Large Language Models
for the
Economic and Social Sciences

Indira Sen, HWS 2025

A little bit about the instructors

Indira Sen
Junior Faculty, background in Computer Science, research on societal impacts of Large Language Models (LLMs), Human-LLM interaction

A little bit about the instructors

Georg Ahnert
PhD student, background in Social Data Science, research on LLMs & public opinion surveys

Abigail Hayes
PhD student, background in Maths and Data Science, research on social networks

Chair of Data Science for the Economic and Social Sciences (DESS)

A little bit about you

About this course

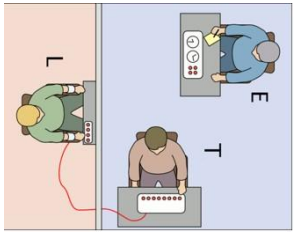
Materials: on ILIAS or Github*

*to be explained in the exercise

The Social Sciences and how we do them

- Social Science is the scientific study of human society and social relationships
- Of human behavior and attitudes, in groups or as individuals
- Traditionally, many different types of data

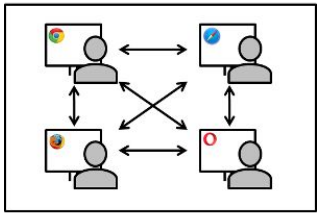
Surveys



Experiments

Observation

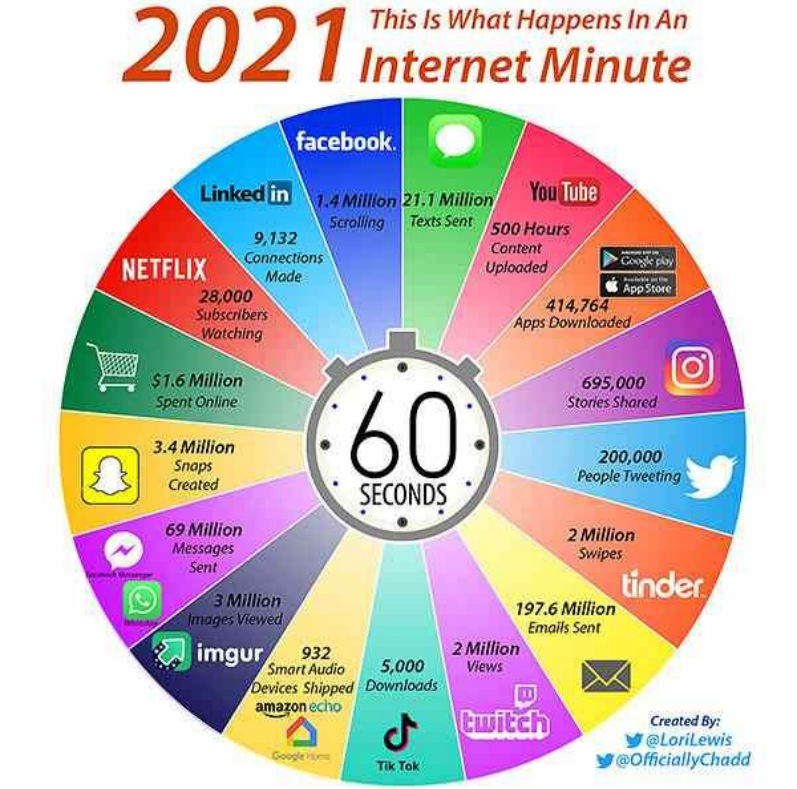
Online/digital surveys



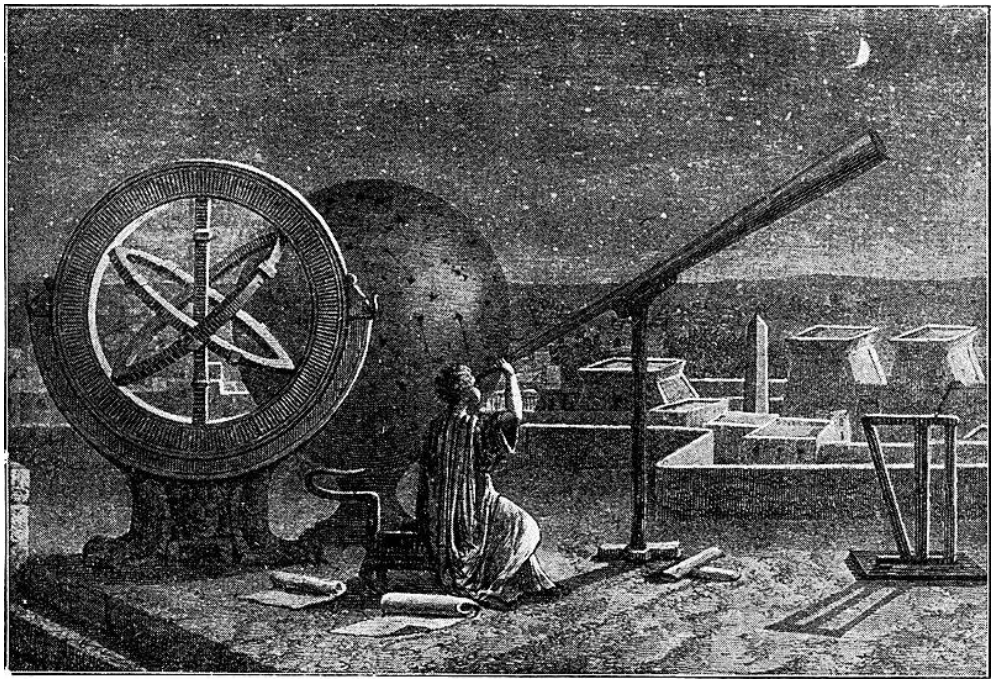
Online/digital experiments

Online/digital observation

Digital Traces: So much data!
(...about people's behavior and attitudes)



A Computational Turn: A Measurement Revolution?



SOCIAL SCIENCE

Computational Social Science

A field is emerging that leverages the capacity to collect and analyze data at a scale that may reveal patterns of individual and group behaviors.

David Lazer,[1] Alex Pentland,[2] Lada Adamic,[3] Sinan Aral,[2,4] Albert-László Barabási,[5] Devon Brewer,[6] Nicholas Christakis,[1] Noshir Contractor,[7] James Fowler,[8] Myron Gutmann,[3] Tony Jebara,[9] Gary King,[1] Michael Macy,[10] Deb Roy,[2] Marshall Van Alstyne[2,11]

Downloaded

We live life in the network. We check our e-mails regularly, make mobile phone calls from almost any location, swipe transit cards to use public transportation, and make purchases with credit cards. Our movements in public places may be captured by video cameras, and our medical records stored as digital files. We may post blog entries accessible to anyone, or maintain friendships through online social networks. Each of these transactions leaves digital traces that can be compiled into comprehensive pictures of both individual and group behavior, with the potential to transform our understanding of our lives, organizations, and societies.

The capacity to collect and analyze massive amounts of data has transformed such fields as biology and physics. But the emergence of a data-driven "computational social science" has been much slower. Leading journals in economics, sociology, and political science show little evidence of this field. But computational social science is occurring—in Internet companies such as Google and Yahoo, and in govern-

[1]Harvard University, Cambridge, MA, USA. [2]Massachusetts Institute of Technology, Cambridge, MA, USA. [3]University of Michigan, Ann Arbor, MI, USA. [4]New York University, New York, NY, USA. [5]Northeastern University, Boston, MA, USA. [6]Interdisciplinary Scientific Research, Seattle, WA, USA. [7]Northwestern University, Evanston, IL, USA. [8]University of California–San Diego, La Jolla, CA, USA. [9]Columbia University, New York, NY, USA [10]Cornell University, Ithaca, NY, USA. [11]Boston University, Boston, MA, USA. E-mail: david_lazer@harvard.edu. Complete affiliations are listed in the supporting online material.

NATURE|Vol 445|1 February 2007

ESSAY

Putting the pieces together

A twenty-first century science

If handled appropriately, data about Internet-based communication and interactivity could revolutionize our understanding of collective human behaviour.

Duncan J. Watts

Few would deny that many of the major problems currently facing humanity are social and economic in nature. From the apparent wave of religious fundamentalism sweeping the Islamic world (and parts of the Western world), to collective economic security, global warming and the great epidemics of our times, powerful yet mysterious social forces come into play.

But few readers of *Nature* would consider social science to be the science of the twenty-first century. Although economics, sociology, political science and anthropology have produced a plethora of findings regarding human social behaviour, they self-reports from participants, which suffer from cognitive biases, errors of perception and framing ambiguities.

The striking proliferation over the past decade of Internet-based communication and interactivity, however, is beginning to lift these constraints. For the first time, we can begin to observe the real-time interactions of millions of people at a resolution that is sensitive to effects at the level of the individual. Meanwhile, ever-faster computers permit us to simulate large networks of social interactions. The result has been tremendous interest in social networks: thousands of papers and a growing number of books have been publish
less than a decade, leading some to h

framework of collective social dynamics. People do not just interact: their interactions have consequences for the choices they, and others, make.

Studies that combine all these features are currently beyond the state of the art, but two of my group's recent projects indicate tentative progress. The first used the anonymized e-mail logs of a university community of around 40,000 people to track daily network evolution over a year as a function of existing network structure, shared activities (such as classes) and individual attributes. Dynamic data of this type may shed light on the relative roles of struc-

Data from the blogosphere. Shown is a link structure within a community of political blogs (from 2004), where red nodes indicate conservative blogs, and blue liberal. Orange links go from liberal to conservative, and purple ones from conservative to liberal. The size of each blog reflects the number of other blogs that link to it. [Reproduced from (*8*) with permission from the Association for Computing Machinery]

Reference:
Computational social science
A Twenty-first Century Science

SOCIAL SCIENCE

Computational Social Science

David Lazer,[1] Alex Pentland,[2] Lada Adamic,[3] Sinan Aral,[2,4] Albert-László Barabási,[5] Devon Brewer,[6] Nicholas Christakis,[1] Noshir Contractor,[7] James Fowler,[8] Myron Gutmann,[3] Tony Jebara,[9] Gary King,[1] Michael Macy,[10] Deb Roy,[2] Marshall Van Alstyne[2,11]

A field is emerging that leverages the capacity to collect and analyze data at a scale that may reveal patterns of individual and group behaviors.

Downloaded

We live life in the network. We check our e-mails regularly, make mobile phone calls from almost any location, swipe transit cards to use public transportation, and make purchases with credit cards. Our movements in public places may be captured by video cameras, and our medical records stored as digital files. We may post blog entries accessible to anyone, or maintain friendships through online social networks. Each of these transactions leaves digital traces that can be compiled into comprehensive pictures of both individual and group behavior, with the potential to transform our understanding of our lives, organizations, and societies.

The capacity to collect and analyze massive amounts of data has transformed such fields as biology and physics. But the emergence of a data-driven "computational social science" has been much slower. Leading journals in economics, sociology, and political science show little evidence of this field. But computational social science is occurring—in Internet companies such as Google and Yahoo, and in govern-

[1]Harvard University, Cambridge, MA, USA. [2]Massachusetts Institute of Technology, Cambridge, MA, USA. [3]University of Michigan, Ann Arbor, MI, USA. [4]New York University, New York, NY, USA. [5]Northeastern University, Boston, MA, USA. [6]Interdisciplinary Scientific Research, Seattle, WA, USA. [7]Northwestern University, Evanston, IL, USA. [8]University of California–San Diego, La Jolla, CA, USA. [9]Columbia University, New York, NY, USA [10]Cornell University, Ithaca, NY, USA. [11]Boston University, Boston, MA, USA. E-mail: david_lazer@harvard.edu. Complete affiliations are listed in the supporting online material.

NATURE|Vol 445|1 February 2007

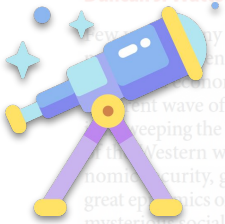
ESSAY

Putting the pieces together

A twenty-first century science

If handled appropriately, data about Internet-based communication and interactivity could revolutionize our understanding of collective human behaviour.

Duncan J. Watts



Digital Traces + computational methods are the ***Telescope*** for the Social Sciences

self-reports from participants, which suffer from cognitive biases, errors of perception and framing ambiguities

framework of collective social dynamics. People do not just interact: their interactions have consequences for the choices they, and others, make.

Studies that combine all these features are currently beyond the state of the art, but two of my group's recent projects indicate tentative progress. The first used the anonymized e-mail logs of a university community of around 40,000 people to track daily network evolution over a year as a function of existing network structure, shared activities (such as classes) and individual attributes. Dynamic data of this type may shed light on the relative roles of struc-

But few readers of *Nature* would consider social science to be the science of the twenty-first century. Although economics, sociology, political science and anthropology have produced a plethora of findings regarding human social behaviour, they

actions of millions of people at a resolution that is sensitive to effects at the level of the individual. Meanwhile, ever-faster computers permit us to simulate large networks of social interactions. The result

has been tremendous interest in social networks: thousands of papers and a growing number of books have been publish

less than a decade, leading some to h

Data from the blogosphere. Shown is a link structure within a community of political blogs (from 2004), where red nodes indicate conservative blogs, and blue liberal. Orange links go from liberal to conservative, and purple ones from conservative to liberal. The size of each blog reflects the number of other blogs that link to it. [Reproduced from (*8*) with permission from the Association for Computing Machinery]

Reference:
Computational social science
A Twenty-first Century Science

Computational Social Science

SOCIAL SCIENCE

Computational Social Science

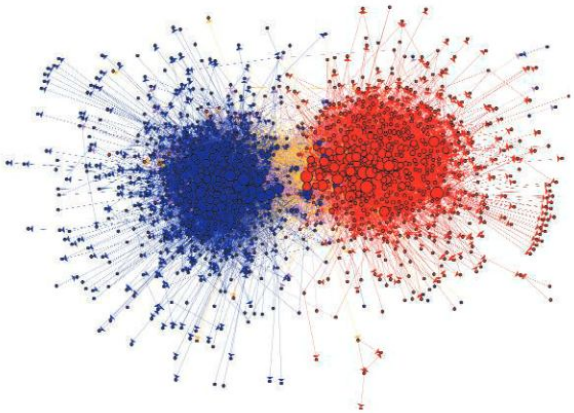
A field is emerging that leverages the capacity to collect and analyze data at a scale that may reveal patterns of individual and group behaviors.

David Lazer,[1] Alex Pentland,[2] Lada Adamic,[3] Sinan Aral,[2,4] Albert-László Barabási,[5] Devon Brewer,[6] Nicholas Christakis,[1] Noshir Contractor,[7] James Fowler,[8] Myron Gutmann,[3] Tony Jebara,[9] Gary King,[1] Michael Macy,[10] Deb Roy,[2] Marshall Van Alstyne[2,11]

We live life in the network. We check our e-mails regularly, make mobile phone calls from almost any location, swipe transit cards to use public transportation, and make purchases with credit cards. Our movements in public places may be captured by video cameras, and our medical records stored as digital files. We may post blog entries accessible to anyone, or maintain friendships through online social networks. Each of these transactions leaves digital traces that can be compiled into comprehensive pictures of both individual and group behavior, with the potential to transform our understanding of our lives, organizations, and societies.

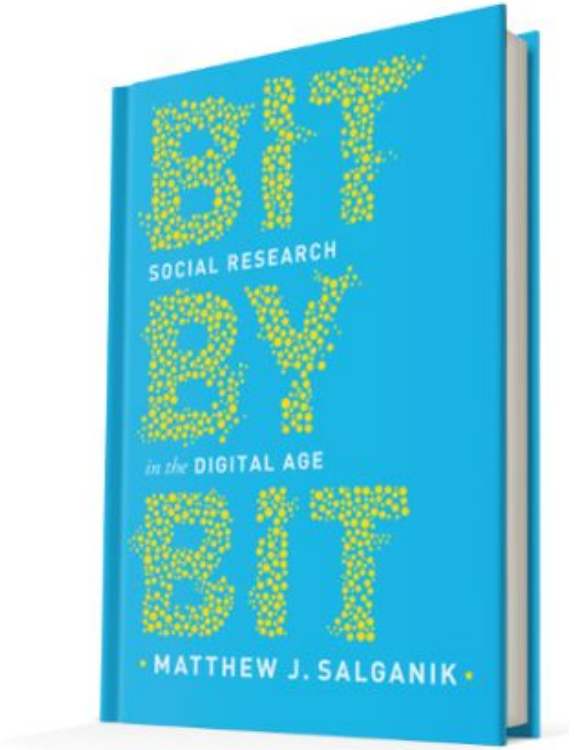
The capacity to collect and analyze massive amounts of data has transformed such fields as biology and physics. But the emergence of a data-driven "computational social science" has been much slower. Leading journals in economics, sociology, and political science show little evidence of this field. But computational social science is occurring—in Internet companies such as Google and Yahoo, and in government agencies such as the U.S. National Security Agency. Computational social science could become the exclusive domain of private companies and government agencies. Alternatively, there might emerge a privileged set of academic researchers presiding over private data from which they produce papers that cannot be critiqued or replicated. Neither scenario will serve the long-term public interest of accumulating, verifying, and disseminating knowledge.

What value might a computational social science—based in an open academic environment—offer society, by enhancing understanding of individuals and collectives? What are the



[1]Harvard University, Cambridge, MA, USA. [2]Massachusetts Institute of Technology, Cambridge, MA, USA. [3]University of Michigan, Ann Arbor, MI, USA. [4]New York University, New York, NY, USA. [5]Northeastern University, Boston, MA, USA. [6]Interdisciplinary Scientific Research, Seattle, WA, USA. [7]Northwestern University, Evanston, IL, USA.

Lazer, David, et al. "Computational social science." *Science* 323.5915 (2009): 721-723.



Salganik, Matthew J. *Bit by bit: Social research in the digital age.* Princeton University Press, 2019.

What is Social Data Science?

The aim of Social Data Science is: **Understanding** of **Social Phenomena** using **Data Science.**

- Understanding: Not just predicting, we want to be able to generalize and combine knowledge, and even to motivate interventions or policies.

- Social Phenomena: ABCs of humans and society — Attitudes, Behaviors, and Characteristics

- Data Science: Using statistical methods, machine learning, Large Language Models!

What are LLMs?

"Computational agents that can interact conversationally with people using natural language" —Jurafsky and Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition

What do LLMs have to do with (social) science?

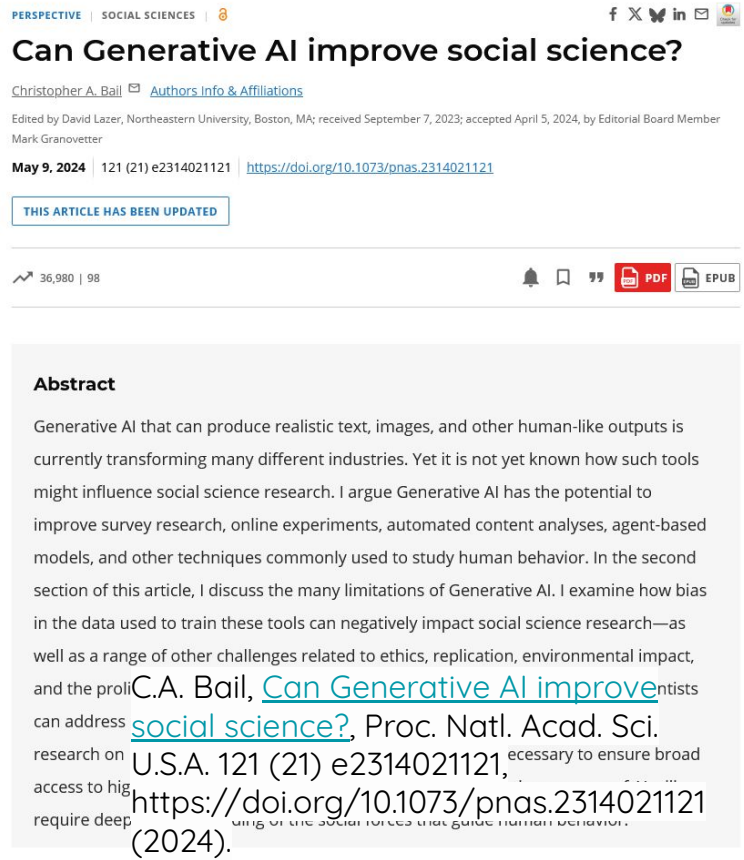
Let's try a small exercise. Open up ChatGPT and ask it the following questions:

1. What is the topic of the following news headline: "The German Chancellor said that the social state can no longer be funded". Answer only with 'politics' or 'not politics'

2. What does a 35 year-old man from Düsseldorf think about the German chancellor in 2025?

What do LLMs have to do with (social) science?

- LLMs can produce “human-like” outputs, mainly free-text
- If** LLMs can behave like people, we can use them to understand behaviors and characteristics of people
    - Use them to mimic human behavior in social settings, like the workplace
    - Or social media!



PERSPECTIVE | SOCIAL SCIENCES |

Can Generative AI improve social science?

Christopher A. Bail | Authors Info & Affiliations

Edited by David Lazer, Northeastern University, Boston, MA; received September 7, 2023; accepted April 5, 2024, by Editorial Board Member Mark Granovetter

May 9, 2024 | 121 (21) e2314021121 | https://doi.org/10.1073/pnas.2314021121

THIS ARTICLE HAS BEEN UPDATED

36,980 | 98 PDF EPUB

Abstract

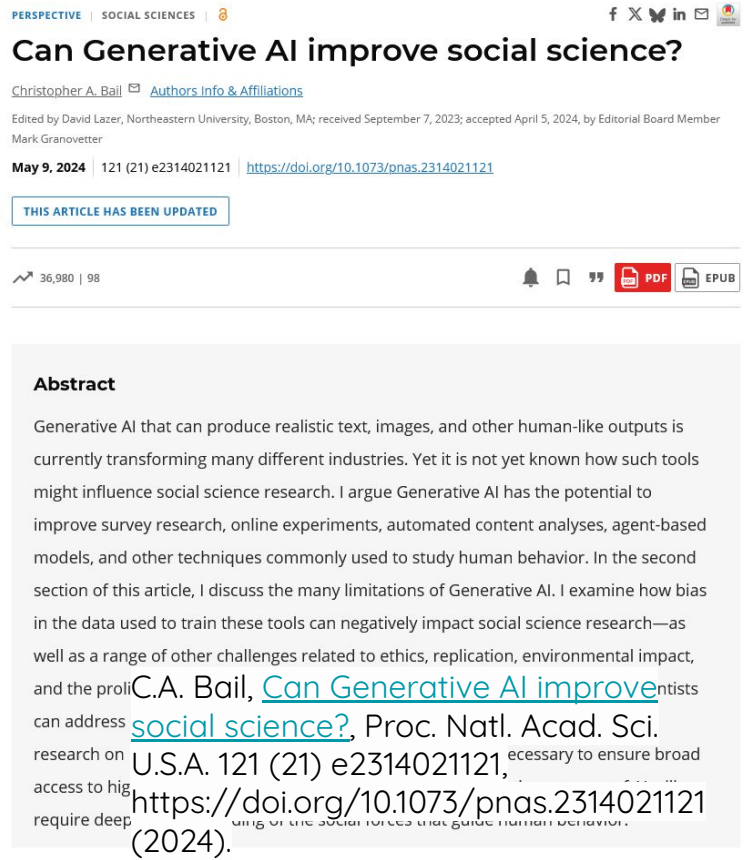
Generative AI that can produce realistic text, images, and other human-like outputs is currently transforming many different industries. Yet it is not yet known how such tools might influence social science research. I argue Generative AI has the potential to improve survey research, online experiments, automated content analyses, agent-based models, and other techniques commonly used to study human behavior. In the second section of this article, I discuss the many limitations of Generative AI. I examine how bias in the data used to train these tools can negatively impact social science research—as well as a range of other challenges related to ethics, replication, environmental impact, and the proli… scientists can address … ecessary to ensure broad access to hig… research on … require deep … ding of the social forces that guide human behavior.

C.A. Bail, Can Generative AI improve social science?, Proc. Natl. Acad. Sci. U.S.A. 121 (21) e2314021121, https://doi.org/10.1073/pnas.2314021121 (2024).

What do LLMs have to do with (social) science?

- LLMs can produce “human-like” outputs, mainly free-text
- If** LLMs can behave like people, we can use them to understand behaviors and characteristics of people
    - Use them to mimic human behavior in social settings, like the workplace
    - Or social media!

** this is a big IF and something that we cannot take for granted



PERSPECTIVE | SOCIAL SCIENCES

Can Generative AI improve social science?

Christopher A. Bail | Authors Info & Affiliations

Edited by David Lazer, Northeastern University, Boston, MA; received September 7, 2023; accepted April 5, 2024, by Editorial Board Member Mark Granovetter

May 9, 2024 | 121 (21) e2314021121 | https://doi.org/10.1073/pnas.2314021121

THIS ARTICLE HAS BEEN UPDATED

36,980 | 98

PDF EPUB

Abstract

Generative AI that can produce realistic text, images, and other human-like outputs is currently transforming many different industries. Yet it is not yet known how such tools might influence social science research. I argue Generative AI has the potential to improve survey research, online experiments, automated content analyses, agent-based models, and other techniques commonly used to study human behavior. In the second section of this article, I discuss the many limitations of Generative AI. I examine how bias in the data used to train these tools can negatively impact social science research—as well as a range of other challenges related to ethics, replication, environmental impact, and the proli ntists can address research on ecessary to ensure broad access to hig require deep

C.A. Bail, Can Generative AI improve social science?, Proc. Natl. Acad. Sci. U.S.A. 121 (21) e2314021121, https://doi.org/10.1073/pnas.2314021121 (2024).

Here are some questions we will explore in this course

- How are LLMs developed?
- Can we replace or substitute humans as social science subjects or facilitators with LLMs?
  - In surveys
  - Behavioral experiments
  - Content analysis
  - Social simulations
- How do we evaluate LLMs in social applications, e.g., when asking them for moral advice?

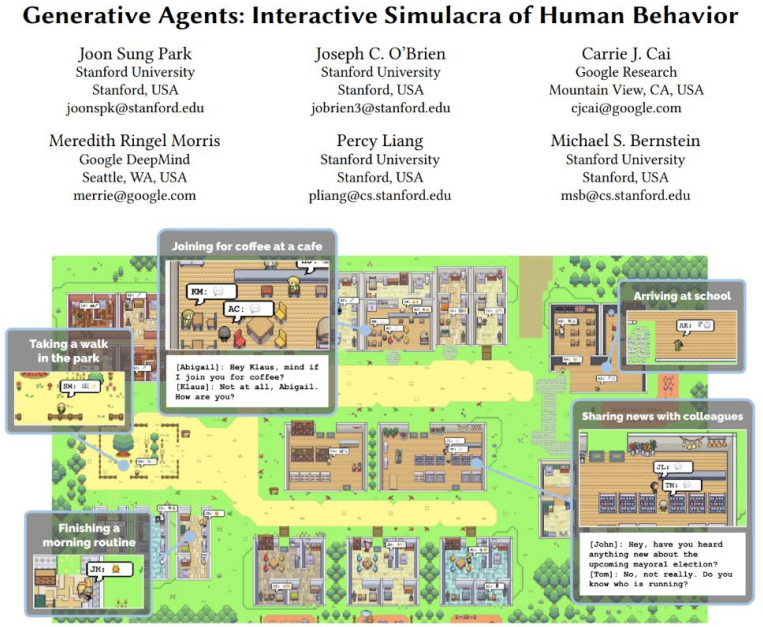


Figure 1: Generative agents are believable simulacra of human behavior for interactive applications. In this work, we demonstrate

Park, Joon Sung, et al. "Generative agents: Interactive simulacra of human behavior." *Proceedings of the 36th annual acm symposium on user interface software and technology*. 2023.

Course outcomes: After this course, you would have:

- Learned LLM fundamentals and their applications:
  - Foundations of Large Language Models
  - The recipe for building LLMs
  - Hands-on practice with prompting and fine-tuning LLMs

- Learned about Computational Social Science:
  - Main methods used in CSS
  - Applications of LLMs to CSS research questions and applications
  - CSS approaches to studying LLMs

- Gotten experience with a hands-on CSS project that uses LLMs

You should take this course, if you have:

- Strong knowledge of and experience in programming, ideally Python
  - All code provided from instructors will be in Python
  - You are expected to do the course project in Python

- An interest in scientific research
  - Ideally, you have read a scientific research article before
  - If not, please check: S.Keshav's "How to Read a Paper"

This course is based on recent, state-of-the-art research in Natural Language Processing and Computational Social Science

You are expected to go over the readings assigned for each lecture and participate in discussions

Course Logistics

Lecture Topics/Weeks

1. Introduction + Demystifying LLMs 1
2. Demystifying LLMs 2
3. Demystifying LLMs 3
4. Interacting with LLMs
5. Infrastructure behind LLMs
6. Content Analysis with LLMs + project pitches
7. AI-augmented Surveys
8. Social Media Simulations with LLMs
9. Midway project presentation
10. Machine Behavior
11. Ethics and Environmental Impact
12. AI Safety and Alignment
13. LLM Audits
14. Summary and Outlook

We have weekly lectures and exercises

- Lectures
  - Monday 1:45-3:15 PM
  - Starting from week 4: 10-20 minutes of group discussion at the end

- Exercises
  - Recap and **hands-on application** of what we saw in the lecture
  - Weekly exercise questions **(not graded)**
  - Solutions are presented in the exercise or the week after
  - Thursday 1:45-3:15 PM
    OR
  - Friday 12:00-1:30 PM

Grading

- Group Project (80%)
- Class Participation (20%)
  - Every lecture session will have 1-2 items announced ahead: mainly papers but could also be blog posts, videos, etc
  - You are expected to read or watch these before the lecture
  - 10-15 mins before the end of every lecture, we will have a short discussion
    - What you learned
    - What you agree of disagree with
    - How you can improve what you read
    - ....

Group Project

- **Equivalent to a scientific short paper: 4-5 pages, double column**
- You have to work in teams of 2-5; team size suggestion will be finalized based on how many people register for the course
- If you cannot find project partners: use the forum on ILIAS
  - Worst case, you'll be merged with another group
- We have some suggestions for projects ideas, but you are encouraged to come up with your own and be creative!

More about the Projects: Timeline and Dates

- Start forming groups: week 2
- Project pitches: 6.10.2025 [5%]
- Midway Presentation: 27.10.2025 (during the lecture) [10%]
- Final presentation; 30.11.2025 (during one or both exercise sessions) [15%]
- Final reports due: 12.12.2025 [50%]

More about the Projects: Timeline and Dates

- Start forming groups: week 2
- Project pitches: 6.10.2025 [5%]
- Midway Presentation: 27.10.2025 (during the lecture) [10%]
- Final presentation; 30.11.2025 (during one or both exercise sessions) [15%]
- Final reports due: 12.12.2025 [50%]

You cannot drop the course after the project pitches, so be sure to deregister earlier if you don't want to participate in the course

More about the Projects: Timeline and Dates

- Start forming groups: week 2
- Project pitches: 6.10.2025 [5%]
- Midway Presentation: 27.10.2025 (during the lecture) [10%]
- Final presentation; 30.11.2025 (during one or both exercise sessions) [15%]
- Final reports due: 12.12.2025 [50%]

You cannot drop the course after the project pitches, so be sure to deregister earlier if you don't want to participate in the course

You also cannot change groups after the project pitches

More about the Projects: Presentations and Report

- Project pitches: 6.10.2025 [5%] —> 5 mins per teams
- Midway Presentation: 27.10.2025 (during the lecture) [10%] —> 10 mins
- Final presentation; 30.11.2025 (during one or both exercise sessions) [15%] —> 20 mins
- Final reports due: 12.12.2025 [50%]
    - All reports should include: introduction, related work, experiments, results, conclusion
    - Team member contributions: all members of a group need not get the same grades
    - Ideally: also include your code

Questions?

Let’s demystify Large Language Models

What are LLMs?

"Computational agents that can interact conversationally with people using natural language" —Jurafsky and Martin, "Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition

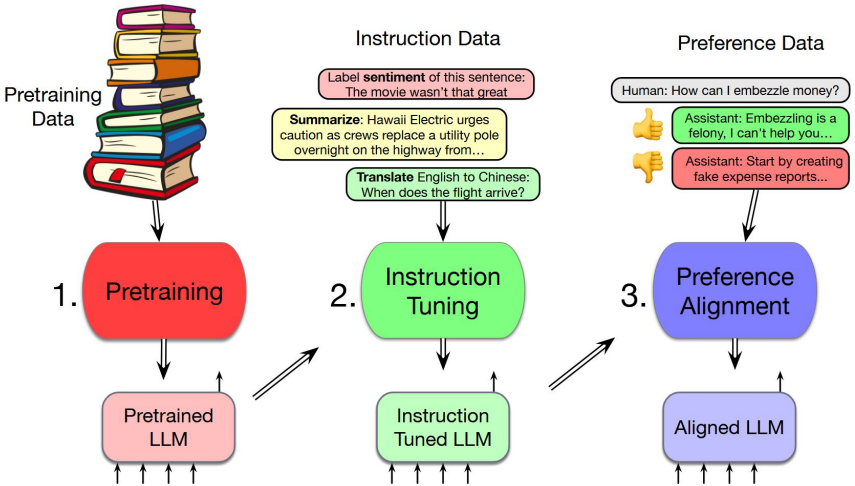
How can we build LLMs?

Simple answer

- We take a LOT of data encoding social, cultural, and of course linguistic traces
- Train NLP models that can generate content on this data
- We further train the model to follow and respond to instructions
- We further modify the model so that it doesn't have unwanted behavior

How can we build LLMs?

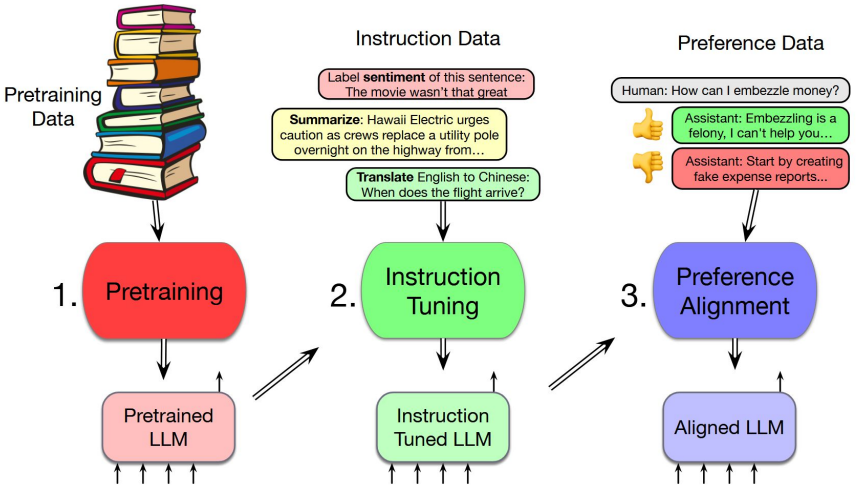
Somewhat simplified answer



Jurafsky and Martin, 2025

How can we build LLMs?

Somewhat simplified answer



Jurafsky and Martin, 2025

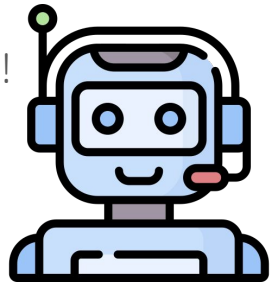
Vastly Simplified answer

- We take a LOT of data encoding social, cultural, and of course linguistic traces
- Train NLP models that can generate content on this data
- We further train the model to follow and respond to instructions
- We further modify the model so that it doesn't have unwanted behavior

Input to and Output from LLMs: Words and Tokens

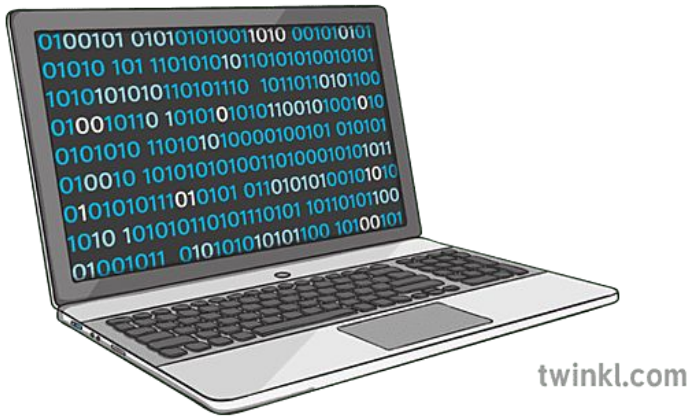
Should I take the course IS 617?

Absolutely!



Talking to computers: Text Representations

- Computers don't understand natural language like humans do, but they know binary
- We need to convert our text into a format that computers understand
- Therefore, we turn words to numbers —> Obtain a **representation**



Input to and Output from LLMs: Words and Tokens

Should I take the course IS 617?

[“Should”, “I”, “take”, “the”, “course”, “IS”, “617”]

Input to and Output from LLMs: Words and Tokens

Should I take the course IS 617?

What's special in the course?

["Should", "I", "take", "the", "course", "IS", "617", "What", "'s", "special", "in"]

11 Types: all words are counted ***once***

Input to and Output from LLMs: Words and Tokens

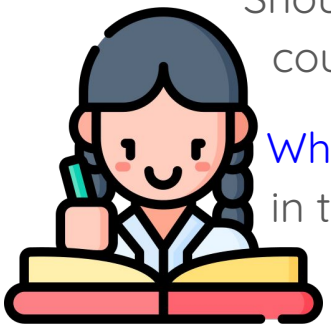
Should I take the course IS 617?

What's special in the course?

["Should", "I", "take", "the", "course", "IS", "617", "What", "'s", "special", "in", "the", "course"]

13 instances: count every occurrence

Input to and Output from LLMs: Words and Tokens



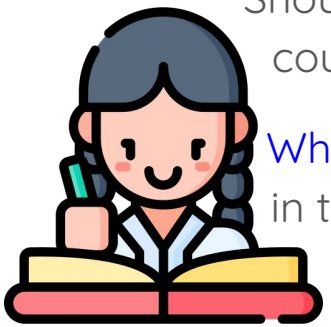
Should I take the course IS 617?

What's special in the course?

["Should", "I", "take", "the", "course", "IS", "617", "What", "'s", "special", "in"]

We just did **tokenization**! The process of breaking down piece of text into smaller pieces

Input to and Output from LLMs: Words and Tokens



Should I take the course IS 617?

What's special in the course?

["Should", "I", "take", "the", "course", "IS", "617", "What", "'s", "special", "in"]

We just did **tokenization**! The process of breaking down piece of text into smaller pieces

This might seem trivial, but often not clear where to split a word.
Some languages don't even have spaces between words, e.g., Chinese

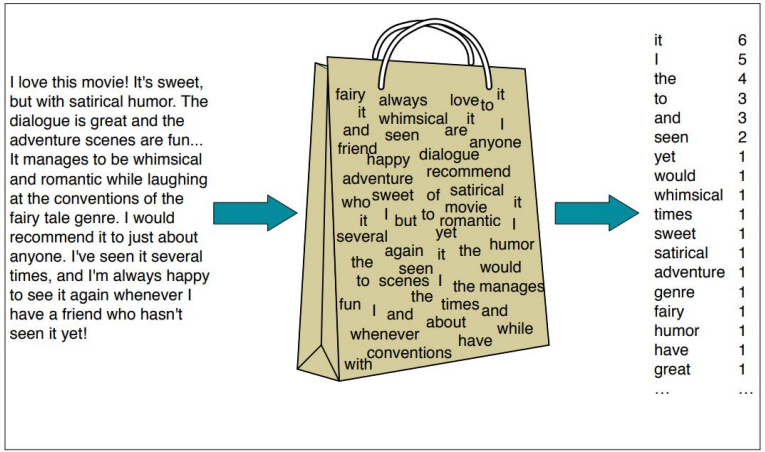
Try it yourself:
https://tiktokenizer.vercel.app/

So, now how do I feed in a LOT of data to a model, i.e., make a representation of my data?

- Take all the documents you have and turn them into lists of tokens
- Now make sure that all documents are comparable and have the same dimensions to get a list of lists
- For each document either
  - Count the number of times a token occurs
  - Count if a token occurs

Bag-of-words Representation

- The most basic: Bag-of-words (BoW)
- Bag of Words assumption: word order does not matter

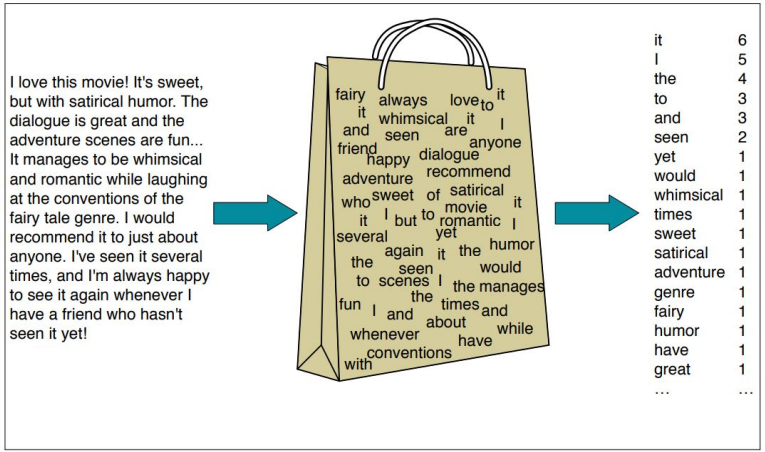


Jurafsky and Martin, Speech and Language Processing

Bag-of-words Representation

- The most basic: Bag-of-words (BoW)
- Bag of Words assumption: word order does not matter

When can this assumption be problematic?



Jurafsky and Martin, Speech and Language Processing

Document D1	*The child makes the dog happy* the: 2, dog: 1, makes: 1, child: 1, happy: 1
Document D2	*The dog makes the child happy* the: 2, child: 1, makes: 1, dog: 1, happy: 1

	child	dog	happy	makes	the	**BoW Vector representations**
D1	1	1	1	1	2	**[1,1,1,1,2]**
D2	1	1	1	1	2	**[1,1,1,1,2]**

Document D1	*The child makes the dog happy* the: 2, dog: 1, makes: 1, child: 1, happy: 1
Document D2	*The dog makes the child happy* the: 2, child: 1, makes: 1, dog: 1, happy: 1

a list of numbers: Vector

	child	dog	happy	makes	the	BoW Vector representations
D1	1	1	1	1	2	[1,1,1,1,2]
D2	1	1	1	1	2	[1,1,1,1,2]

Document D1	*The child makes the dog happy* the: 2, dog: 1, makes: 1, child: 1, happy: 1
Document D2	*The dog makes the child happy* the: 2, child: 1, makes: 1, dog: 1, happy: 1

a list of numbers: Vector

	child	dog	happy	makes	the	BoW Vector representations
D1	1	1	1	1	2	[1,1,1,1,2]
D2	1	1	1	1	2	[1,1,1,1,2]

a list of lists: Matrix

Let’s say we have 4 social media posts in our data

- Documents
  - “I love cheese”
  - “Dogs hate cheese”
  - “I hate chocolate”
  - “I love dogs”
- Convert these to tokens
- Get the total vocabulary
- For each document, create a vector based on whether a token occurs or not

Let’s say we have 4 social media posts in our data

- Documents
  - “I love cheese” —> [“i”, “love”, “cheese”]
  - “Dogs hate cheese” —> [“dogs”, “hate”, “cheese”]
  - “I hate chocolate” —> [“i”, “hate”, “chocolate”]
  - “I love dogs” —> [“i”, “love”, “dogs”]
- **Convert these to tokens**
- Get the total vocabulary
- For each document, create a vector based on whether a token occurs or not

Let’s say we have 4 social media posts in our data

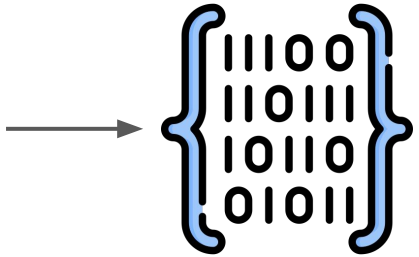
- Documents
  - “I love cheese” —> [“i”, “love”, “cheese”]
  - “Dogs hate cheese” —> [“dogs”, “hate”, “cheese”]
  - “I hate chocolate” —> [“i”, “hate”, “chocolate”]
  - “I love dogs” —> [“i”, “love”, “dogs”]
- Convert these to tokens
- **Get the total vocabulary** = [“i”, “love”, “cheese”, “dogs”, “hate”, “chocolate”]
  (for vocabulary always count types, not instances!)
- For each document, create a vector based on whether a token occurs or not

Let’s say we have 4 social media posts in our data

- Documents
  - “I love cheese” —> [“i”, “love”, “cheese”] —> [1, 1, 1, 0, 0, 0]
  - “Dogs hate cheese” —> [“dogs”, “hate”, “cheese”] —> [0, 0, 1, 1, 1, 0]
  - “I hate chocolate” —> [“i”, “hate”, “chocolate”] —> [1, 0, 0, 0, 1, 1]
  - “I love dogs” —> [“i”, “love”, “dogs”] —> [1, 1, 0, 1, 0, 0]
- Convert these to tokens
- Get the total vocabulary = [“i”, “love”, “cheese”, “dogs”, “hate”, “chocolate”]
  (for vocabulary always count types, not instances!)
- **For each document, create a vector based on whether a token occurs or not**

Let’s say we have 4 social media posts in our data

- Documents
  - “I love cheese” —> [“i”, “love”, “cheese”] —> [1, 1, 1, 0, 0, 0]
  - “Dogs hate cheese” —> [“dogs”, “hate”, “cheese”] —> [0, 0, 1, 1, 1, 0]
  - “I hate chocolate” —> [“i”, “hate”, “chocolate”] —> [1, 0, 0, 0, 1, 1]
  - “I love dogs” —> [“i”, “love”, “dogs”] —> [1, 1, 0, 1, 0, 0]



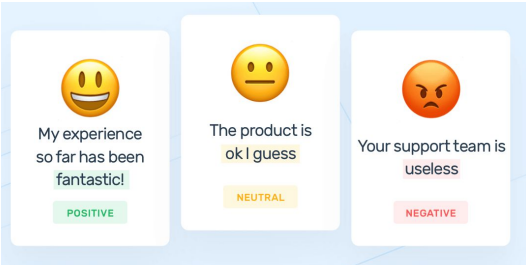
- Convert these to tokens

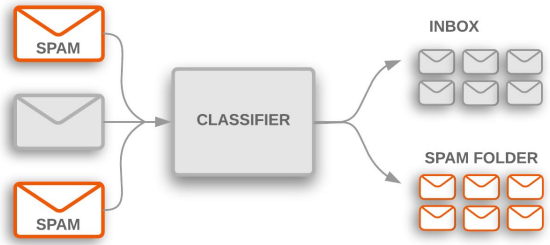
- Get the total vocabulary = [“i”, “love”, “cheese”, “dogs”, “hate”, “chocolate”]
  (for vocabulary always count types, not instances!)

- **For each document, create a vector based on whether a token occurs or not**

Doing useful things with representations: classification

Text Classification: One of the most basic and popular NLP tasks



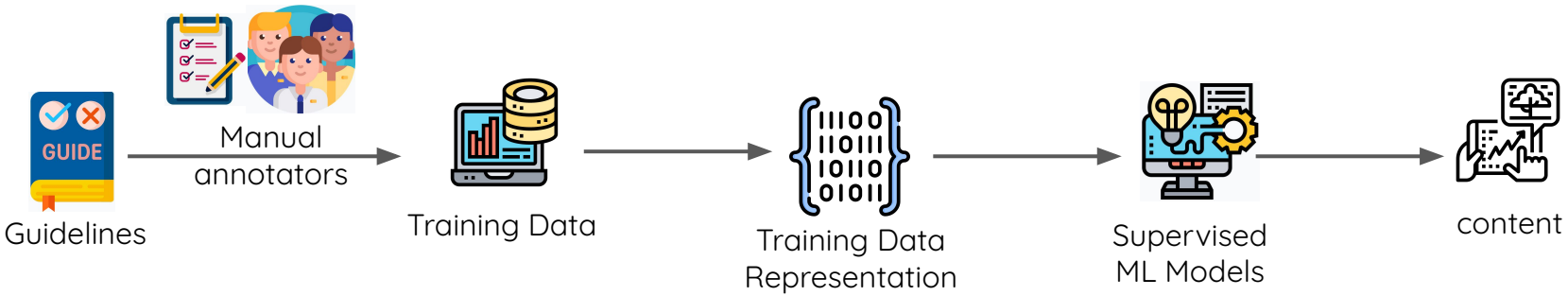


Check out https://lena-voita.github.io/nlp_course/text_classification.html for an overview of different text classification datasets

How to do supervised classification

We can use the representations we created (e.g., Bag-of-words) to further use for classification.

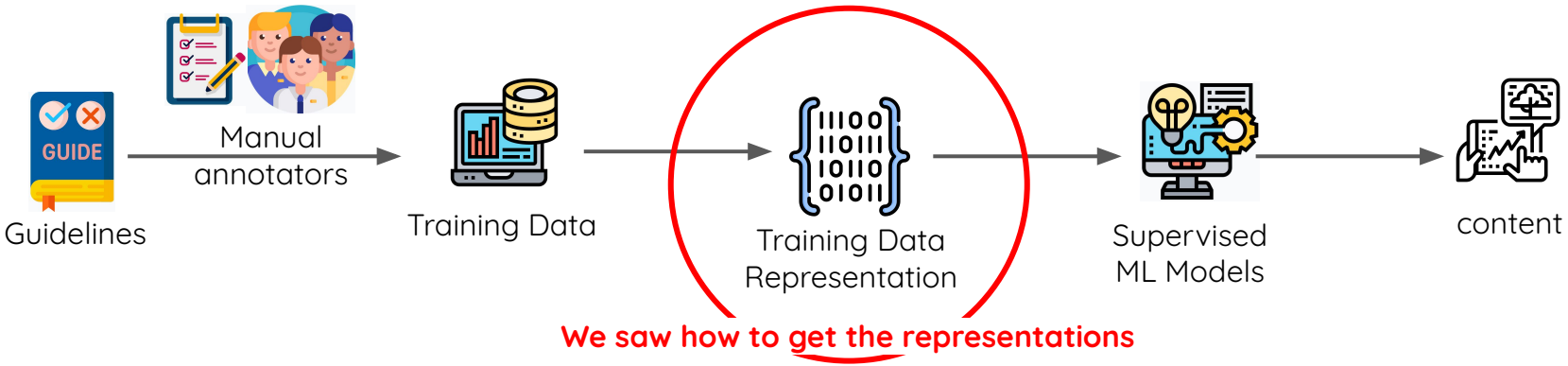
All we need is some labeled data or training data.



How to do supervised classification

We can use the representations we created (e.g., Bag-of-words) to further use for classification.

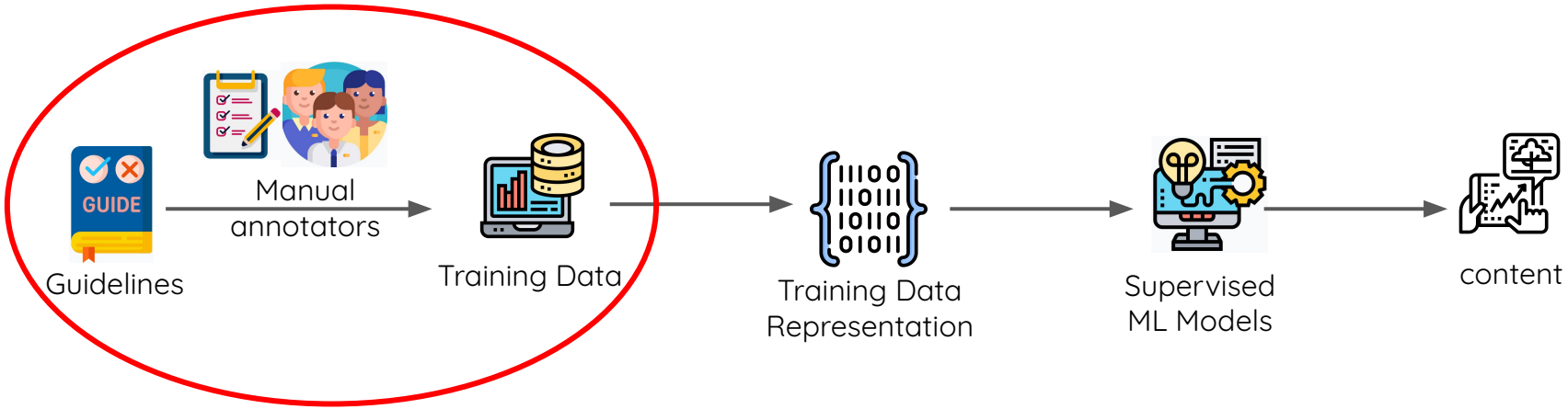
All we need is some labeled data or training data.



How to do supervised classification

We can use the representations we created (e.g., Bag-of-words) to further use for classification.

Now, all we need is some **labeled data** or **training data.**

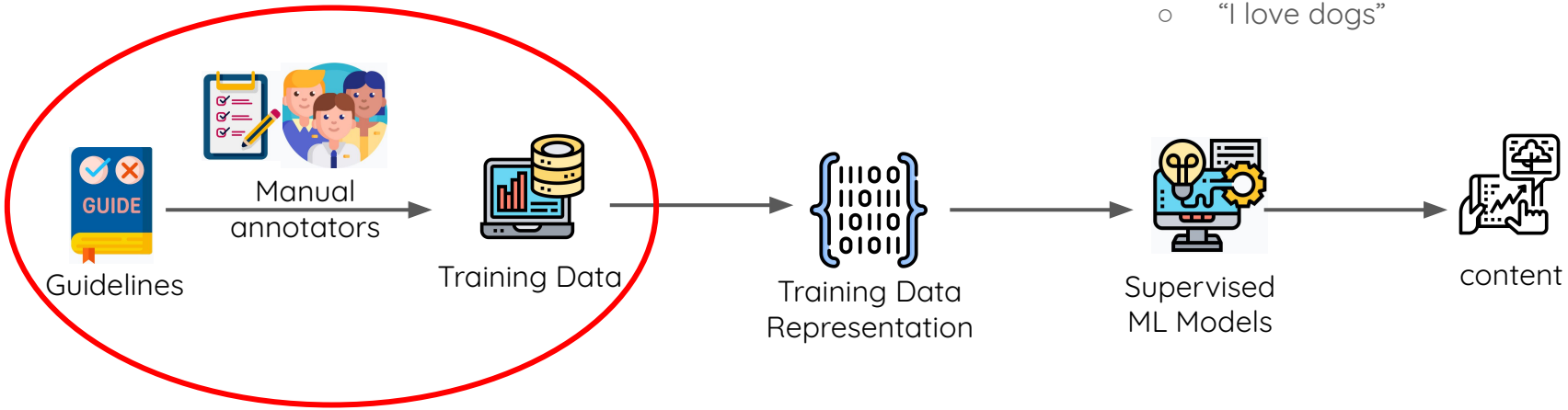


How to do supervised classification

We can use the representations we created (e.g., Bag-of-words) to further use for classification.

Now, all we need is some **labeled data** or **training data.**

- "I love cheese"
- "Dogs hate cheese"
- "I hate chocolate"
- "I love dogs"

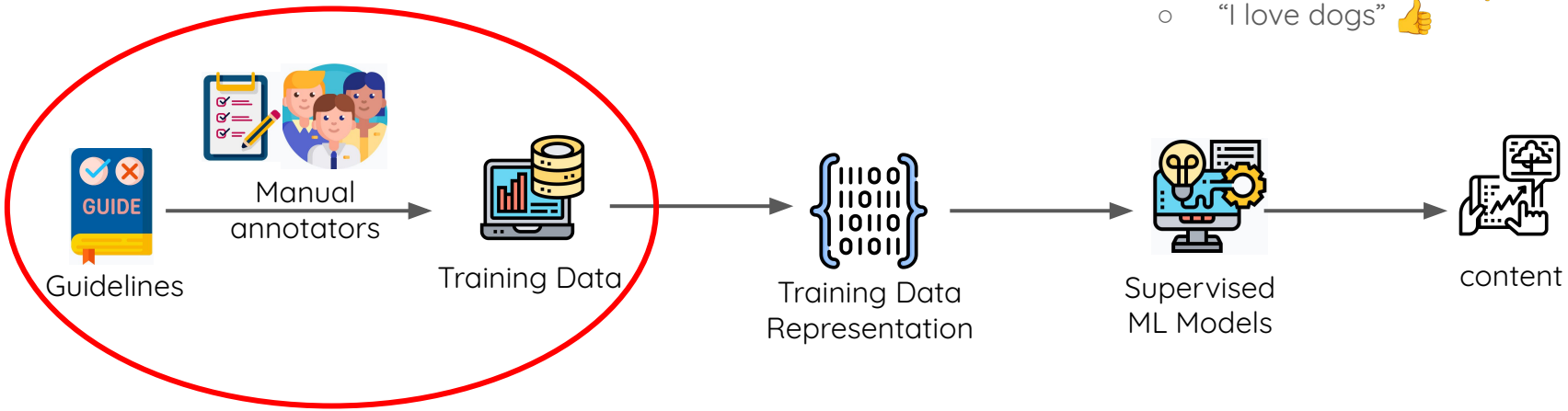


How to do supervised classification

We can use the representations we created (e.g., Bag-of-words) to further use for classification.

Now, all we need is some **labeled** **data** or **training data.**

- "I love cheese" 👍
- "Dogs hate cheese" 👎
- "I hate chocolate" 👎
- "I love dogs" 👍

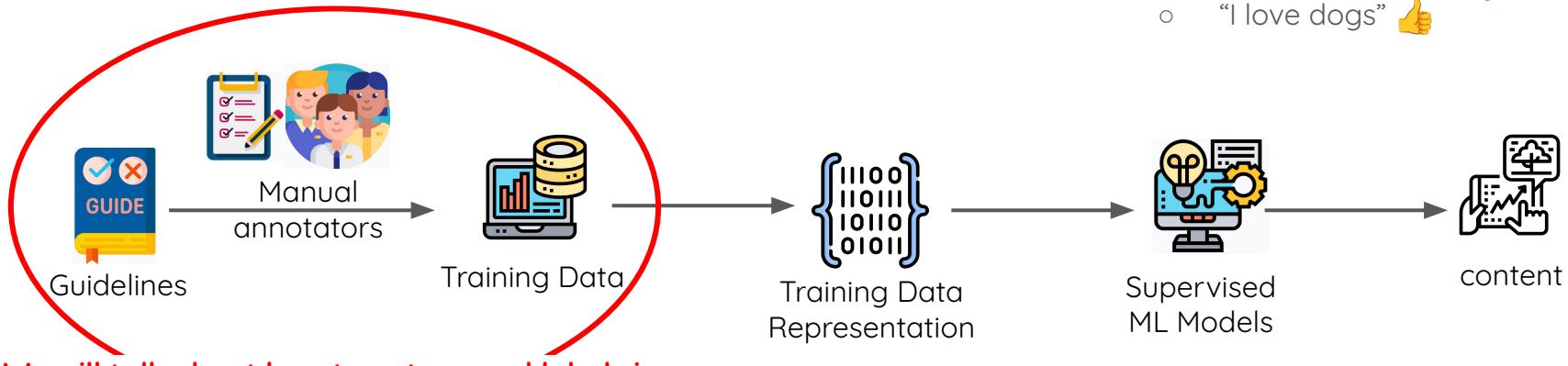


How to do supervised classification

We can use the representations we created (e.g., Bag-of-words) to further use for classification.

Now, all we need is some **labeled** **data** or **training data.**

- "I love cheese" 👍
- "Dogs hate cheese" 👎
- "I hate chocolate" 👎
- "I love dogs" 👍



We will talk about how to get manual labels in detail in Week 6 (content analysis)

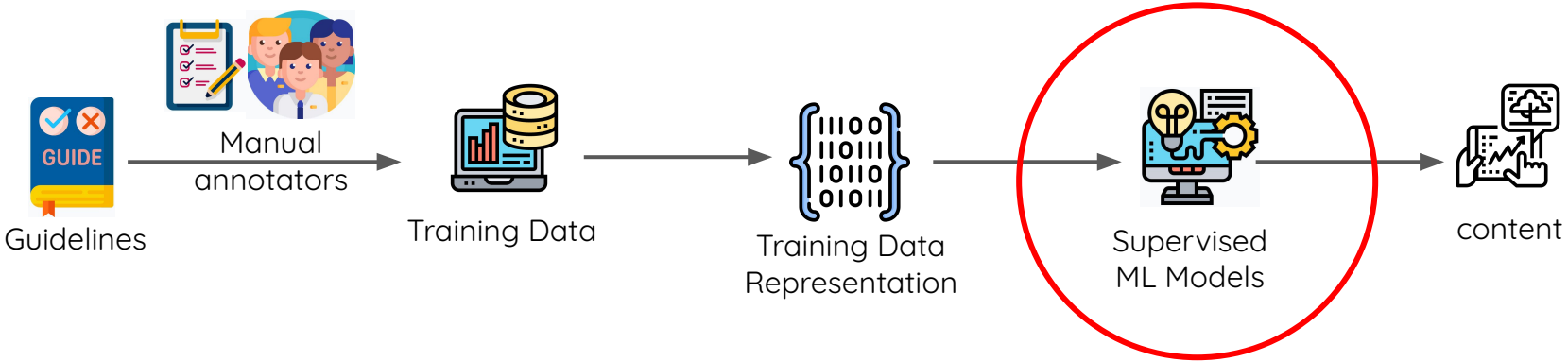
How to do supervised classification

We can use the representations we created (e.g., Bag-of-words) to further use for classification.

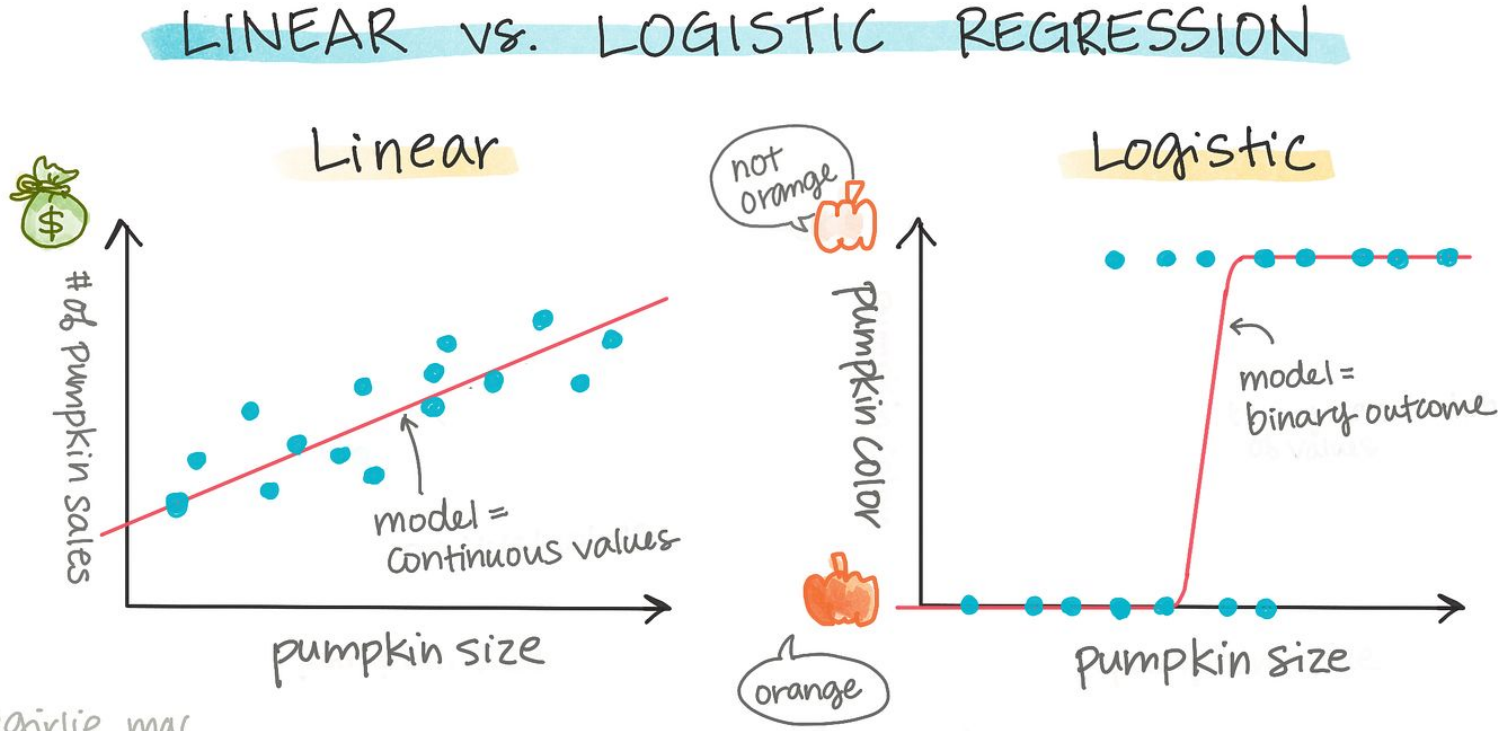
Now, all we need is some **labeled** **data** or **training data.**

- "I love cheese" 👍
- "Dogs hate cheese" 👎
- "I hate chocolate" 👎
- "I love dogs" 👍

And a **classifier**!



Classification: Logistic Regression

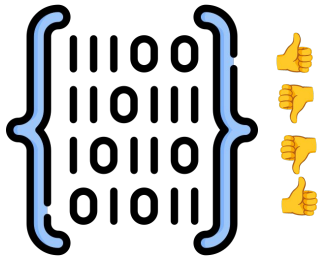


Classification: Logistic Regression

- “I love cheese” 👍
- “Dogs hate cheese”👎
- “I hate chocolate” 👎
- “I love dogs” 👍

Classification: Logistic Regression

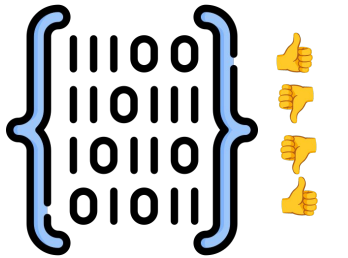
- “I love cheese” 👍
- “Dogs hate cheese”👎
- “I hate chocolate” 👎
- “I love dogs” 👍



👍
👎
👎
👍

Classification: Logistic Regression

- “I love cheese” 👍
- “Dogs hate cheese” 👎
- “I hate chocolate” 👎
- “I love dogs” 👍

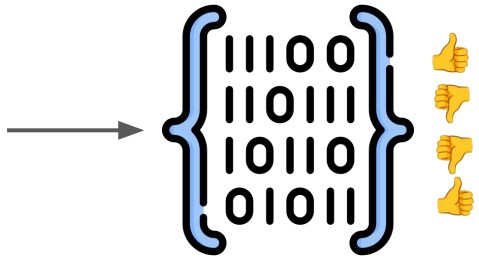


X y

features labels

Classification: Logistic Regression

- “I love cheese” 👍
- “Dogs hate cheese” 👎
- “I hate chocolate” 👎
- “I love dogs” 👍



A classifier *learns* the weights *W* or parameters associated with the features *X*

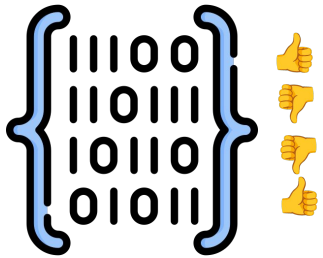
X *y*

features **labels**

$$z = W.X + b$$

Classification: Logistic Regression

- "I love cheese" 👍
- "Dogs hate cheese" 👎
- "I hate chocolate" 👎
- "I love dogs" 👍



A classifier *learns* the weights *W* or parameters associated with the features *X*

X *y*

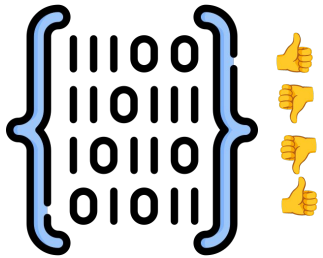
features **labels**

$$z = W.X + b$$

For each feature x_i, w_i tells us the importance of that feature We also have a bias term *b*.

Classification: Logistic Regression

- "I love cheese" 👍
- "Dogs hate cheese" 👎
- "I hate chocolate" 👎
- "I love dogs" 👍



A classifier *learns* the weights *W* or parameters associated with the features *X*

X *y*

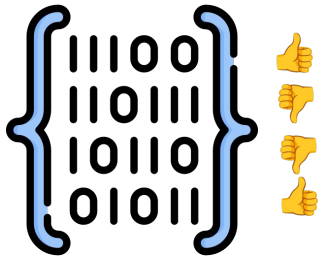
features **labels**

$$z = W.X + b$$

For each feature x_i, w_i tells us the importance of that feature We also have a bias term *b*. But how do we get *y* (the labels) from *z*?

Classification: Logistic Regression

- "I love cheese" 👍
- "Dogs hate cheese" 👎
- "I hate chocolate" 👎
- "I love dogs" 👍



A classifier *learns* the weights *W* or parameters associated with the features *X*

X *y*

features **labels**

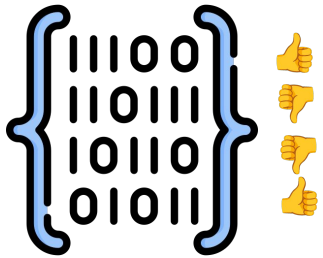
$$z = W.X + b$$

For each feature x_i, w_i tells us the importance of that feature We also have a bias term *b*. But how do we get *y* (the labels) from *z*?

Using the sigmoid or **logistic** function!

Classification: Logistic Regression

- "I love cheese" 👍
- "Dogs hate cheese" 👎
- "I hate chocolate" 👎
- "I love dogs" 👍



A classifier *learns* the weights *W* or parameters associated with the features *X*

X *y*

features **labels**

$z = W.X + b$

For each feature x_i, w_i tells us the importance of that feature We also have a bias term *b*. But how do we get *y* (the labels) from *z*?

Using the sigmoid or **logistic** function!

$$\hat{y} = \sigma(z) = \frac{1}{1+e^{-z}}$$

But how do we get w_i? And what is $\hat{y}$?

But how do we get w_i? And what is $\hat{y}$?

$$z = W.X + b$$

We can **estimate** y, i.e., $\hat{y}$

We start with some initial weights for W

$$\hat{y} = \sigma(z) = \frac{1}{1 + e^{-z}}$$

Then **optimize** to W's ideal value based on the difference between y and $\hat{y}$

But how do we get w_i? And what is $\hat{y}$?

$$z = W.X + b$$

We can **estimate** y, i.e., $\hat{y}$

We start with some initial weights for W

$$\hat{y} = \sigma(z) = \frac{1}{1 + e^{-z}}$$

Then **optimize** to W's ideal value based on the difference between y and $\hat{y}$

Concretely, we need:

But how do we get w_i? And what is $\hat{y}$?

$$z = W.X + b$$

We can **estimate** y, i.e., $\hat{y}$

We start with some initial weights for W

$$\hat{y} = \sigma(z) = \frac{1}{1 + e^{-z}}$$

Then **optimize** to W's ideal value based on the difference between y and $\hat{y}$

Concretely, we need:

- A **loss function** to measure the difference between y and $\hat{y}$

But how do we get w_i? And what is $\hat{y}$?

$$z = W.X + b$$

We can **estimate** y, i.e., $\hat{y}$

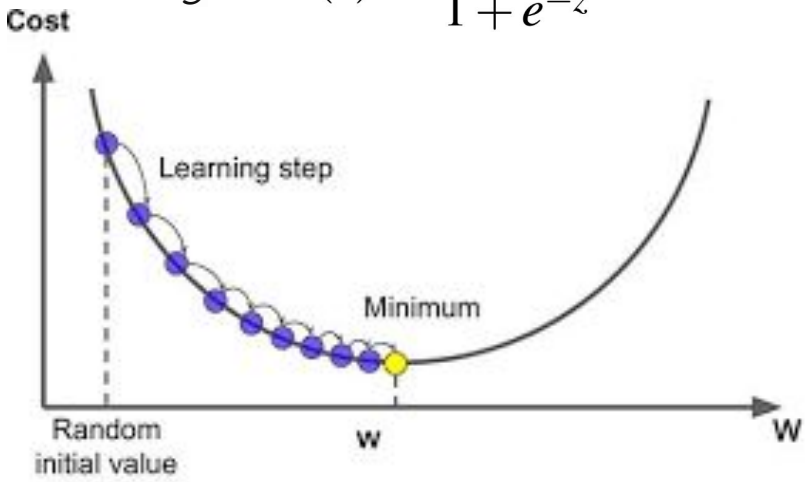
We start with some initial weights for W

Then **optimize** to W's ideal value based on the difference between y and $\hat{y}$

Concretely, we need:

- A **loss function** to measure the difference between y and $\hat{y}$
- An **optimization algorithm**

$$\hat{y} = \sigma(z) = \frac{1}{1 + e^{-z}}$$



Gradient Descent

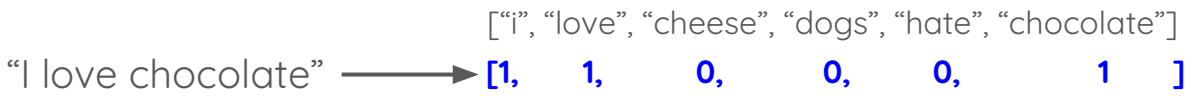
Labeling unknown documents or **Inference** or **Prediction**

For a new **unlabeled** document, we can get it's label using *W*

"I love chocolate"

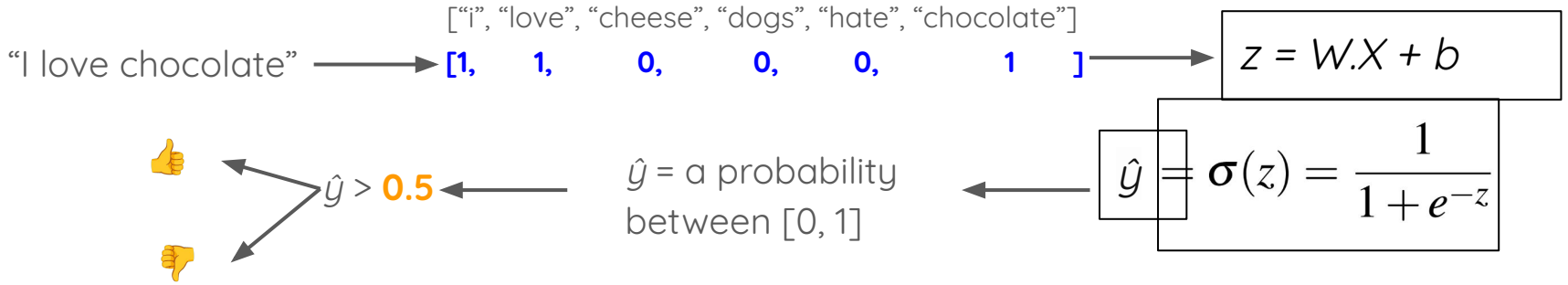
Labeling unknown documents or **Inference** or **Prediction**

For a new **unlabeled** document, we can get it's label using *W* and the new document's **feature representation**



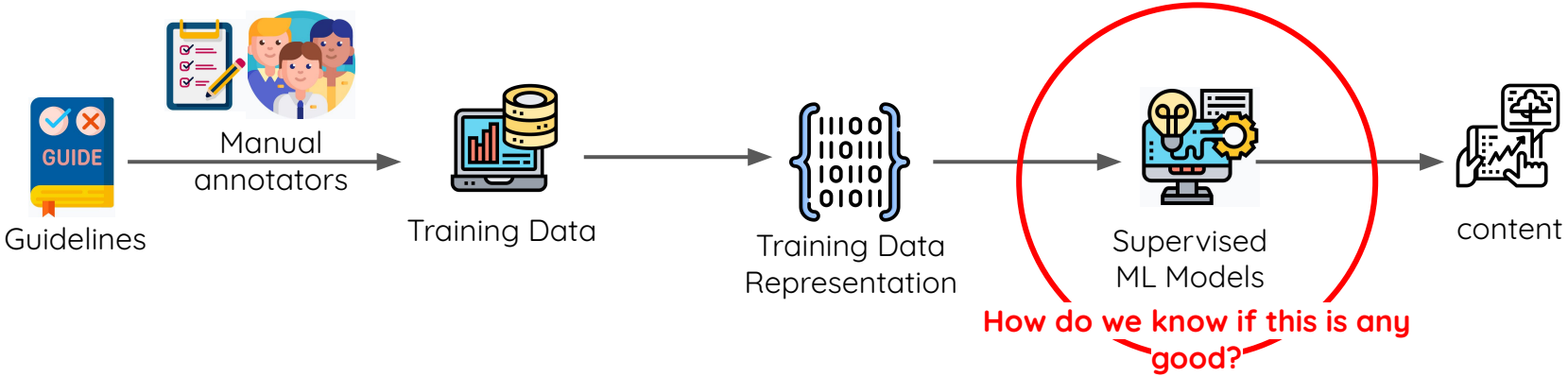
Labeling unknown documents or **Inference** or **Prediction**

For a new **unlabeled** document, we can get it's label using *W* and the new document's **feature representation** and a classification **threshold**:

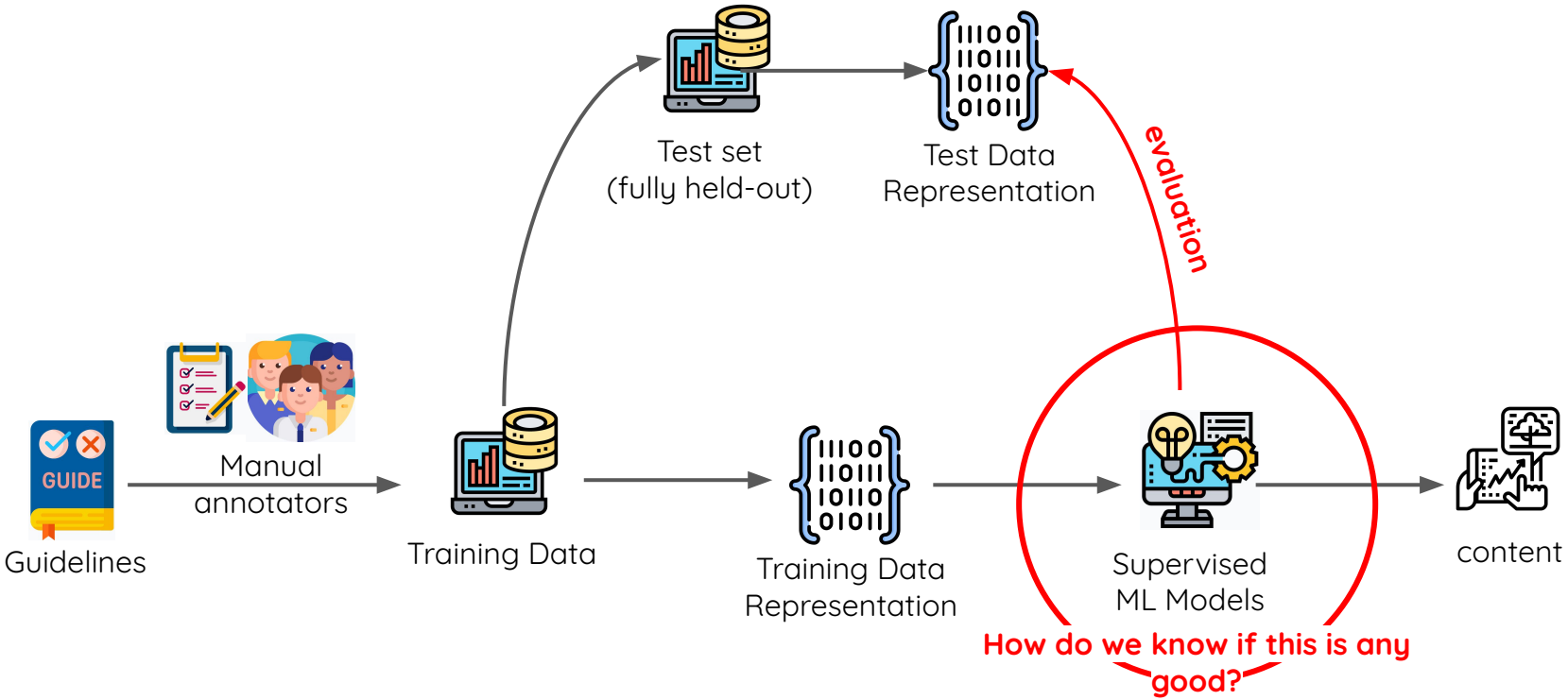


How do systematically evaluate the classifier’s performance?

Evaluating Text Classification



Evaluating Text Classification



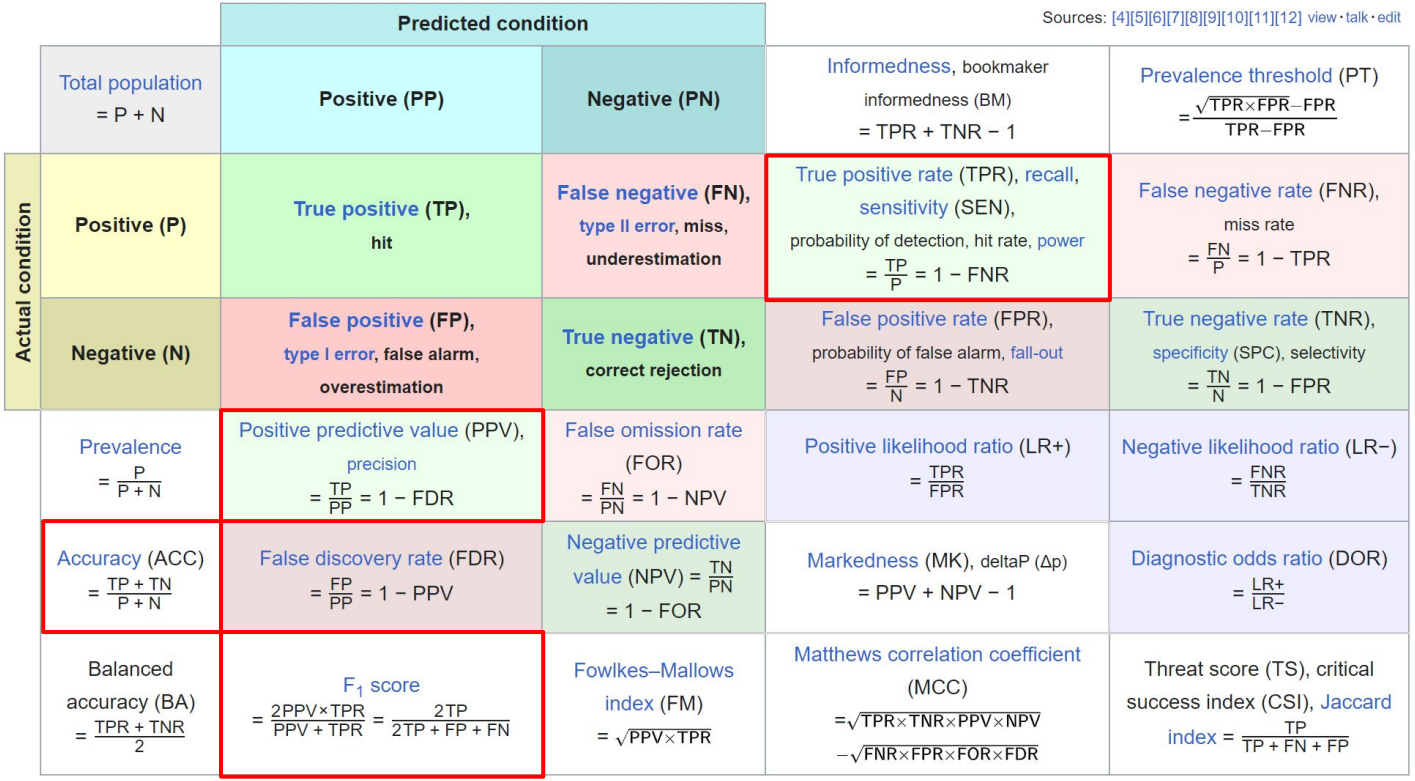
Evaluating Text Classification

Sources: [4][5][6][7][8][9][10][11][12] view · talk · edit

		Predicted condition			
	Total population = P + N	**Positive (PP)**	**Negative (PN)**	Informedness, bookmaker informedness (BM) = TPR + TNR − 1	Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	**Positive (P)**	**True positive (TP),** hit	**False negative (FN),** type II error, miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate $= \frac{FN}{P} = 1 - TPR$
	Negative (N)	**False positive (FP),** type I error, false alarm, overestimation	**True negative (TN),** correct rejection	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
	Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR−) $= \frac{FNR}{TNR}$
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN}$ $= 1 - FOR$	Markedness (MK), deltaP (Δp) = PPV + NPV − 1	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
	Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	F_1 score $= \frac{2PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$	Fowlkes–Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$

https://en.wikipedia.org/wiki/Precision_and_recall

Evaluating Text Classification

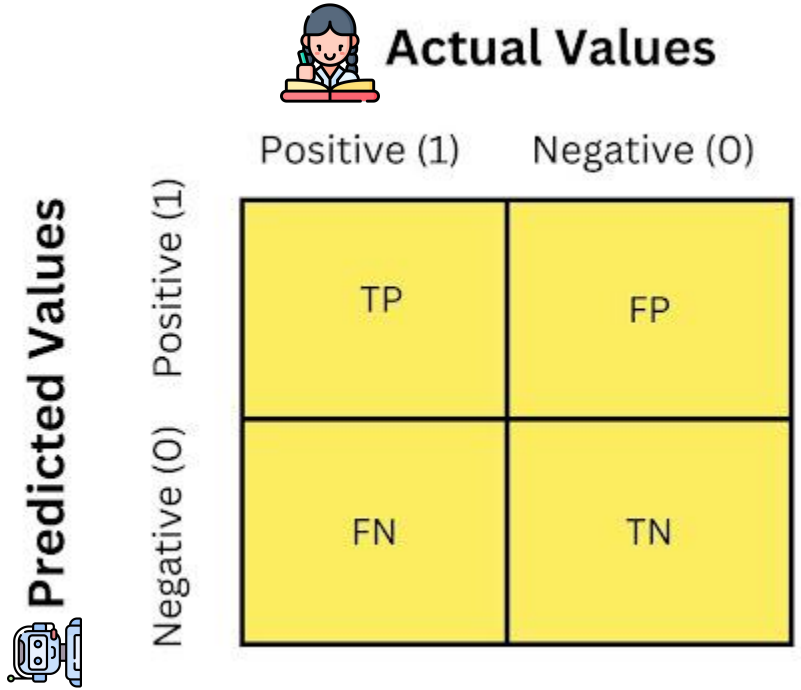


		Predicted condition			
	Total population = P + N	**Positive (PP)**	**Negative (PN)**	Informedness, bookmaker informedness (BM) = TPR + TNR − 1	Prevalence threshold (PT) = $\frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	**Positive (P)**	**True positive (TP),** hit	**False negative (FN),** type II error, miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power = $\frac{TP}{P}$ = 1 − FNR	False negative rate (FNR), miss rate = $\frac{FN}{P}$ = 1 − TPR
	Negative (N)	**False positive (FP),** type I error, false alarm, overestimation	**True negative (TN),** correct rejection	False positive rate (FPR), probability of false alarm, fall-out = $\frac{FP}{N}$ = 1 − TNR	True negative rate (TNR), specificity (SPC), selectivity = $\frac{TN}{N}$ = 1 − FPR
	Prevalence = $\frac{P}{P+N}$	Positive predictive value (PPV), precision = $\frac{TP}{PP}$ = 1 − FDR	False omission rate (FOR) = $\frac{FN}{PN}$ = 1 − NPV	Positive likelihood ratio (LR+) = $\frac{TPR}{FPR}$	Negative likelihood ratio (LR−) = $\frac{FNR}{TNR}$
	Accuracy (ACC) = $\frac{TP+TN}{P+N}$	False discovery rate (FDR) = $\frac{FP}{PP}$ = 1 − PPV	Negative predictive value (NPV) = $\frac{TN}{PN}$ = 1 − FOR	Markedness (MK), deltaP (Δp) = PPV + NPV − 1	Diagnostic odds ratio (DOR) = $\frac{LR+}{LR-}$
	Balanced accuracy (BA) = $\frac{TPR+TNR}{2}$	F_1 score = $\frac{2PPV \times TPR}{PPV+TPR} = \frac{2TP}{2TP+FP+FN}$	Fowlkes–Mallows index (FM) = $\sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) = $\sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index = $\frac{TP}{TP+FN+FP}$

Sources: [4][5][6][7][8][9][10][11][12] view · talk · edit

https://en.wikipedia.org/wiki/Precision_and_recall

Confusion Matrix: Basic, but useful



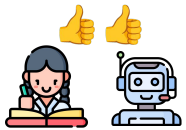
Example for Sentiment Analysis

- “Ground truth” or “gold labels” = manual labels 

Example for Sentiment Analysis

- “Ground truth” or “gold labels” = manual labels 

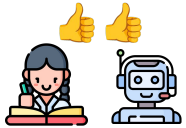
  - True positive (TP) = when something that was manually labeled as positive was also labeled positive by the classifier

    “I had a great day today” 

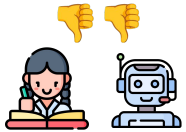
Example for Sentiment Analysis

- “Ground truth” or “gold labels” = manual labels 

  - True positive (TP) = when something that was manually labeled as positive was also labeled positive by the classifier

    “I had a great day today” 

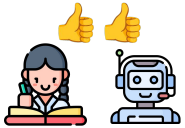
  - True negative (TN) = when something that was manually labeled as negative was also labeled negative by the classifier

    “I had an awful day today” 

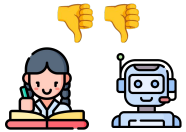
Example for Sentiment Analysis

- “Ground truth” or “gold labels” = manual labels 

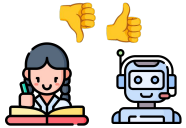
  - True positive (TP) = when something that was manually labeled as positive was also labeled positive by the classifier

    “I had a great day today” 

  - True negative (TN) = when something that was manually labeled as negative was also labeled negative by the classifier

    “I had an awful day today” 

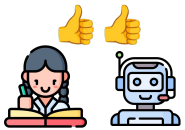
  - False positive (FP) = when something that was manually labeled as negative was ***falsely*** labeled positive by the classifier

    “I had a not so great day today” 

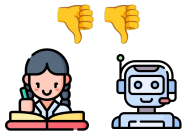
Example for Sentiment Analysis

- "Ground truth" or "gold labels" = manual labels 

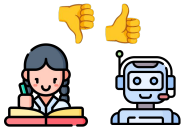
  - True positive (TP) = when something that was manually labeled as positive was also labeled positive by the classifier

    "I had a great day today" 

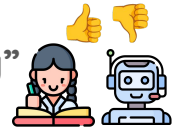
  - True negative (TN) = when something that was manually labeled as negative was also labeled negative by the classifier

    "I had an awful day today" 

  - False positive (FP) = when something that was manually labeled as negative was ***falsely*** labeled positive by the classifier

    "I had a not so great day today" 

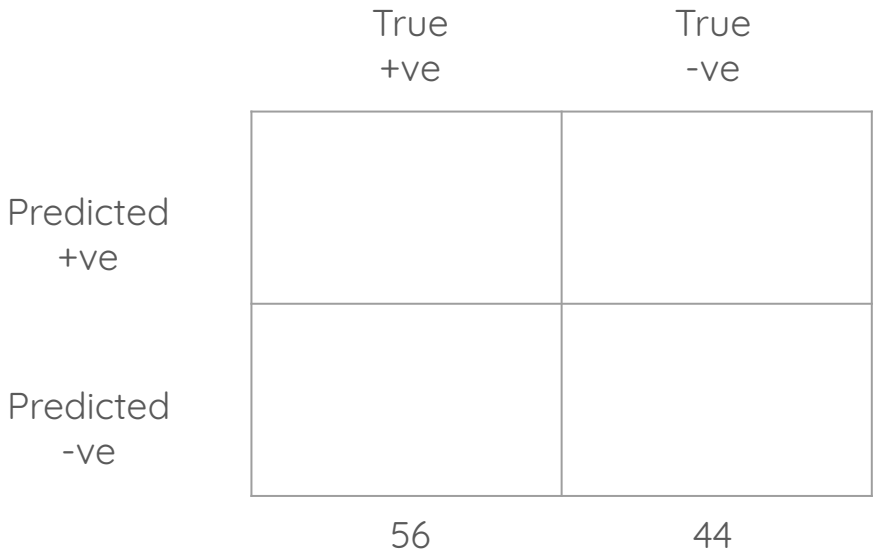
  - False negative (FN) = when something that was manually labeled as positive was ***falsely*** labeled negative by the classifier

    "I had an awfully great day today" 

Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$

- The classifier labels
  - __ posts as positive
  - __ posts as negative



Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$
- The classifier labels
  - __ posts as positive
  - __ posts as negative

	True +ve	True -ve
Predicted +ve	**48**	**3**
Predicted -ve	**8**	**41**
	56	44

Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$
- The classifier labels
  - 51 posts as positive
  - 49 posts as negative

	True +ve	True -ve	
Predicted +ve	**48**	**3**	51
Predicted -ve	**8**	**41**	49
	56	44	

Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$

- The classifier labels
  - 51 posts as positive
  - 49 posts as negative

- TPs:
- TNs:
- FPs:
- FNs:

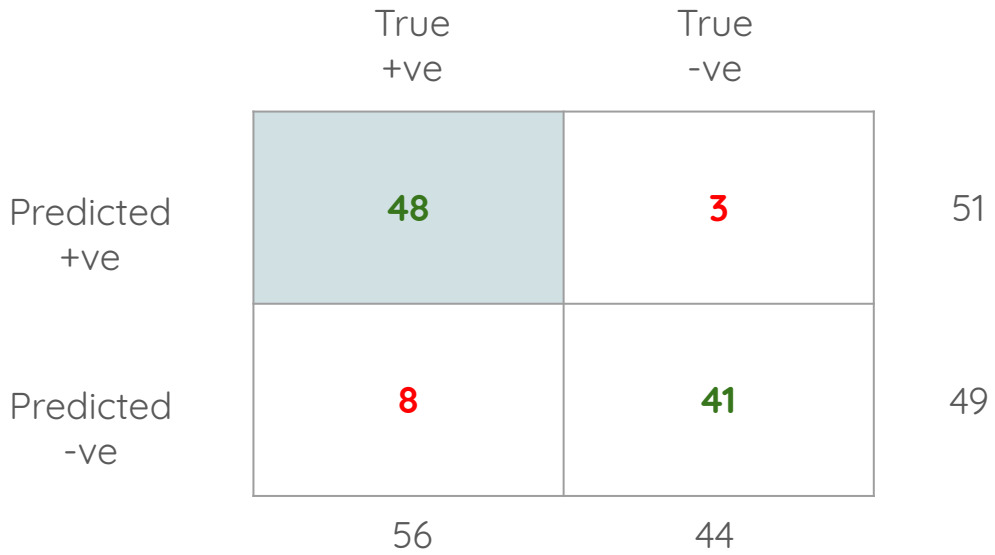
	True +ve	True -ve	
Predicted +ve	**48**	**3**	51
Predicted -ve	**8**	**41**	49
	56	44	

Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$

- The classifier labels
  - 51 posts as positive
  - 49 posts as negative

- TPs: 48
- TNs:
- FPs:
- FNs:

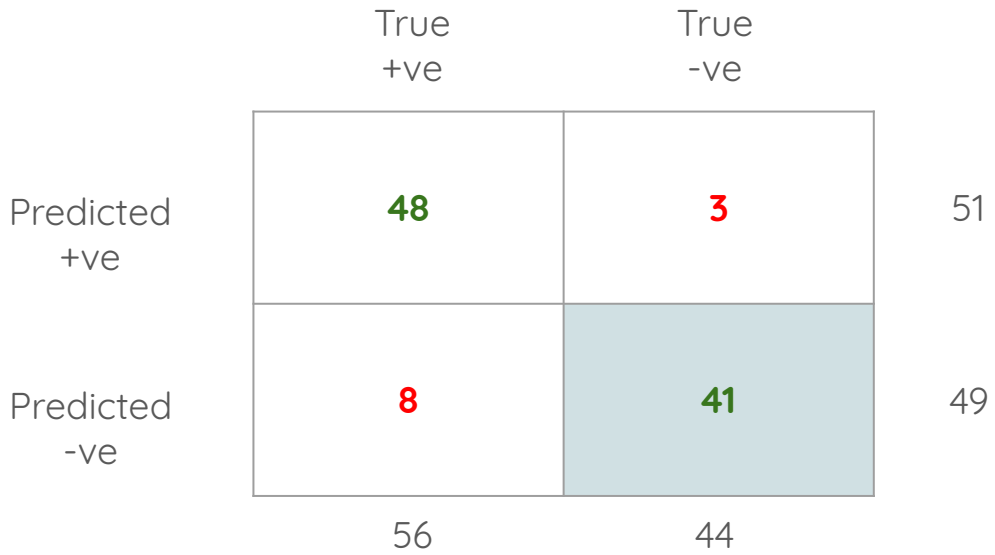


Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$

- The classifier labels
  - 51 posts as positive
  - 49 posts as negative

- TPs: 48
- TNs: 41
- FPs:
- FNs:

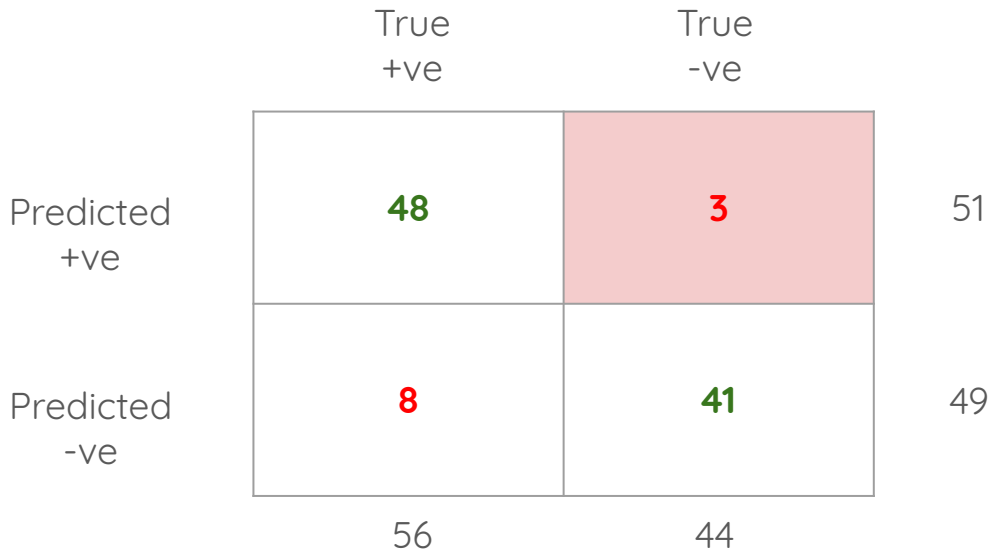


Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$

- The classifier labels
  - 51 posts as positive
  - 49 posts as negative

- TPs: 48
- TNs: 41
- FPs: 3
- FNs:

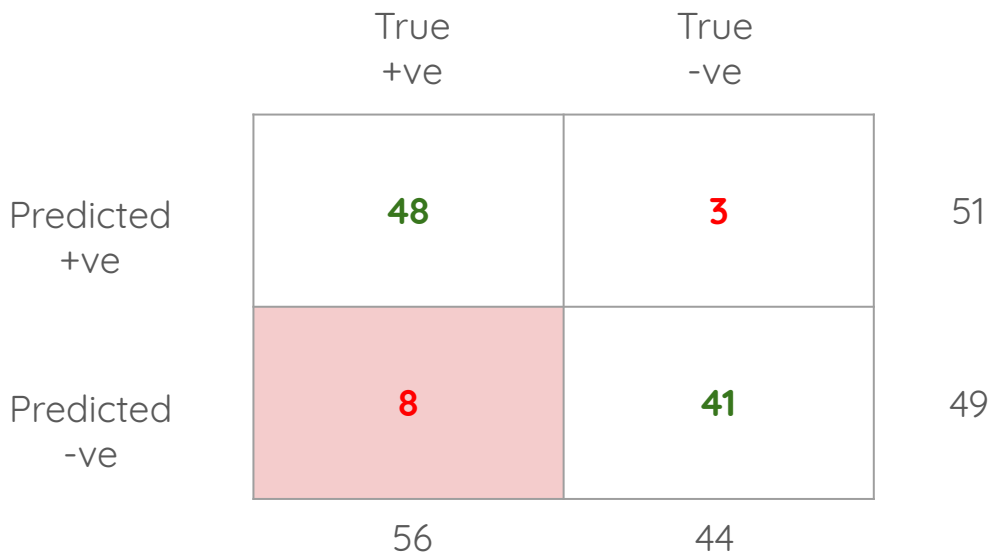


Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$

- The classifier labels
  - 51 posts as positive
  - 49 posts as negative

- TPs: 48
- TNs: 41
- FPs: 3
- FNs: 8



Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$

- The classifier labels
  - 51 posts as positive
  - 49 posts as negative

- TPs: 48
- TNs: 41
- FPs: 3
- FNs: 8

$$\text{Accuracy (ACC)} = \frac{TP + TN}{P + N}$$

- Accuracy =

	True +ve (P)	True -ve (N)	
Predicted +ve	**48**	**3**	51
Predicted -ve	**8**	**41**	49
	56	44	

Example for Sentiment Analysis

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**56**) and negative (**44**), i.e., $N_{classes} = 2$

- The classifier labels
  - 51 posts as positive
  - 49 posts as negative

- TPs: 48
- TNs: 41
- FPs: 3
- FNs: 8

Accuracy (ACC)

$$= \frac{TP + TN}{P + N}$$

- Accuracy = (48 + 41)/(100) = 89%

	True +ve (P)	True -ve (N)	
Predicted +ve	**48**	**3**	51
Predicted -ve	**8**	**41**	49
	56	44	

Example for Sentiment Analysis: Imbalanced classes

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**96**) and negative (**4**), i.e., $N_{classes} = 2$

- The classifier labels
  - 99 posts as positive
  - 1 post as negative

- TPs:
- TNs:
- FPs:
- FNs:

$$\text{Accuracy (ACC)} = \frac{TP + TN}{P + N}$$

- Accuracy =

	True +ve (P)	True -ve (N)	
Predicted +ve	**96**	**3**	99
Predicted -ve	**0**	**1**	1
	96	4	

Example for Sentiment Analysis: Imbalanced classes

- Let's say we have 100 item in our test set, i.e., 100 social media posts manually annotated into positive (**96**) and negative (**4**), i.e., $N_{classes} = 2$

- The classifier labels
  - 99 posts as positive
  - 1 post as negative

- TPs: 96
- TNs: 1
- FPs: 3
- FNs: 0

$$\text{Accuracy (ACC)} = \frac{TP + TN}{P + N}$$

- Accuracy = (96 + 1)/(100) = 97%

	True +ve (P)	True -ve (N)	
Predicted +ve	**96**	**3**	99
Predicted -ve	**0**	**1**	1
	96	4	

Evaluating Text Classification

		Predicted condition			
	Total population = P + N	**Positive (PP)**	**Negative (PN)**	Informedness, bookmaker informedness (BM) = TPR + TNR − 1	Prevalence threshold (PT) = $\frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	**Positive (P)**	**True positive (TP),** hit	**False negative (FN),** type II error, miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power = $\frac{TP}{P}$ = 1 − FNR	False negative rate (FNR), miss rate = $\frac{FN}{P}$ = 1 − TPR
	Negative (N)	**False positive (FP),** type I error, false alarm, overestimation	**True negative (TN),** correct rejection	False positive rate (FPR), probability of false alarm, fall-out = $\frac{FP}{N}$ = 1 − TNR	True negative rate (TNR), specificity (SPC), selectivity = $\frac{TN}{N}$ = 1 − FPR
	Prevalence = $\frac{P}{P+N}$	Positive predictive value (PPV), precision = $\frac{TP}{PP}$ = 1 − FDR	False omission rate (FOR) = $\frac{FN}{PN}$ = 1 − NPV	Positive likelihood ratio (LR+) = $\frac{TPR}{FPR}$	Negative likelihood ratio (LR−) = $\frac{FNR}{TNR}$
	Accuracy (ACC) = $\frac{TP+TN}{P+N}$	False discovery rate (FDR) = $\frac{FP}{PP}$ = 1 − PPV	Negative predictive value (NPV) = $\frac{TN}{PN}$ = 1 − FOR	Markedness (MK), deltaP (Δp) = PPV + NPV − 1	Diagnostic odds ratio (DOR) = $\frac{LR+}{LR-}$
	Balanced accuracy (BA) = $\frac{TPR+TNR}{2}$	F_1 score = $\frac{2PPV \times TPR}{PPV+TPR} = \frac{2TP}{2TP+FP+FN}$	Fowlkes–Mallows index (FM) = $\sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) = $\sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index = $\frac{TP}{TP+FN+FP}$

Sources: [4][5][6][7][8][9][10][11][12] view · talk · edit

https://en.wikipedia.org/wiki/Precision_and_recall

Summary

- This session we learned:
  - About the computational turn in the Social Sciences
  - About LLMs and how they contribute to the Social Sciences
  - The first building blocks of LLMs
    - Words —> Tokens
    - Text Representations
    - Using text representations in text classification

Summary and Next Week

- This session we learned:
  - About the computational turn in the Social Sciences
  - About LLMs and how they contribute to the Social Sciences
  - The first building blocks of LLMs
    - Words —> Tokens
    - Text Representations
    - Using text representations in text classification

- Next time:
  - More efficient representations —> word embeddings
  - Doing more with text representations —> text generation
  - More powerful models for classification —> deep learning

Questions?

Please talk to me after the course if you had problems registering

Additional Resources

- The Text Analytics course (IS 661)
- Daniel Jurafsky and James H. Martin. 2025. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition with Language Models, 3rd edition. Online manuscript released August 24, 2025. https://web.stanford.edu/~jurafsky/slp3
  - https://web.stanford.edu/~jurafsky/slp3/slides/tokens_aug25.pdf
  - https://web.stanford.edu/~jurafsky/slp3/slides/logreg25aug.pdf
- Elena Voita's NLP course for you: https://lena-voita.github.io/nlp_course.html
  - https://lena-voita.github.io/nlp_course/text_classification.html

Classification: Naive Bayes

Upcoming Courses

IS 662	Network Science ☀️
IS 557	Public Blockchains ☀️
IE/IS 661	Text Analytics ☃️
IS 628	Seminar Advances in Public Blockchain ☃️
CS 721	Seminar Data Science I (Methods) ☀️/☃️
IS 723	Seminar Data Science II (Empirical Studies) ☀️/☃️
	Team Projects ☀️/☃️
	Master's Theses ☀️/☃️

up-to-date list online