has reaped the rewards of government while the people have borne the co
st.\n\nWashington flourished - but the people did not share in its wealth.\n\nPoliticians prospered - but the jobs left, and
 the factories closed.\n\nThe establishment protected itself, but not the citizens of our country.\n\nTheir victories have no
t been your victories; their triumphs have not been your triumphs; and while they celebrated in our nation's capital, there w
as little to celebrate for struggling families all across our land.\n\nThat all changes - starting right here, and right now,
 because this moment is your moment: it belongs to you.\n\nIt belongs to everyone gathered here today and everyone watching a
ll across America.\n\nThis is your day. This is your celebration.\n\nAnd this, the United States of America, is your countr
y.\n\nWhat truly matters is not which party controls our government, but whether our government is controlled by the peopl
e.\n\nJanuary 20, 2017, will be remembered as the day the people became the rulers of this nation again.\n\nThe forgotten men
 and women of our country will be forgotten no longer.\n\nEveryone is listening to you now.\n\nYou came by the tens of millio
ns to become part of a historic movement the likes of which the world has never seen before.\n\nAt the center of this movemen
t is a crucial conviction: that a nation exists to serve its citizens.\n\nAmericans want great schools for their children, sa
```

https://quanteda.io/

Some First Descriptive Analyses

```
# Store info from summary into "tokeninfo"
tokeninfo <- summary(data_corpus_inaugural)

# Longest inaugural speech (number of "tokens" = words)?
tokeninfo[which.max(tokeninfo$Tokens), ]

# Shortest inaugural speech (number of "tokens" = words)?
tokeninfo[which.min(tokeninfo$Tokens), ]

# Plot associations between "Year" and "Tokens" (Number of Words)
ggplot(data = tokeninfo, aes(x = Year, y = Tokens)) +
  geom_line() + geom_point() +
  scale_x_continuous(labels  = c(seq(1789, 2017, 12)),
  breaks = seq(1789, 2017, 12))
```

- for the ggplot syntax, see https://github.com/rstudio/cheatsheets/raw/master/data-visualization-2.1.pdf



https://quanteda.io/

Further Descriptive Analyses

Locate keywords-in-context

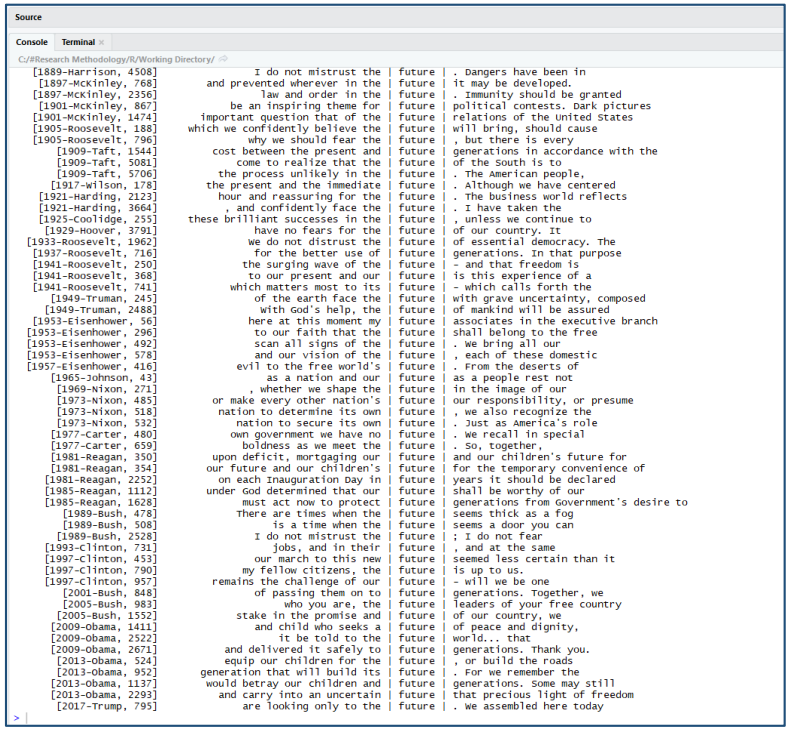
You can see how keywords are used in the actual contexts in a concordance view produced by ***kwic()***

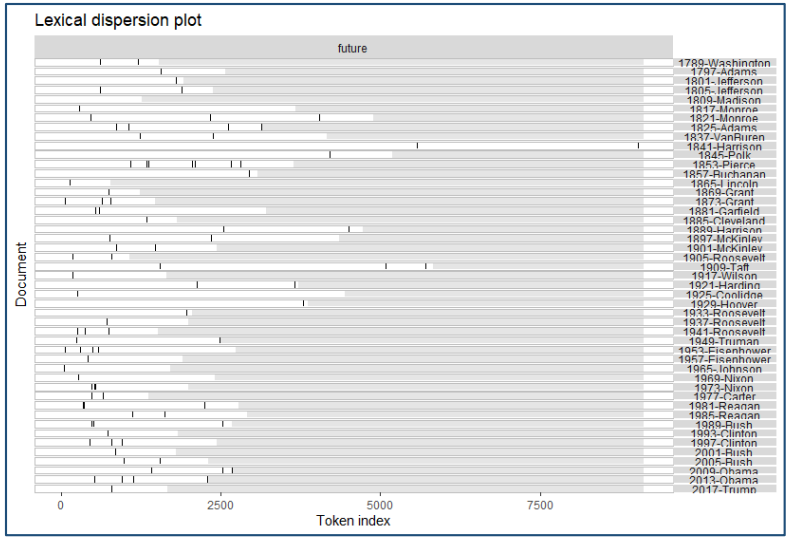
```
# See in which context the word "future“ was used
kw_future <- kwic(data_corpus_inaugural,  pattern =  "future")
kw_future

# Search for phrase, increase window, and see in html
kw_bless <- kwic(data_corpus_inaugural,  phrase("god bless"), window = 10)
View(kw_bless)
```

Plot the dispersion of keywords

Plots a dispersion or "x-ray" plot of selected word pattern(s) across texts.

```
# Plot the pattern for "future" with absolute lenght
textplot_xray(kwic(data_corpus_inaugural,  pattern = "future"), scale = "absolute")
```



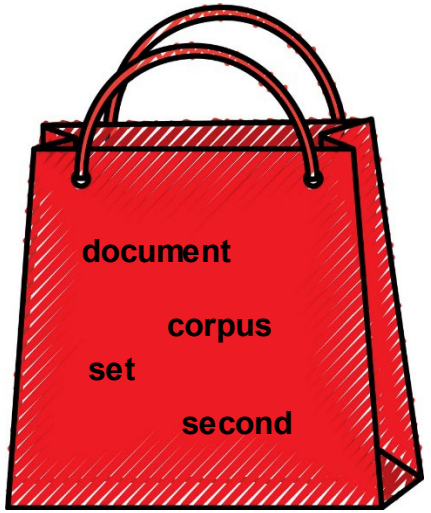


https://quanteda.io/

Creating the Document-Feature Matrix ("Bag-of-Words")

The bag-of-words approach **disregards grammar and word order** and uses **word frequencies**.

Many text analysis techniques only use the frequencies of words in documents. This is also called the bag-of-words assumption, because texts are then treated as bags of individual words. Despite ignoring much relevant information in the order of words and syntax, this approach has proven to be very powerful and efficient.



Why?

- Context is often uninformative: Individual word usage tends to be associated with a particular degree of e.g., affect without regard to the context of word usage
- Single words tend to be the most informative, co-occurrences of multiple words (*n-grams*) are rather rare

Some approaches focus on the occurrence of a word as a **binary variable** (not frequency)

LIWC

The way that LIWC works is similar and fairly simple. Basically, it reads a given text and counts the percentage of words that reflect different emotions and other categories. Most of the LIWC output variables are percentages of total words within a text. For example, imagine you have analyzed a blog and discover that the Positive Emotions (or "posemo") number was 4.20. That means that 4.20 percent of all the words in the blog were positive emotion words.

https://quanteda.io/

Key Concepts of the Bag-of-Words Approach

Our basic automated text analysis adopts a bag-of-words approach!

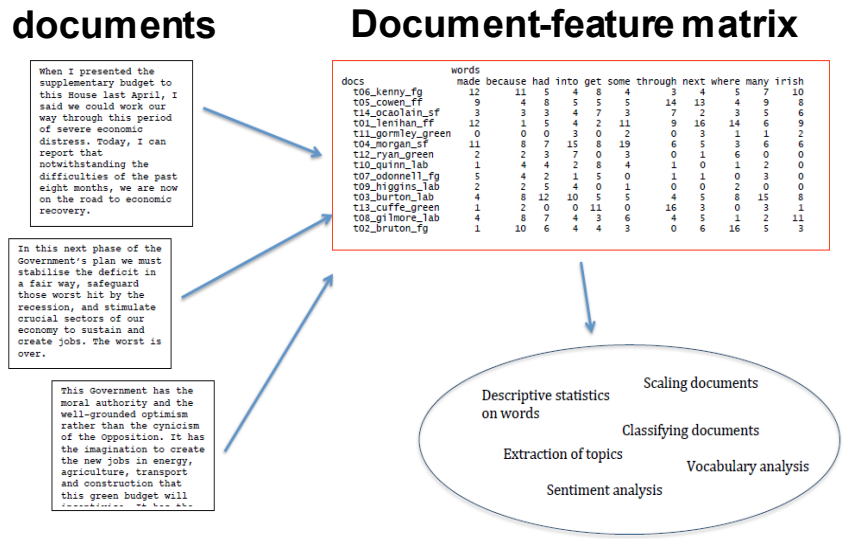
stems words with suffixes removed (using a set of predefined rules)

lemmas canonical word form (the base form of a word that has the same meaning even when different suffixes or prefixes are attached)

Main difference: ***Stemming*** just finds any base form, which is not always a word in the language! ***Lemmatization*** finds the actual root of a word with morphological analysis.

word	win	winning	wins	won	winner
stem	win	win	win	won	winner
lemma	win	win	win	win	win

➢ **Lemmatization is a more advanced topic no covered today**



stop words words that are designated for exclusion from any analysis of a text

"key" words words selected because of special attributes or rates of occurrence

https://quanteda.io/

Common English stop words

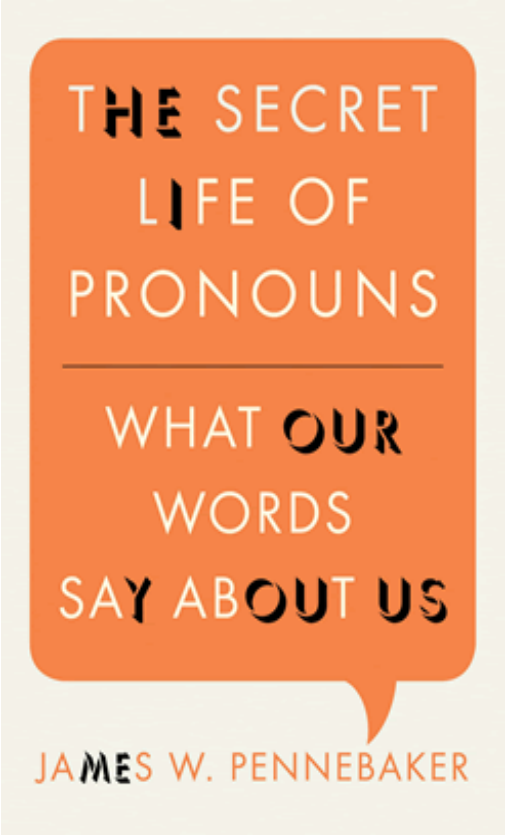
a, able, about, across, after, all, almost, also, am, among, an, and, any, are, as, at, be, because, been, but, by, can, cannot, could, dear, did, do, does, either, else, ever, every, for, from, get, got, had, has, have, he, her, hers, him, his, how, however, I, if, in, into, is, it, its, just, least, let, like, likely, may, me, might, most, must, my, neither, no, nor, not, of, off, often, on, only, or, other, our, own, rather, said, say, says, she, should, since, so, some, than, that, the, their, them, then, there, these, they, this, tis, to, too, us, wants, was, we, were, what, when, where, which, while, who, whom, why, will, with, would, yet, you, your

➢ **but no list should be considered universal!**

https://quanteda.io/

A more comprehensive list of stop words...

as, able, about, above, according, accordingly, across, actually, after, afterwards, again, against, ain't, all, allow, allows, almost, alone, along, already, also, although, always, am, among, amongst, an, and, another, any, anybody, anyhow, anyone, anything, anyway, anyways, anywhere, apart, appear, appreciate, appropriate, are, aren't, around, as, aside, ask, asking, associated, at, available, away, awfully, be, became, because, become, becomes, becoming, been, before, beforehand, behind, being, believe, below, beside, besides, best, better, between, beyond, both, brief, but, by, c'mon, c's, came, can, can't, cannot, cant, cause, causes, certain, certainly, changes, clearly, co, com, come, comes, concerning, consequently, consider, considering, contain, containing, contains, corresponding, could, couldn't, course, currently, definitely, described, despite, did, didn't, different, do, does, doesn't, doing, don't, done, down, downwards, during, each, edu, eg, eight, either, else, elsewhere, enough, entirely, especially, et, etc, even, ever, every, everybody, everyone, everything, everywhere, ex, exactly, example, except, far, few, fifth, first, five, followed, following, follows, for, former, formerly, forth, four, from, further, furthermore, get, gets, getting, given, gives, go, goes, going, gone, got, gotten, greetings, had, hadn't, happens, hardly, has, hasn't, have, haven't, having, he, he's, hello, help, hence, her, here, here's, hereafter, hereby, herein, hereupon, hers, herself, hi, him, himself, his, hither, hopefully, how, howbeit, however, i'd, i'll, i'm, i've, ie, if, ignored, immediate, in, inasmuch, inc, indeed, indicate, indicated, indicates, inner, insofar, instead, into, inward, is, isn't, it, it'd, it'll, it's, its, itself, just, keep, keeps, kept, know, knows, known, last, lately, later, latter, latterly, least, less, lest, let, let's, like, liked, likely, little, look, looking, looks, ltd, mainly, many, may, maybe, me, mean, meanwhile, merely, might, more, moreover, most, mostly, much, must, my, myself, name, namely, nd, near, nearly, necessary, need, needs, neither, never, nevertheless, new, next, nine, no, nobody, non, none, noone, nor, normally, not, nothing, novel, now, nowhere, obviously, of, off, often, oh, ok, okay, old, on, once, one, ones, only, onto, or, other, others, otherwise, ought, our, ours, ourselves, out, outside, over, overall, own, particular, particularly, per, perhaps, placed, please, plus, possible, presumably, probably, provides, que, quite, qv, rather, rd, re, really, reasonably, regarding, regardless, regards, relatively, respectively, right, said, same, saw, say, saying, says, second, secondly, see, seeing, seem, seemed, seeming, seems, seen, self, selves, sensible, sent, serious, seriously, seven, several, shall, she, should, shouldn't, since, six, so, some, somebody, somehow, someone, something, sometime, sometimes, somewhat, somewhere, soon, sorry, specified, specify, specifying, still, sub, such, sup, sure, t's, take, taken, tell, tends, th, than, thank, thanks, thanx, that, that's, thats, the, their, theirs, them, themselves, then, thence, there, there's, thereafter, thereby, therefore, therein, theres, thereupon, these, they, they'd, they'll, they're, they've, think, third, this, thorough, thoroughly, those, though, three, through, throughout, thru, thus, to, together, too, took, toward, towards, tried, tries, truly, try, trying, twice, two, un, under, unfortunately, unless, unlikely, until, unto, up, upon, us, use, used, useful, uses, using, usually, value, various, very, via, viz, vs, want, wants, was, wasn't, way, we, we'd, we'll, we're, we've, welcome, well, went, were, weren't, what, what's, whatever, when, whence, whenever, where, where's, whereafter, whereas, whereby, wherein, whereupon, wherever, whether, which, while, whither, who, who's, whoever, whole, whom, whose, why, will, willing, wish, with, within, without, won't, wonder, would, would, wouldn't, yes, yet, you, you'd, you'll, you're, you've, your, yours, yourself, yourselves, zero



Sometimes stop words can be informative!

https://quanteda.io/

From Words to Numbers...

1. Preprocess text: (a) lowercase, (b) remove stopwords and punctuation, (c) stem, and (d) tokenize into unigrams

original

A corpus is a set of documents.
This is the second document in the corpus.

a corpus is a set of documents
this is the second document in the corpus."

~~**a**~~ corpus ~~**is a set of**~~ documents~~**.**~~
~~**this is the**~~ second document ~~**in the**~~ corpus~~**.**~~

corpus set document~~**s**~~
second document corpus

1

corpus, set, document
second, document, corpus

2. Document-feature matrix:
W: matrix of *N* documents by *M* unique n-grams
w_{im} = number of times *m*-th n-gram appears in *i*-th document.

Document-Feature Matrix

2

N Documents	*M* unique n-grams			
docs	corpus	set	document	second
document 1	1	1	1	0
document 2	1	0	1	1
...				
document n	1/0	1/0	1/0	1/0

https://quanteda.io/

From Words to Numbers in R

```
# Use the text from the example
text <-  c(d1 = "A corpus is a set of documents.",
           d2 = "This is the second document in the corpus.")

# Convert text into a Document-Feature Matrix "as it is"
dtm1 <- dfm(text)
dtm1

# Convert text into a Document-Feature Matrix with preprocessing
dtm2 <- dfm(text, tolower=TRUE, remove=stopwords("en"),
            remove_punct=TRUE, stem=TRUE)
dtm2

# Inspect the stop words used
stopwords("en")
```

```
> # Convert text into a Document-Feature Matrix "as it is"
> dtm1 <- dfm(text)
> dtm1
Document-feature matrix of: 2 documents, 12 features (37.5% sparse).
    features
docs a corpus is set of documents . this the second
  d1 2      1  1  1  1         1 1    0   0      0
  d2 0      1  1  0  0         0 1    1   2      1
[ reached max_nfeat ... 2 more features ]
```

```
> # Convert text into a Document-Feature Matrix with preprocessing
> dtm2 <- dfm(text, tolower=TRUE, remove=stopwords("en"), remove_punct=TRUE, stem=TRUE)
> dtm2
Document-feature matrix of: 2 documents, 4 features (25.0% sparse).
    features
docs corpus set document second
  d1      1   1        1      0
  d2      1   0        1      1
>
```

- If you are using texts in another language, make sure to specify the language, such as stopwords('fr') for French or stopwords('de') for German
- There are various alternative preprocessing techniques, including more advanced techniques that are not implemented in *quanteda*. Whether, when and how to use these techniques is a broad topic that we won't cover today
- For more details please see http://vanatteveldt.com/p/welbers-text-r.pdf.

https://quanteda.io/

Word Frequency: Zipf's Law

Given some corpus of natural language utterances, the frequency of any word is inversely proportional to its rank in the frequency table.

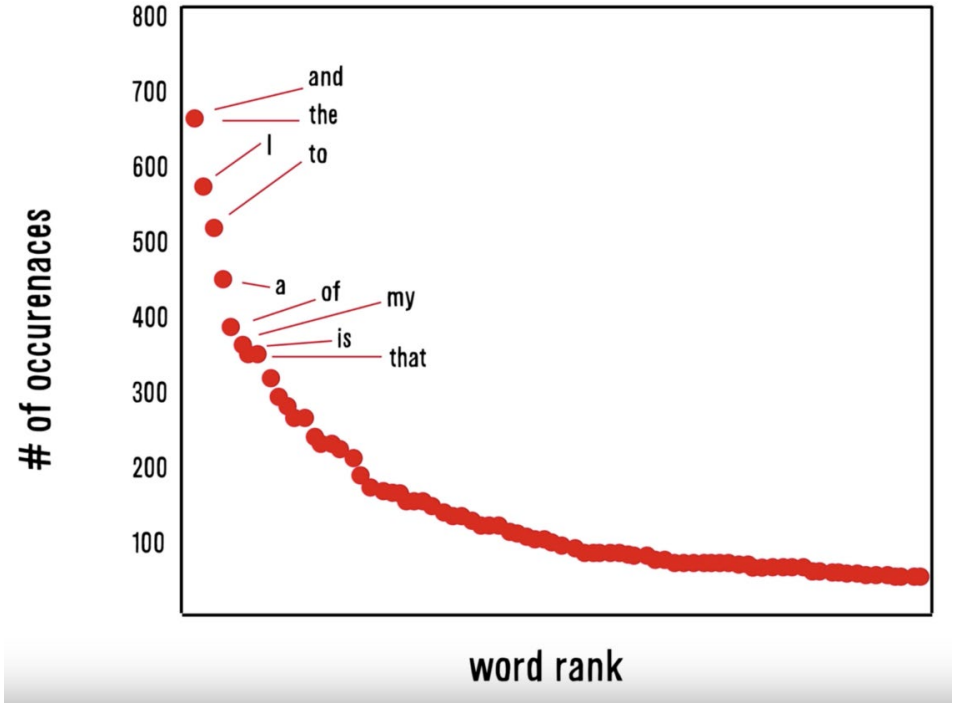
The simplest case of Zipf's law is a "***1/f function***"

> Given a set of Zipfian distributed frequencies, sorted from most common to least common, the second most common frequency will occur 1/2 as often as the first. The third most common frequency will occur 1/3 as often as the first. The *n-th* most common frequency will occur *1/n* as often as the first.

In the **English language**, the probability of encountering the most common word is given roughly by $P(r) = \frac{0.1}{r}$ for r up to appr. 1000.

The assumption is that words and phrases mentioned most often are those reflecting important concerns in **every communication**.

Formulaically:
If a word occurs f times and has a rank r in a list of frequencies, then for all words $P(r) = \frac{a}{r^b}$ where a and b are constants and b is close to 1.

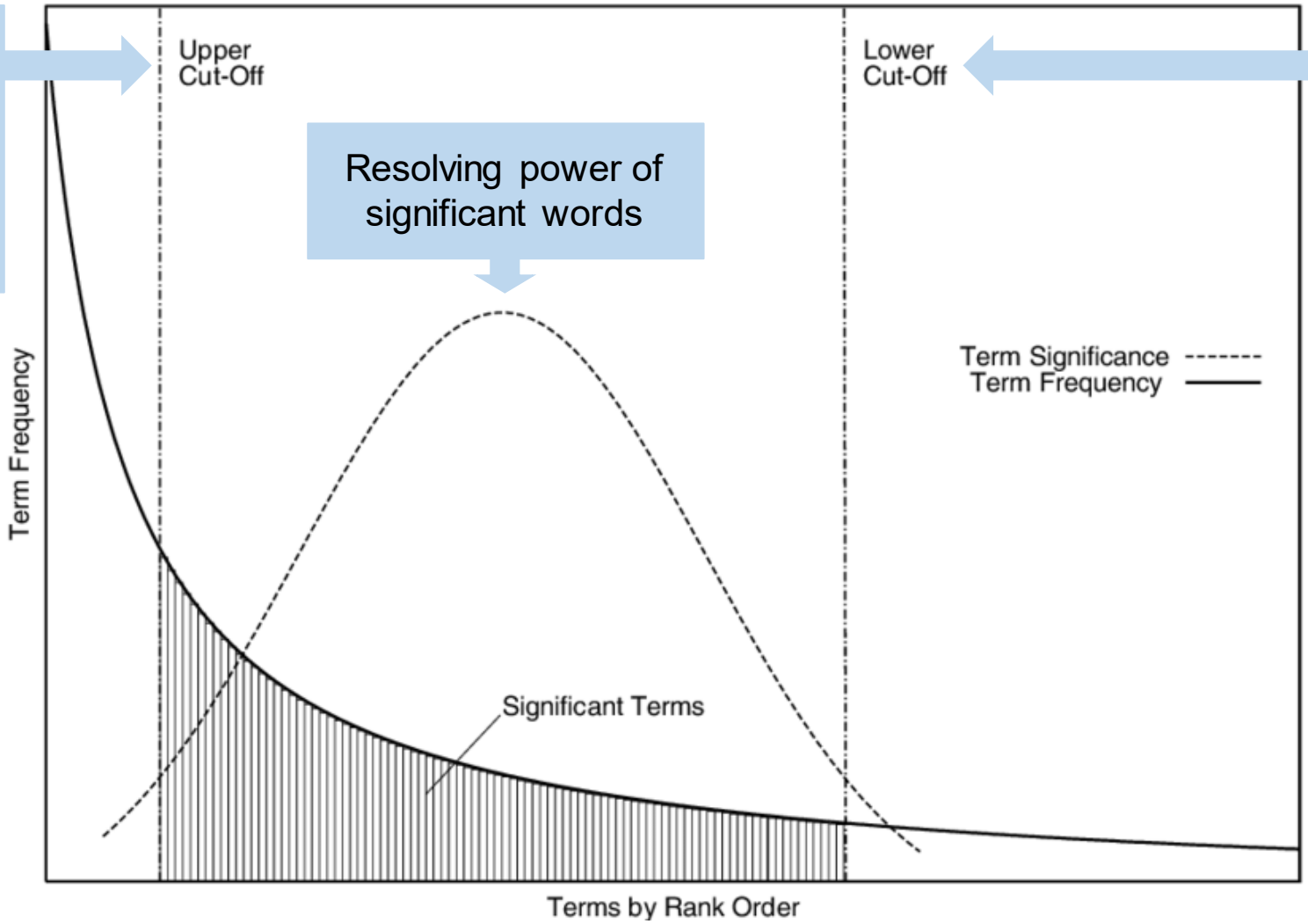


Word frequency and rank in Romeo and Juliet

https://quanteda.io/

Selecting Relevant Words

Remove words that do not bear meaning ("stop words") or do not differentiate between documents



Remove highly infrequent words that do not differentiate between documents to reduce **model complexity**

- The actual cut-offs are determined by trial and error, and often depend on the research context

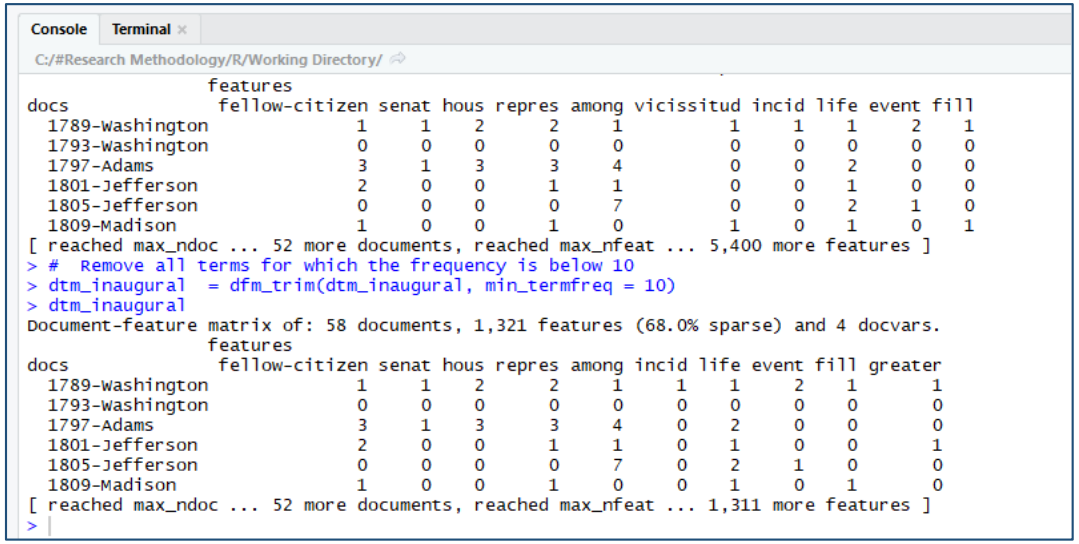
Creating a Complex Document-Feature Matrix

```
# Convert inaugural speeches into a Document-Feature Matrix
dtm_inaugural <- dfm(data_corpus_inaugural,
          tolower=TRUE, remove=stopwords("en"),
          remove_punct=TRUE, stem=TRUE)
dtm_inaugural
```

- This ***dtm*** has 58 documents and 5,410 features (i.e., terms).
- Depending on the type of analysis we might not need this many words, or might actually run into **computational limitations**.
- Luckily, many of these features are **not that informative**.
- The distribution of term frequencies have a **very long tail**, with many words occurring only once or a few times in the corpus.
- For many types of bag-of-words analysis it would not harm to remove these words, and it might actually **improve results**.

We can use the ***dfm_trim*** function to remove columns based on criteria specified in the arguments.

```
#  Remove all terms for which the frequency is below 10
dtm_inaugural  = dfm_trim(dtm_inaugural, min_termfreq = 10)
dtm_inaugural
```

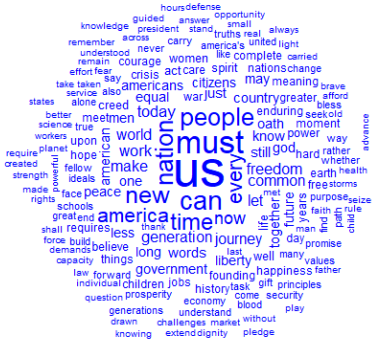


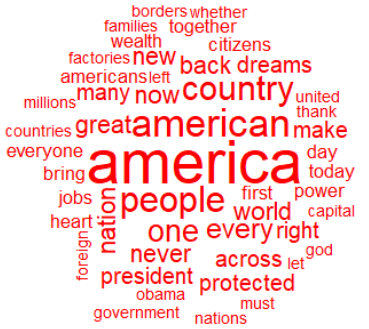
- ?dfm_trim shows all options
- termfreq_type = how min_termfreq is interpreted:
  - "count" sums the frequencies
  - "prop" divides the term frequencies by the total sum
  - "rank" is matched against the inverted ranking of features in terms of overall frequency, so that 1, 2, ... are the highest and second highest frequency features, and so on
  - "quantile" sets the cutoffs according to the quantiles
- min_docfreq = minimum values of a feature's document frequency, below which features will be removed

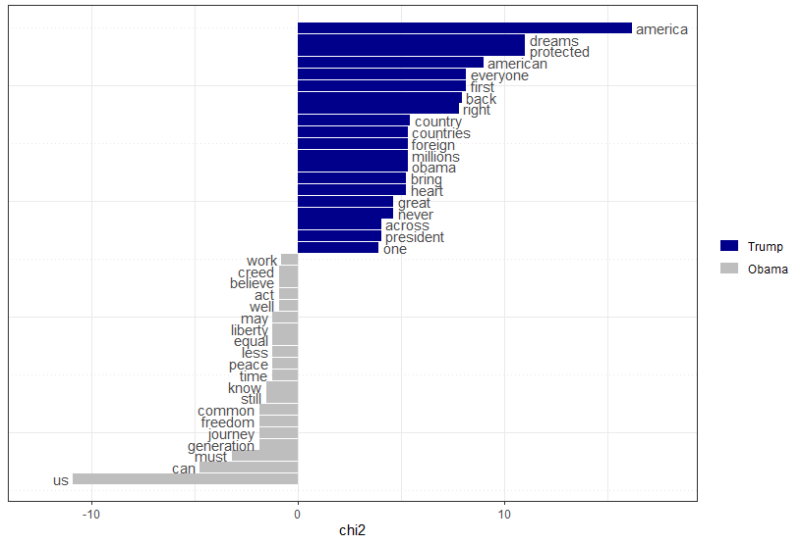
More Descriptive Analyses

corpus_subset() allows to select documents in a corpus based on document-level variables.

```
# Plot features of the inaugural speech as a wordcloud
data_corpus_inaugural %>%
  corpus_subset(President == "Obama") %>%
  dfm(remove = stopwords("en"), remove_punct=TRUE) %>%
  textplot_wordcloud(color = c("blue"))

data_corpus_inaugural %>%
  corpus_subset(President == "Trump") %>%
  dfm(remove = stopwords("en"), remove_punct=TRUE) %>%
  textplot_wordcloud(color = c("red"))

# Plot word keyness (words that occur differentially across the two presidents)
data_corpus_inaugural %>%
  corpus_subset(President %in%
                  c("Obama" , "Trump")) %>%
  dfm(groups = "President", remove = stopwords("en")) %>%
  textstat_keyness(target = "Trump") %>%
  textplot_keyness()
```







➢ %>% helps to make the code more readable: It combines two lines with each other by forwarding the result of an expression into the next expression

https://quanteda.io/

Already Advanced Analyses

We can group documents which share the same value

```
# Convert the last 20 speeches into a dfm according to the "party"
dfmat_pres <- dfm(tail(data_corpus_inaugural, 20), groups = "Party",
  tolower=TRUE, remove=stopwords("en"),
  remove_punct=TRUE, stem=TRUE)

# We can sort this dfm, and inspect it:
dfm_sort(dfmat_pres)
```

```
> dfm_sort(dfmat_pres)
Document-feature matrix of: 2 documents, 3,000 features (28.1% sparse) and 1 docvar.
             features
docs          us nation world peopl america can must new american freedom
  Democratic 130    125    95    92      70  80   87  88       68      44
  Republican 140    128   117   105     105  84   68  66       73      88
[ reached max_nfeat ... 2,990 more features ]
```

We can further calculate the similarities between different documents

```
# Convert the speeches since 1980 into a dfm
ddfmat_inaug_post1980 <- dfm(corpus_subset(data_corpus_inaugural, Year > 1980),
  tolower=TRUE, remove=stopwords("en"),
  remove_punct=TRUE, stem=TRUE)

# Calculate the cosine similarity between presidents
tstat_ot <- textstat_simil(dfmat_inaug_post1980, dfmat_inaug_post1980
            [c("2009-Obama", "2013-Obama", "2017-Trump"), ],
             margin = "documents", method = "cosine")
tstat_ot
```

```
> tstat_ot
textstat_simil object; method = "cosine"
             2009-Obama 2013-Obama 2017-Trump
1981-Reagan       0.623      0.638      0.494
1985-Reagan       0.643      0.663      0.529
1989-Bush         0.625      0.578      0.494
1993-Clinton      0.628      0.627      0.551
1997-Clinton      0.659      0.647      0.556
2001-Bush         0.602      0.619      0.533
2005-Bush         0.527      0.587      0.539
2009-Obama        1.000      0.682      0.519
2013-Obama        0.682      1.000      0.516
2017-Trump        0.519      0.516      1.000
>
```

https://quanteda.io/

Exercise with Your Own Data

- **Start with some simple descriptive analyses**
  1. Plot association between “tokens” and a document variable
  2. Locate some keywords-in-context
  3. Plot the dispersion of keywords
- **Creating a document-feature matrix**
  1. Generate a document-feature matrix without and with preprocessing
  2. Plot features of different documents as word clouds
  3. Calculate the similarities between different documents

Try to use and adapt the code from the examples!

Further Reading on Automated Text Analysis

Aggarwal, C. C., & Zhai, C. (Eds.). (2012). *Mining text data*. Springer Science & Business Media.

Balducci, B., & Marinova, D. (2018). Unstructured data in marketing. *Journal of the Academy of Marketing Science*, 46(4), 557-590.

Benoit, K., Watanabe, K., Wang, H., Nulty, P., Obeng, A., Müller, S., & Matsuo, A. (2018). *quanteda: An R package for the quantitative analysis of textual data*. Journal of Open Source Software, 3(30), 774.

Berger, J., Humphreys, A., Ludwig, S., Moe, W. W., Netzer, O., & Schweidel, D. A. (2020). Uniting the tribes: Using text for marketing insight. *Journal of Marketing*, 84(1), 1-25.

Feldman, R., & Sanger, J. (2007). *The text mining handbook: advanced approaches in analyzing unstructured data.* Cambridge university press.

Hartmann, J., Huppertz, J., Schamp, C., & Heitmann, M. (2019). Comparing automated text classification methods. *International Journal of Research in Marketing*, 36(1), 20-38.

Kwartler, T. (2017). *Text mining in practice with R*. John Wiley & Sons.

Humphreys, A., & Wang, R. J. H. (2018). Automated text analysis for consumer research. *Journal of Consumer Research*, 44(6), 1274-1306.

McKenny, A. F., Aguinis, H., Short, J. C., & Anglin, A. H. (2018). What doesn't get measured does exist: Improving the accuracy of computer-aided text analysis. *Journal of Management*, 44(7), 2909-2933.

Ordenes, F. V., & Zhang, S. (2019). From words to pixels: text and image mining methods for service research. *Journal of Service Management*.

Silge, J., & Robinson, D. (2017). *Text mining with R: A tidy approach*. O'Reilly Media.

Tausczik, Y. R., & Pennebaker, J. W. (2010). The psychological meaning of words: LIWC and computerized text analysis methods. *Journal of language and social psychology*, 29(1), 24-54.

Further Reading on Web Scraping

Articles and Books on Web Scraping

Aydin, O. (2018). *R Web Scraping Quick Start Guide: Techniques and tools to crawl and scrape data from websites*. Packt Publishing Ltd.

Bradley, A., & James, R. J. (2019). Web scraping using R. *Advances in Methods and Practices in Psychological Science*, 2(3), 264-270.

Landers, R. N., Brusso, R. C., Cavanaugh, K. J., & Collmus, A. B. (2016). A primer on theory-driven web scraping: Automatic extraction of big data from the Internet for use in psychological research. *Psychological methods*, 21(4), 475.

Munzert, S., Rubba, C., Meißner, P., & Nyhuis, D. (2014). *Automated data collection with R: A practical guide to web scraping and text mining*. John Wiley & Sons.

Tutorials on Web Scraping with R

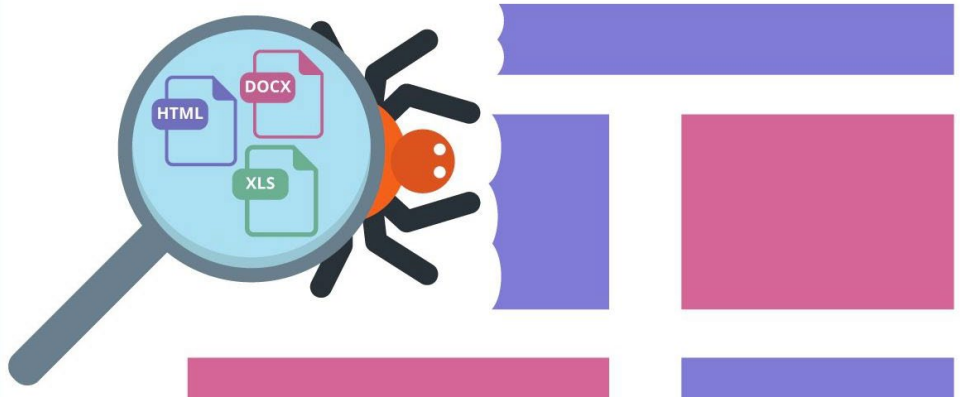
https://www.freecodecamp.org/news/an-introduction-to-web-scraping-using-r-40284110c848/

https://www.datacamp.com/community/tutorials/r-web-scraping-rvest

https://towardsdatascience.com/tidy-web-scraping-in-r-tutorial-and-resources-ac9f72b4fe47

https://www.r-bloggers.com/2019/04/practical-introduction-to-web-scraping-in-r/

https://www.scrapingbee.com/blog/web-scraping-r/



R Packages for Web Scraping

httr: Tools for Working with URLs and HTTP (https://cran.r-project.org/web/packages/httr/)

rvest: Easily Harvest (Scrape) Web Pages (https://cran.r-project.org/web/packages/rvest/)

RSelenium: R Bindings for 'Selenium WebDriver' (https://cran.r-project.org/web/packages/RSelenium/)

Packages for Data Collection in R

twitteR is an R package which provides access to the Twitter API. Most functionality of the API is supported, with a bias towards API calls that are more useful in data analysis as opposed to daily interaction: https://www.rdocumentation.org/packages/twitteR/versions/1.1.9

tuber allows you to Get comments posted on YouTube videos, information on how many times a video has been liked, search for videos with particular content, and much more. You can also scrape captions from videos: https://www.rdocumentation.org/packages/tuber/versions/0.9.9

edgar is a tool for the U.S. SEC EDGAR retrieval and parsing of corporate filings. The EDGAR database automated system collects all the different necessary filings and makes it publicly available. This package facilitates retrieving, storing, searching, and parsing of all the available filings on the EDGAR server: https://cran.r-project.org/web/packages/edgar/index.html

Scraping Amazon Reviews in R: https://martinctc.github.io/blog/vignette-scraping-amazon-reviews-in-r/
Good tutorial but requires a bit more coding...

- Note: As a data collection activity, web-scraping may have legal implications depending on your country. For most countries, as a general rule you can legally web-scrape anything out there that is in the public domain, but it is recommended that you obtain the site owner's permission if you are reporting on the data or using the data for commercial use!

Contact

Prof. Dr. Dennis Herhausen

Associate Professor of Marketing
KEDGE Business School
Domaine de Luminy
Rue Antoine Bourdelle
13009 Marseille, France
dennis.herhausen@kedgebs.com

To connect: LinkedIn