PennState
College of the
Liberal Arts

C-SoDA
Center for Social Data Analytics

Day 1 - Introduction

Advanced Text as Data: Natural Language Processing
Essex Summer School in Social Science Data Analysis

Burt L. Monroe (Instructor) & Sam Bestvater (TA)
Pennsylvania State University

July 26, 2021

Why are we here?

Amazing advances in technology for understanding and generating natural language suggest that NLP has more to offer the social scientific study of "text-as-data."

We're going beyond the bag-of-words...

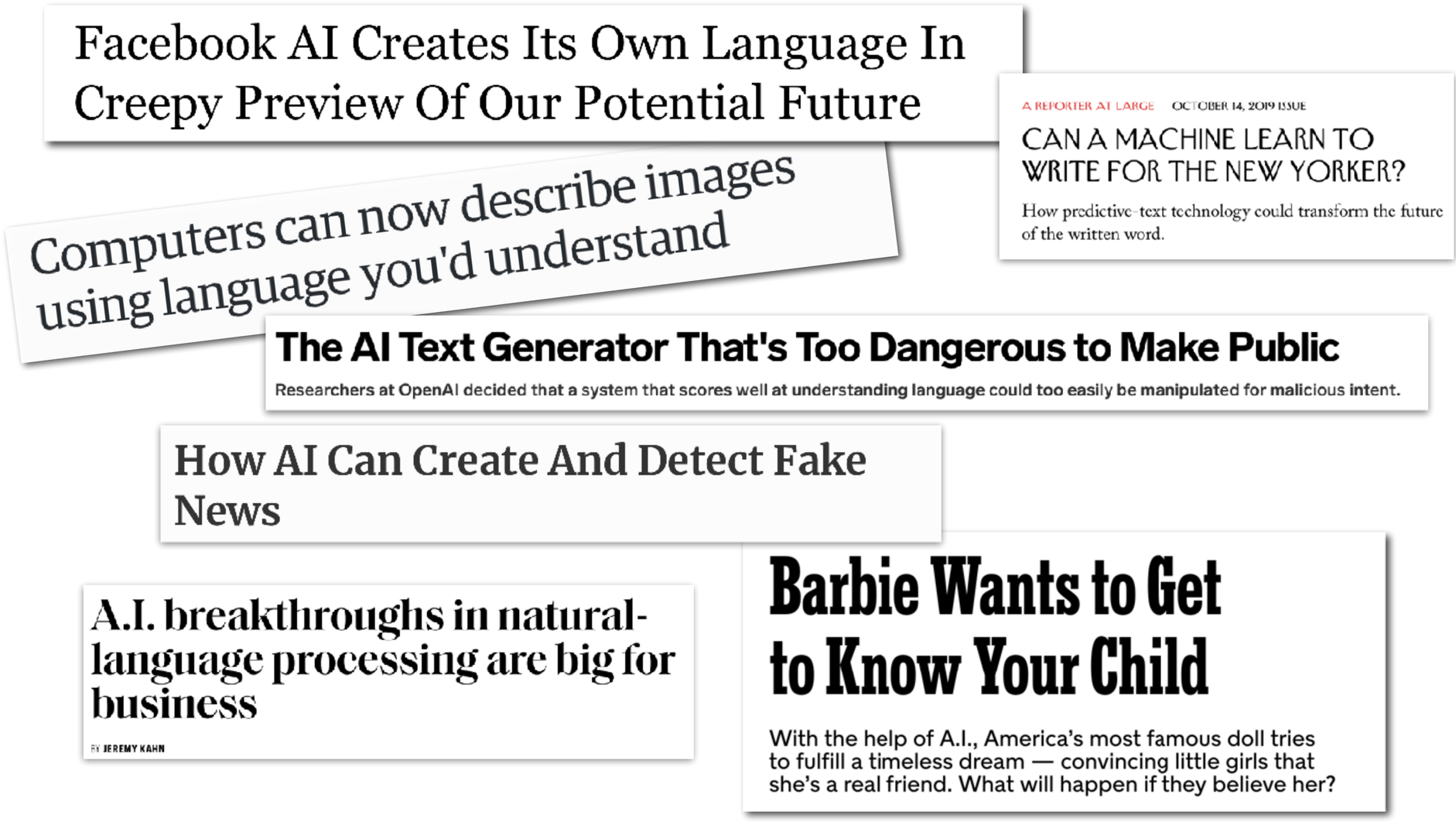


Image source: Julia Hockenmaier, 2020

Maybe, I thought, I can get AI to teach the course!

Maybe, I thought, I can get AI to teach the course!

So I asked the model GPT-2 to tell me what you might be concerned about.

Maybe, I thought, I can get AI to teach the course!

So I asked the model GPT-2 to tell me what you might be concerned about.

https://transformer.huggingface.co/doc/gpt2-large

The students were apprehensive about the first day of class with Professor Monroe.
They had heard he was

Maybe, I thought, I can get AI to teach the course!

So I asked the model GPT-2 to tell me what you might be concerned about.

https://transformer.huggingface.co/doc/gpt2-large

The students were apprehensive about the first day of class with Professor Monroe. They had heard he was a bit of a crank, and they were still not certain if he was even a professor. The professor made no attempt to help the students, instead he sat down, crossed his arms, and started lecturing. "First, let me begin by acknowledging that you're the first person to attend this course. I'd like you to be the first to know that I have no interest in your particular subject. As such, you're the first one who will be going to any classes , or any lectures. We're all here for the learning, and I'd like you to help us learn it."

Maybe, I thought, I can get AI to teach the course!

So I asked the model GPT-2 to tell me what you might be concerned about.

https://transformer.huggingface.co/doc/gpt2-large

The students were apprehensive about the first day of class with Professor Monroe. They had heard he was a bit of a crank, and they were still not certain if he was even a professor. The professor made no attempt to help the students, instead he sat down, crossed his arms, and started lecturing. "First, let me begin by acknowledging that you're the first person to attend this course. I'd like you to be the first to know that I have no interest in your particular subject. As such, you're the first one who will be going to any classes , or any lectures. We're all here for the learning, and I'd like you to help us learn it."

Hmmm … I guess we do need a human at the wheel.

Maybe, I thought, I can get AI to teach the course!

So I asked the model GPT-2 to tell me what you might be concerned about.

https://transformer.huggingface.co/doc/gpt2-large

The students were apprehensive about the first day of class with Professor Monroe. They had heard he was a bit of a crank, and they were still not certain if he was even a professor. The professor made no attempt to help the students, instead he sat down, crossed his arms, and started lecturing. "First, let me begin by acknowledging that you're the first person to attend this course. I'd like you to be the first to know that I have no interest in your particular subject. As such, you're the first one who will be going to any classes , or any lectures. We're all here for the learning, and I'd like you to help us learn it."

Hmmm … I guess we do need a human at the wheel.

Although my wife says this is flawless proof that it works.

Couldn’t resist another. This one from GPT-3.

Human Prompt (given to GPT-3)

At the party, I poured myself a glass of lemonade,
but it turned out to be too sour, so I added a little sugar.
I didn’t see a spoon handy, so I stirred it with a cigarette.
But that turned out to be a bad idea because

[GPT-3’s generated continuation]

it kept falling on the floor. That’s when he decided to start the Cremation Association of North America, which has become a major cremation provider with 145 locations.

from Marcus & Davis ’20
https://www.technologyreview.com/2020/08/22/1007539/gpt3-openai-language-generator-artificial-intelligence-ai-opinion/

Today

- Introductions
- Course Logistics
- Overview - “Text as Data” and/vs. NLP
- Modern/neural NLP

- Demos: R in RStudio Cloud, Python in Google Colab

Introductions

Instructor: Burt Monroe
burtmonroe@psu.edu

TA: Sam Bestvater
seb64@psu.edu

Course Schedule Page

- https://burtmonroe.github.io/TextAsDataCourse/Essex/
- Readings (and any changes)
- Slides for that day's lecture.
- R and/or Python materials to be covered that day (more on that in a minute).
    - R materials will be in the class space on RStudioCloud.
    - Python materials will be on Google Colab.

Software

- We will be demonstrating various techniques in R and Python.

- Where sensible and feasible, we have tried to provide notebooks in both. This isn't entirely practical for all of the material. You will, in particular, notice that more neural net / deep learning examples are in Python than R.

- R material, in the form of notebooks, is provided through Essex's RStudio Cloud account. This will allow you to modify and run copies of the notebooks without installation, versioning, memory, etc. worries on your own machines. (If you do try to run on your own machines, be careful/deliberate about installations of things like RTools/devtools and Python / Java / etc. that might be required for some code.).

- Python material, also in notebooks is provided through Google Colab. Again, this allows you to modify and run copies of the notebooks without installation worries on your own machines. Further, Google Colab provides free access to GPU and TPU computing, which can be the difference in training neural nets between "that wasn't too bad" and "I'm going to have grandchildren before this finishes."

Stuff I am assuming you already know, more or less

- Basics of string manipulation / regular expressions (in R &/or Python)
- Basic text data management, preprocessing and creation of bag-of-words / doc-term matrix data.
- Cosine similarity.
- Building, training, evaluating and interpreting a “classical” (non-neural) text classifier [e.g., Naive Bayes, logistic regression / lasso / ridge regression, support vector machine, random forest, XGBoost] based on a bag-of-words representation.
- Building, training, evaluating and interpreting a topic model [e.g., LDA, CTM, STM] (based on a bag-of-words representation).
- I also won’t be going over any detail about data collection and cleaning — things like web scraping, OCR, and pdfs. (I do have some tutorials available.)

Bridging Text-as-Data and NLP — Beyond the “Bag of Words”

- Not just “words” — changing the things in the bag. (New or more targeted features.)
- Not discrete words, but infinite shades of meaning. (e.g., Embeddings)
- The bags are filled with different sets of words (e.g., Multilingual text)
- All the bags — leveraging what we’ve learned from *other* related bags. (Transfer learning)
- Not just a bag — getting words from a pipe or finding them on a network. (Better models of language production — more accurate measurement, more validity. Language generation.)

Bridging Text-as-Data and NLP — Beyond the “Bag of Words”

- Doing similar tasks differently and hopefully better (more validity, greater accuracy, less bias, more fairness).
    - New ways to turn text into data, which we analyze the same way.
    - New ways to analyze text-based data. (e.g., deep learning / neural NLP)
- Doing similar tasks more cheaply, at greater scale, at broader scope, in more depth. Increasing / democratizing the accessibility of tasks to more people.
- Doing new types of tasks.

What we *will* cover

- NLP “pipelines”. General concepts and practical application of NLP labeling tasks like part-of-speech tagging, named entity recognition, and dependency parsing. (Day 2)
- Word embeddings - what, why and how. How to use them. How to estimate them. (Day 3)
- Deep learning / neural NLP - General neural network concepts, feedforward networks, building and training models (mainly in Keras/Tensorflow). (Days 4 & 5)
- Recurrent & convolutional neural nets, attention and transformers (Day 6)
- Contextual embeddings, pretrained language models, and transfer learning (BERT and similar) (Day 7)
- Multilingual text as data, machine translation (Day 8)
- Natural language understanding (NLU) & Natural language generation (NLG) (Day 9)
- Fairness & bias in NLP (Day 10)

Text as Data

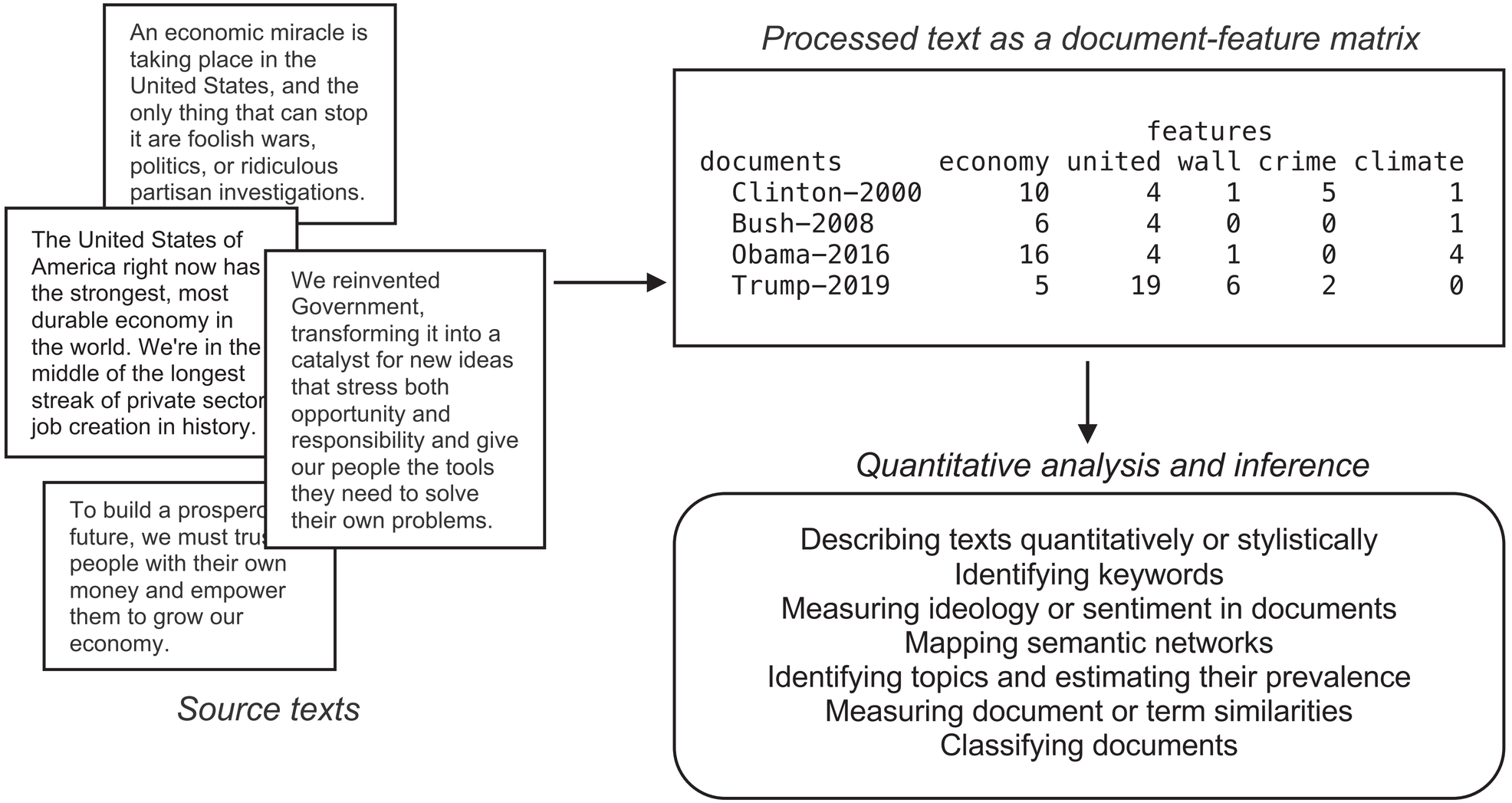


Figure 26.1 From text to data to data analysis

Source: Benoit. 2020

Common goals & methods in text as data

Quantitative analysis and inference

Describing texts quantitatively or stylistically
Identifying keywords
Measuring ideology or sentiment in documents
Mapping semantic networks
Identifying topics and estimating their prevalence
Measuring document or term similarities
Classifying documents

Source: Benoit 2020.

Most text-as-data tasks are latent measurement tasks

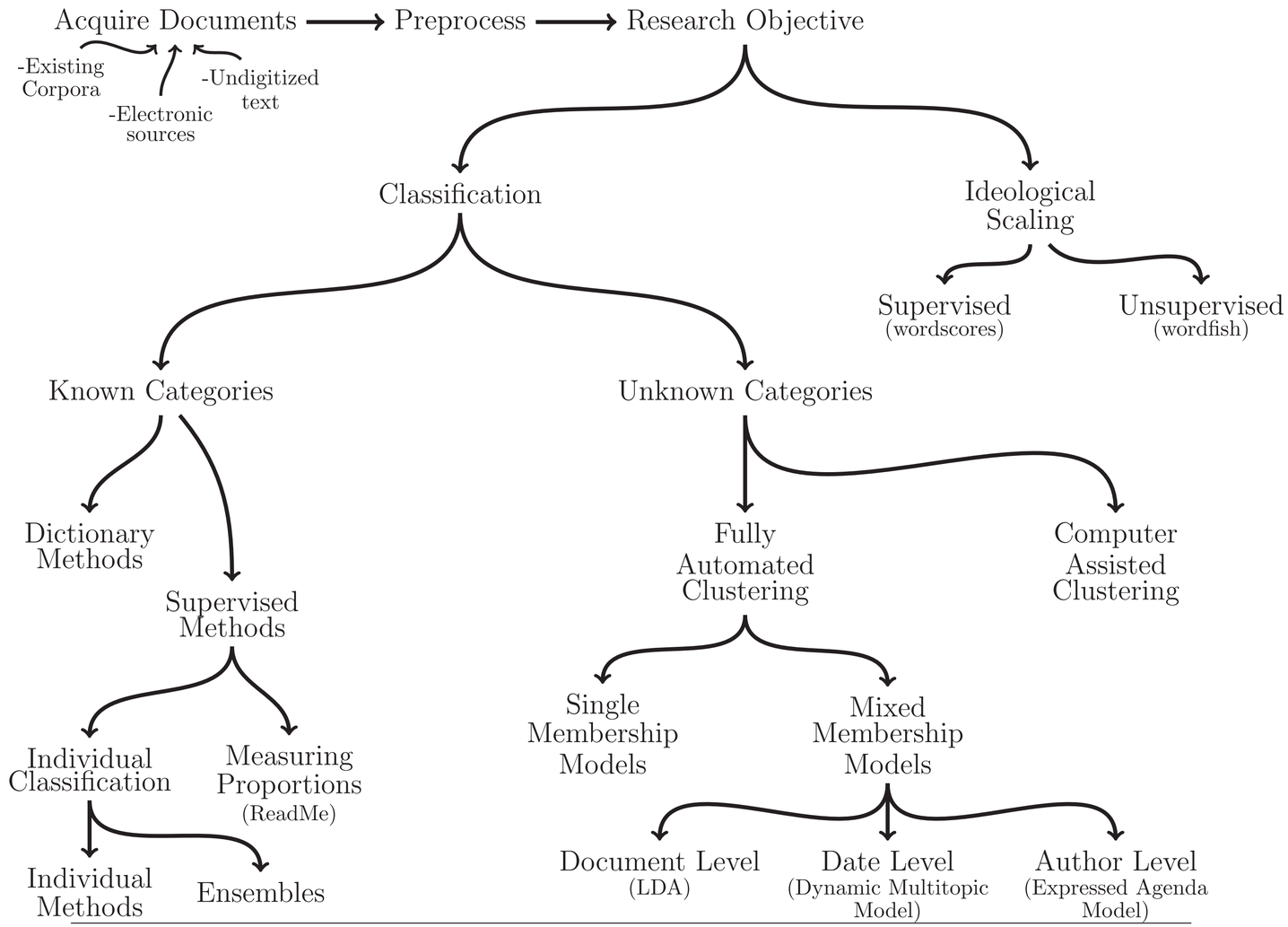


Fig. 1 An overview of text as data methods.

Source: Grimmer and Stewart, 2013.

Common goals & methods in text as data

Quantitative analysis and inference

Describing texts quantitatively or stylistically can be
Identifying keywords can be
Measuring ideology or sentiment in documents
Mapping semantic networks
Identifying topics and estimating their prevalence
Measuring document or term similarities can be
Classifying documents

Source: Benoit 2020.

Most text-as-data tasks are
latent measurement tasks

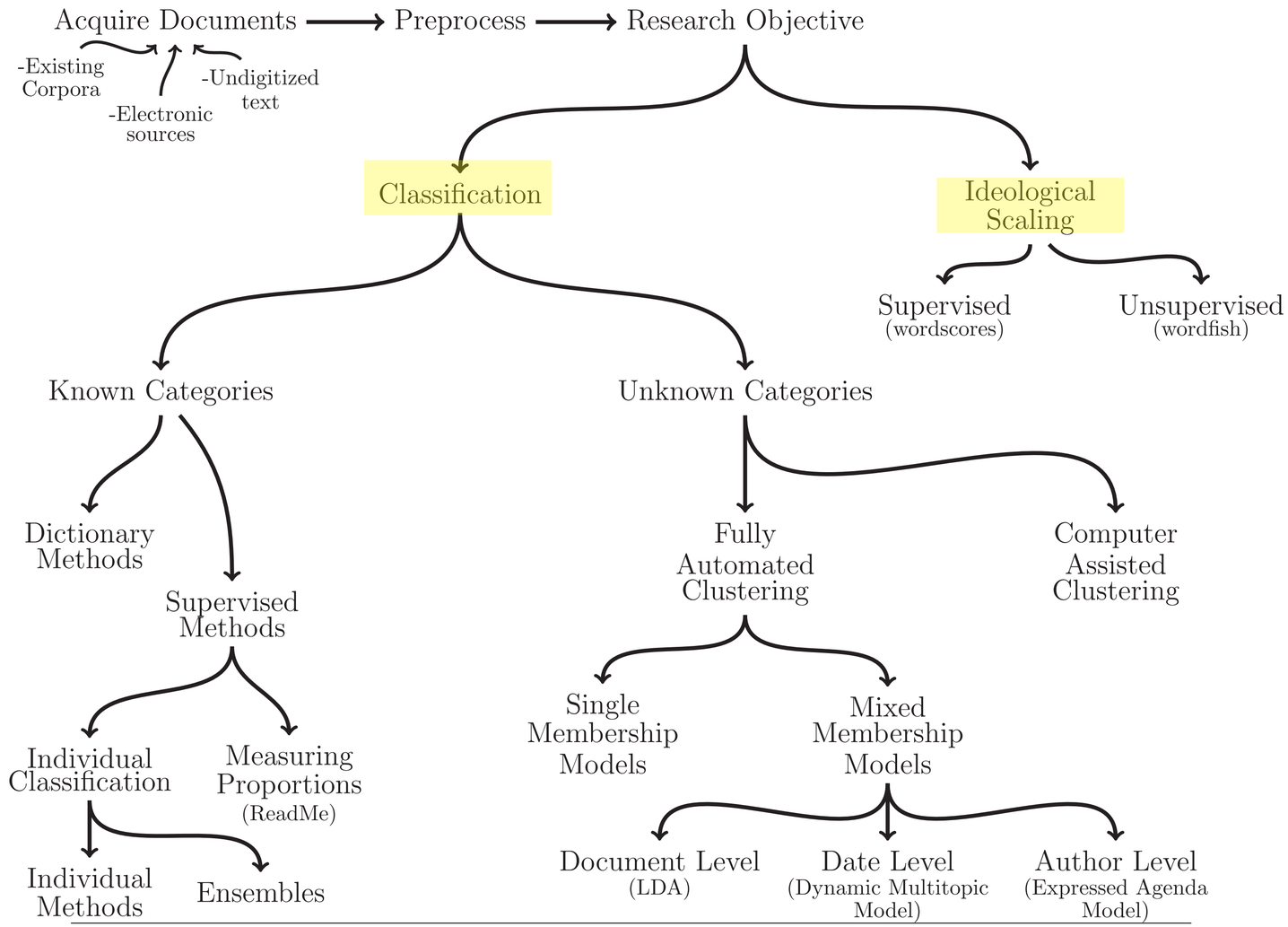


Fig. 1 An overview of text as data methods.

Source: Grimmer and Stewart, 2013.

This typical text-as-data pipeline converts text to numeric data through "pre-processing" that

(1) divides the text into the units we might care about,
(2) throws out some we're pretty sure we don't care about, and
(3) combines units that we're pretty sure should count as the same thing.

Tomorrow, we will look at NLP pipelines that do more elaborate and language-aware processing.

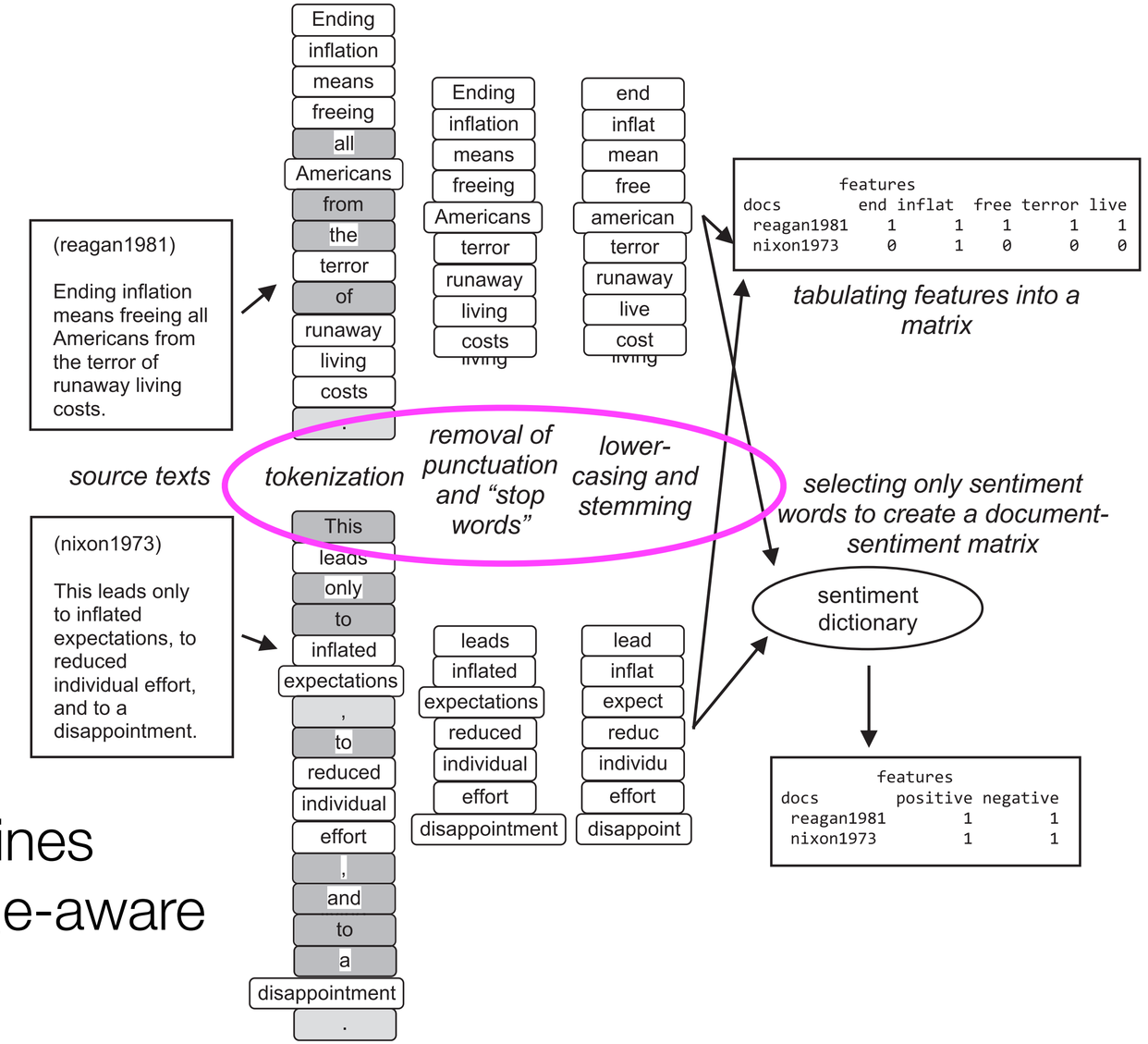


Figure 26.2 From text to tokens to matrix

Figure source: Benoit. 2020

Dictionary Methods

- Most basic - you have a list of words that count, or have a predetermined “score” for each category and you have the computer count/add them up.
- Very popular approach to sentiment analysis.
  - VADER
  - Lexicoder
  - LIWC

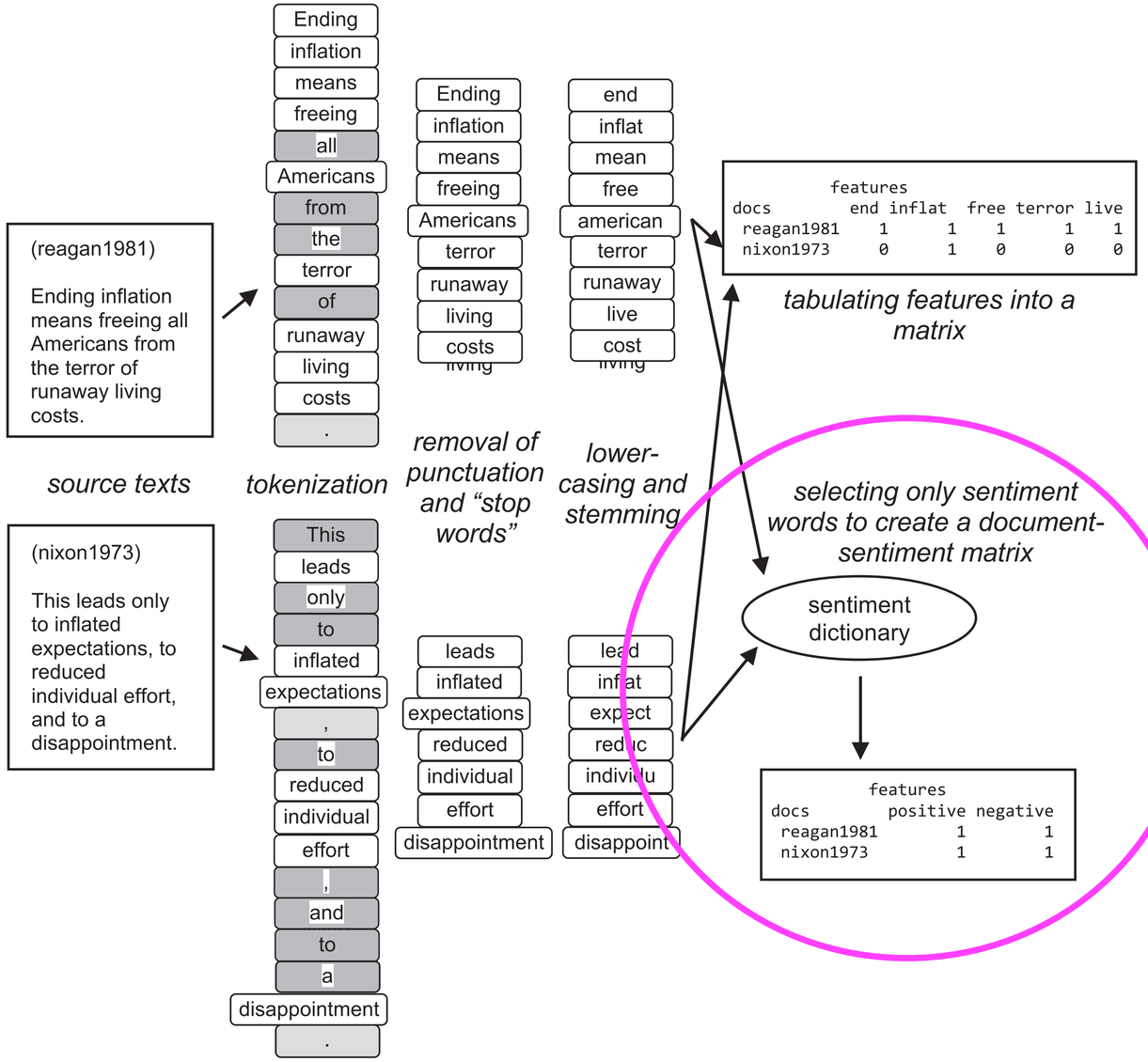


Figure 26.2 From text to tokens to matrix

Figure source: Benoit. 2020

Dictionary Methods

- Dictionary-based sentiment analysis is popular because it's easy, not because it's good.
- Consider this pair of sentences from JFK's optimism-filled inaugural speech, as coded by Lexicoder.

```
dfm("Let us never negotiate out of fear. But let us never fear to negotiate.",
    tolower = TRUE,    # casefold
    stem = FALSE,                  # do not stem
    remove_punct = TRUE,           # remove punctuation
    dictionary = data_dictionary_LSD2015)
```

```
Document-feature matrix of: 1 document, 4 features (75.0% sparse).
1 x 4 sparse Matrix of class "dfm"
       features
docs    negative positive neg_positive neg_negative
  text1        2        0            0            0
```

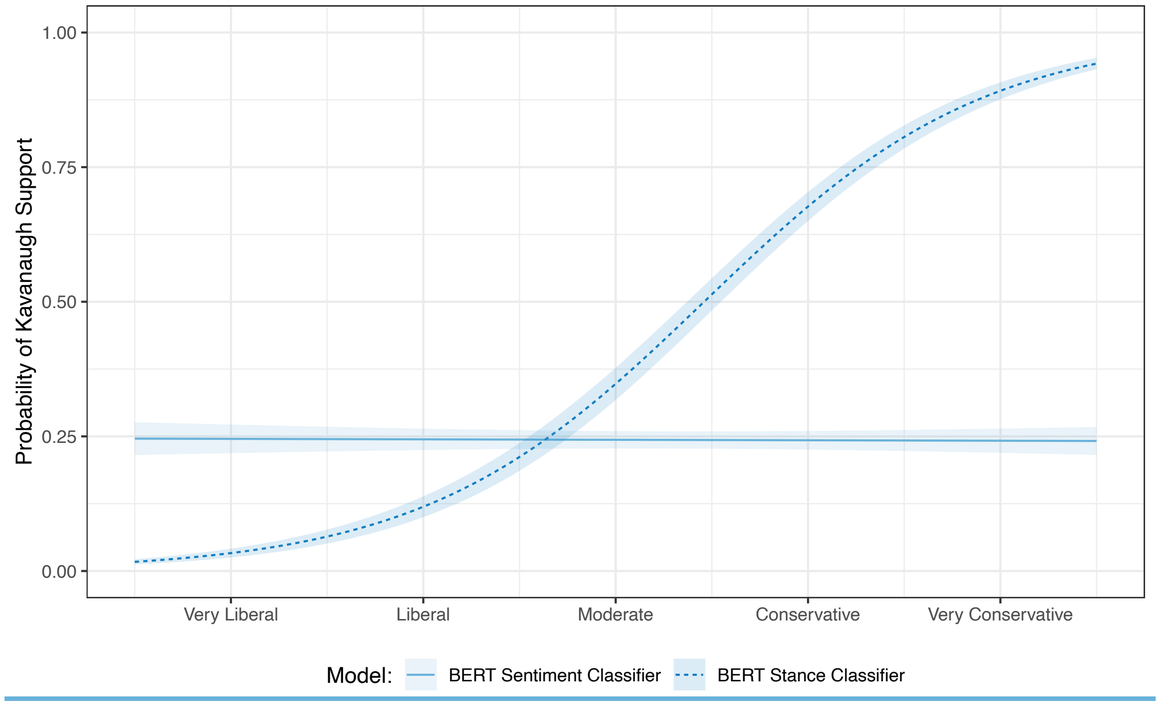
Supervised Learning - Classification

Table 6. Classifier Performance: Kavanaugh Tweets

Classifier	F1 Score (Predicting Sentiment)	F1 Score (Predicting Stance)
Lexicoder	0.788 (0.005)	0.572 (0.014)
VADER	0.754 (0.005)	0.514 (0.011)
SVM (sentiment-trained)	0.943 (0.003)	0.514 (0.012)
BERT (sentiment-trained)	0.954 (0.002)	0.582 (0.005)
SVM (stance-trained)		0.935 (0.006)
BERT (stance-trained)		0.938 (0.002)

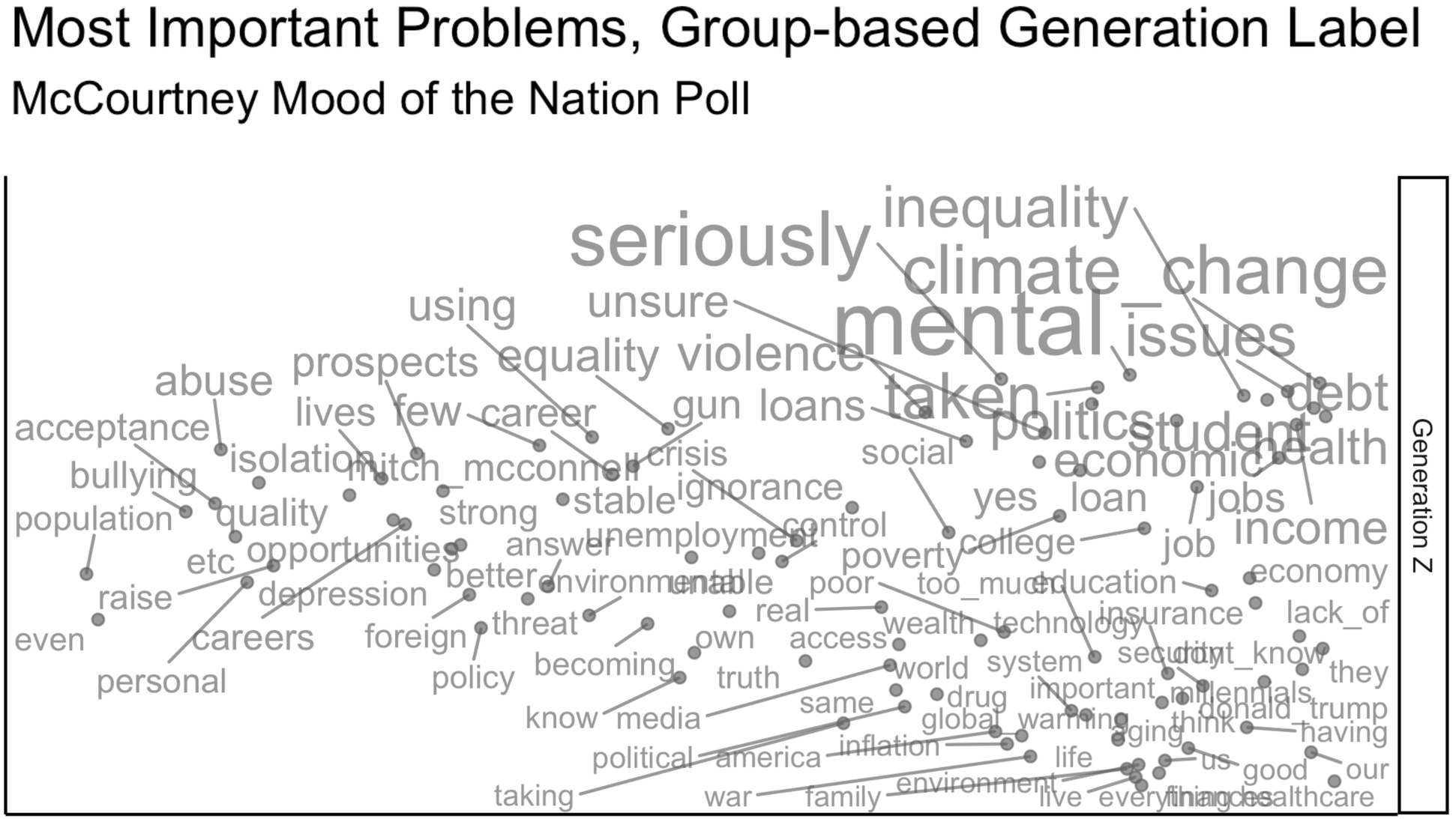
Reported figures are the average F1 score over 5-fold cross validation
Standard Errors in parentheses

Figure 5. Predicting Kavanaugh Support from Ideology

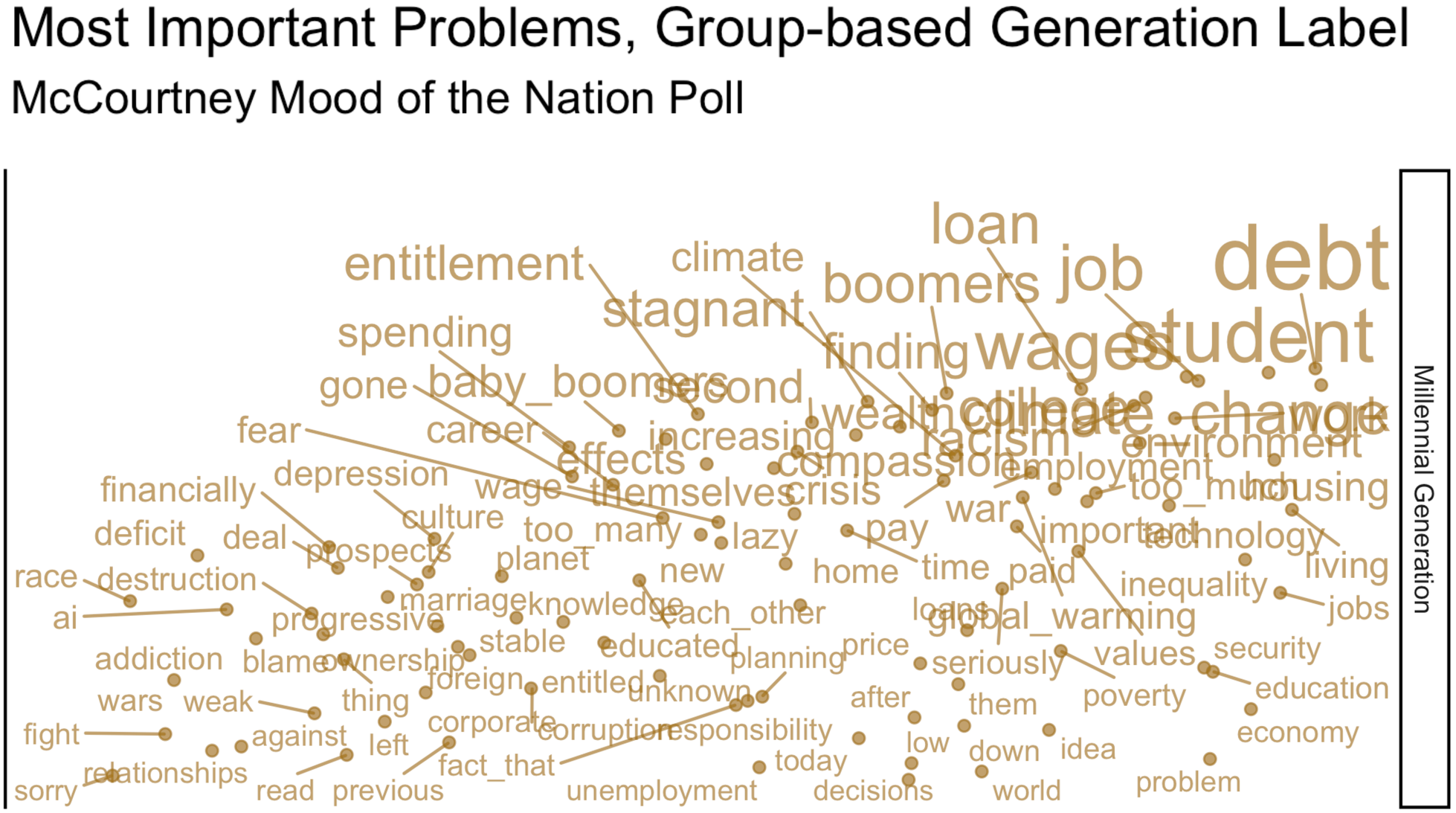


Source: Bestvater and Monroe, 2020.

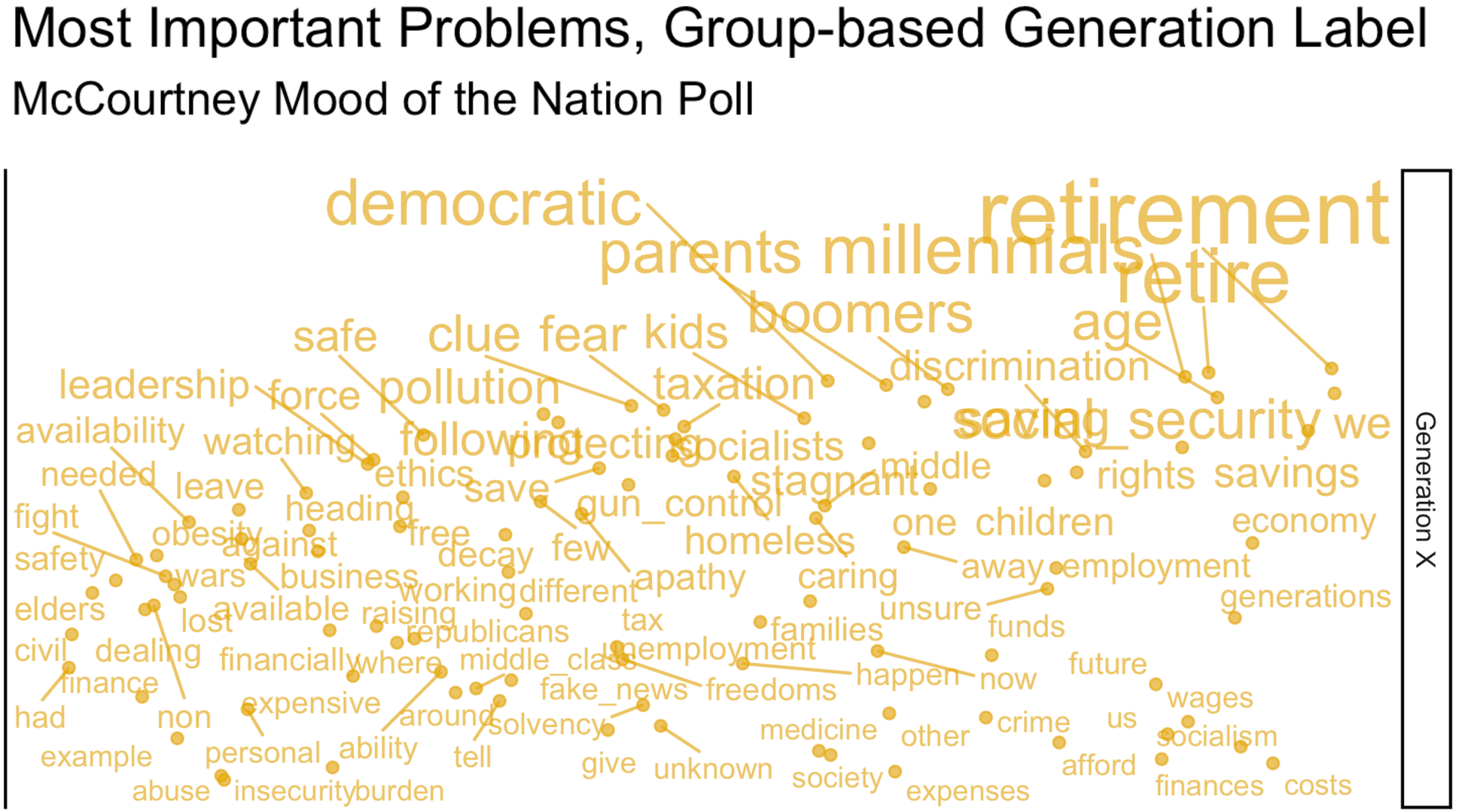
Interpreting differences in classes



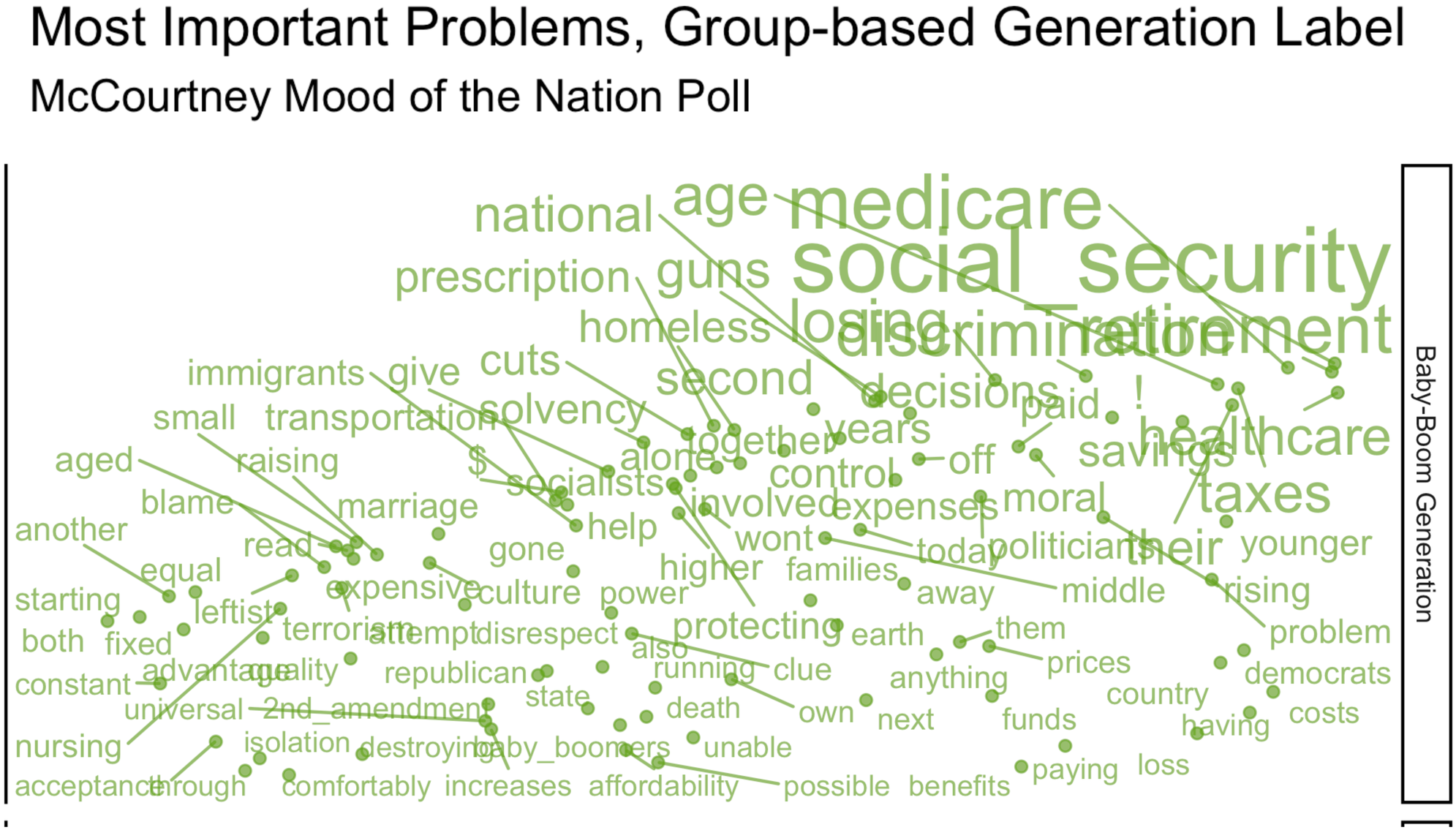
Interpreting differences in classes



Interpreting differences in classes



Interpreting differences in classes



Interpreting differences in classes

Unsupervised Learning

- We don’t have labeled examples (documents) — no training data.
- This is often used as an exploratory / discovery exercise or for theory testing.
- Some of the main variants:
  - Clustering
  - Topic modeling
  - Scaling (e.g., on a left-right scale from liberal to conservative)
  - Word embeddings

Unsupervised Learning - Topic Modeling

- Unlike a classification problem, you don't have pre-defined categories.
- But one main output of a topic model is a measure of how words are associated with the topic.
- When a topic model really works well, the appropriate label is obvious.
- A model of speeches in the US Senate contained the topic to the right … what label would you give it?

school
teacher
educ
student
children
test
local
learn
district
class
account
classroom
achiev
teach
better

Table 4: *Key Words on* [redacted]

Unsupervised Learning - Topic Modeling

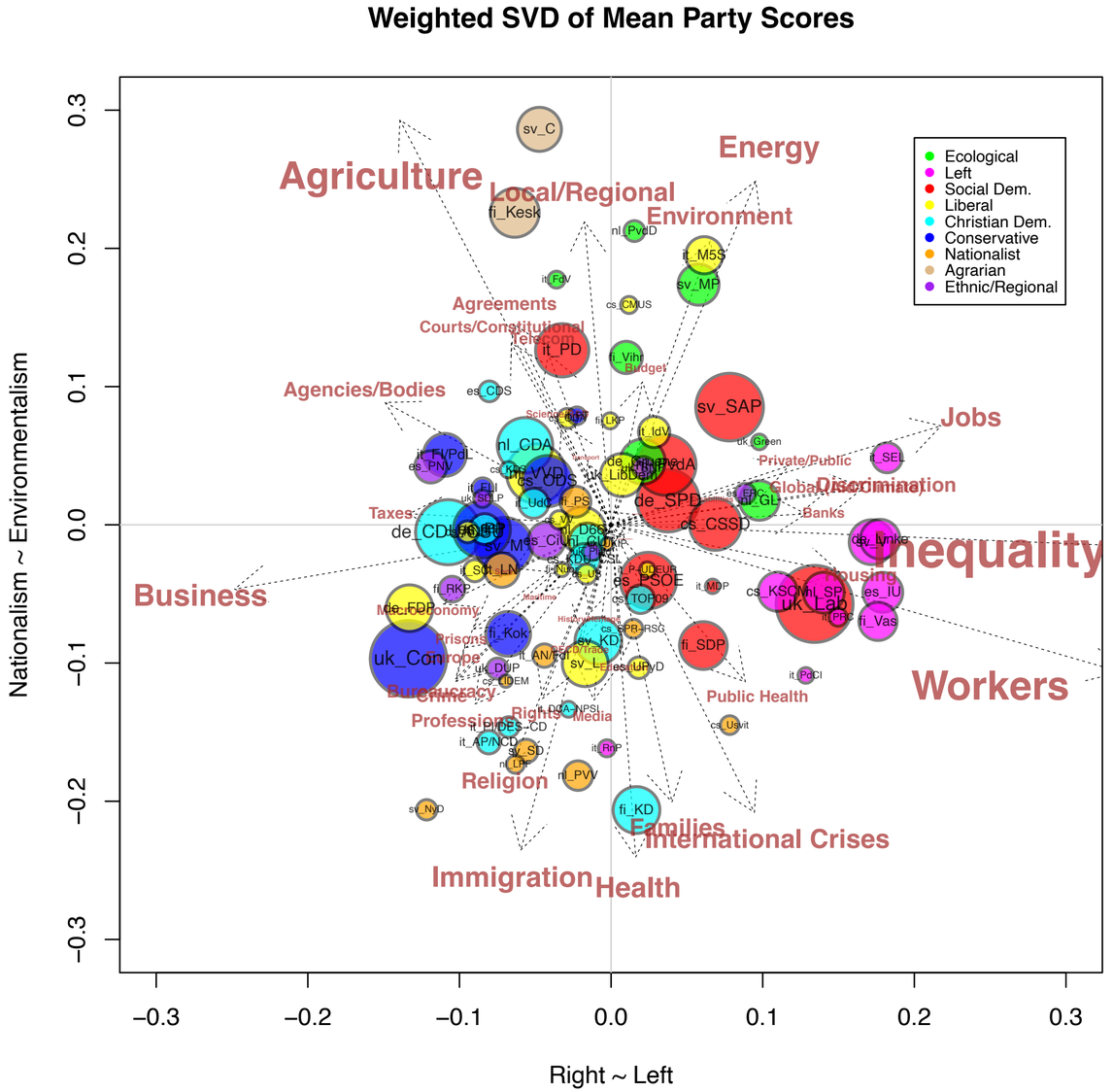
- Unlike a classification problem, you don't have pre-defined categories.
- But one main output of a topic model is a measure of how words are associated with the topic.
- When a topic model really works well, the appropriate label is obvious.
- A model of speeches in the US Senate contained the topic to the right … what label would you give it?

school
teacher
educ
student
children
test
local
learn
district
class
account
classroom
achiev
teach
better

Table 4: *Key Words on 'Education'.*

Unsupervised Learning - Scaling

- Here, the target isn't categories but a location on continuous dimensions.
- The plot to the right shows scaling locations of political parties in eight different parliaments (from text in eight different languages).
- The horizontal axis appears to be a traditional left-right dimension.
- The vertical axis appears to separate nationalist parties and environmentalist parties



Source: Goist and Monroe, 2018.

Natural Language Processing (NLP)

Tasks in NLP

- Some break “NLP” into “NLU”, “NLG,” and the rest (back to “NLP”). These overlap.

- Core “NLP” is often traditionally thought of as a pipeline for “annotation” or “sequence labeling.” Text flows through the connected segments, emerging from each with some added embellishment. Tokenization/segmentation, morphological analysis/lemmatization, POS-tagging, word sense disambiguation, syntactic parsing, semantic parsing (NLU?), named-entity recognition, coreference resolution (NLU?)

- NLU - Natural language understanding: extract information, translate raw text into a representation of meaning, reason about information given in the text. Classification, sentiment analysis, relationship extraction, reading comprehension, distributed representations/embeddings.

- NLG - Natural language generation: convert meaning representations into natural language text or text-to-text, produce appropriate utterances/responses in a dialog or Q&A (NLU?), summarize (NLU?), machine translate, style transfer, caption images (image-to-text).

Tasks in NLP, NLU, and NLG

- Some break “NLP” into “NLU”, “NLG,” and the rest (back to “NLP”). These overlap.

- Core “NLP” is often traditionally thought of as a pipeline for “annotation” or “sequence labeling.” Text flows through the connected segments, emerging from each with some added embellishment. Tokenization/segmentation, morphological analysis/lemmatization, POS-tagging, word sense disambiguation, syntactic parsing, semantic parsing (NLU?), named-entity recognition, coreference resolution (NLU?) There are a lot of objectives overlapping with “text-as-data” in there.

- NLU - Natural language understanding: extract information, translate raw text into a representation of meaning, reason about information given in the text. Classification, sentiment analysis, relationship extraction, reading comprehension, distributed representations/embeddings.

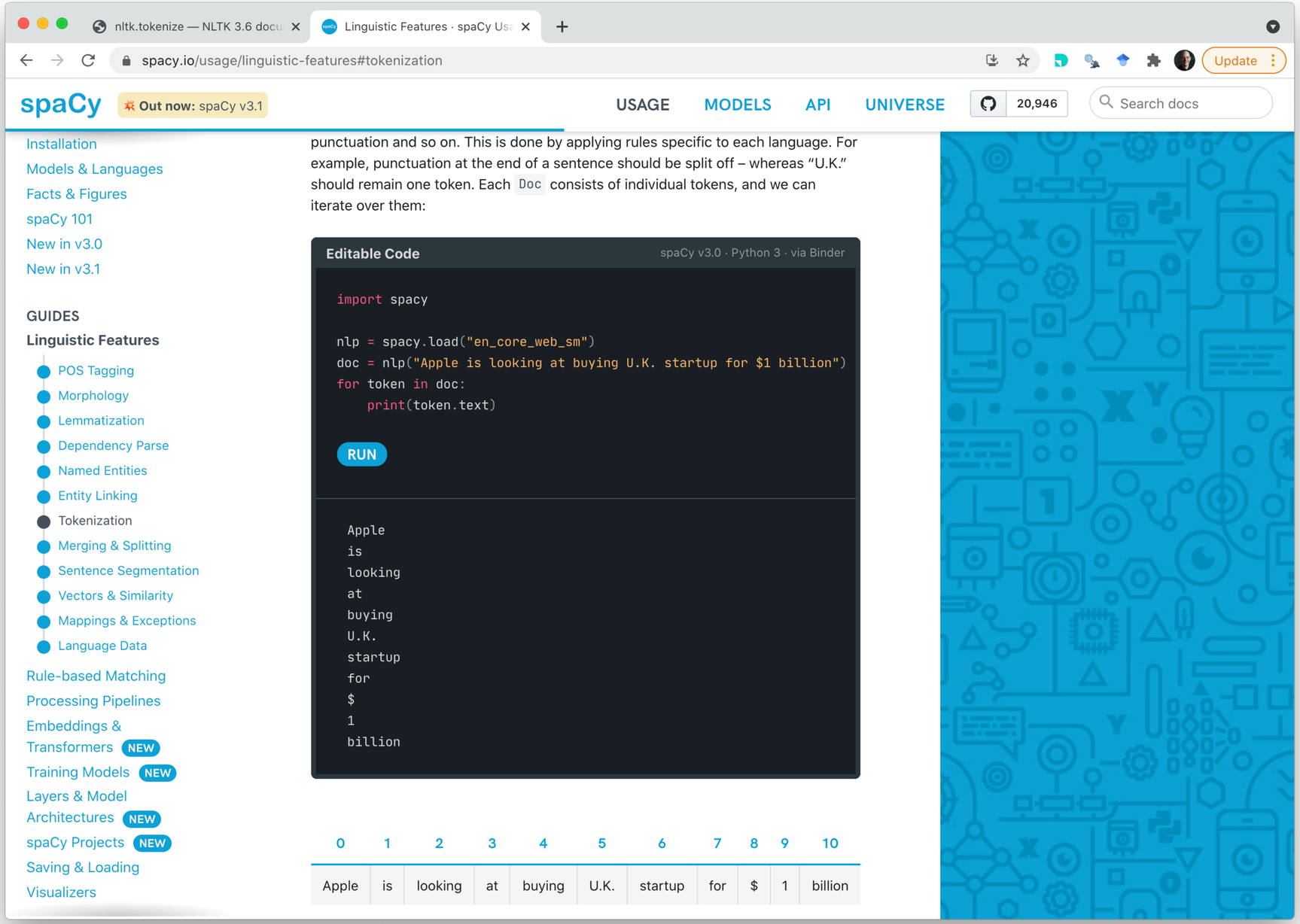
- NLG - Natural language generation: convert meaning representations into natural language text or text-to-text, produce appropriate utterances/responses in a dialog or Q&A (NLU?), summarize (NLU?), machine translate, style transfer, caption images (image-to-text).

Online demos of a handful of NLP tasks (feel free to poke around).

How much “understanding” is happening?

Tokenization

https://spacy.io/usage/linguistic-features

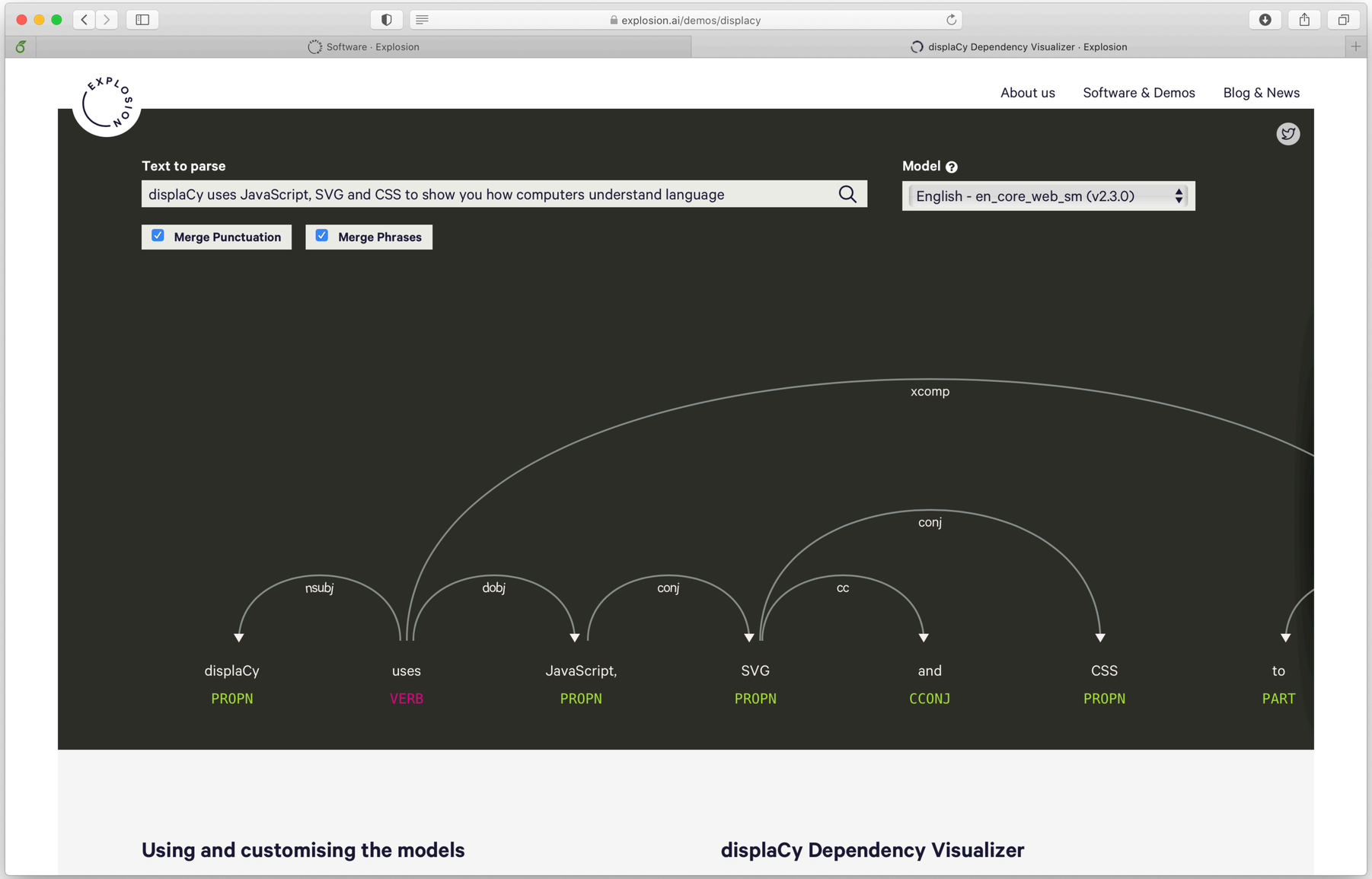


Annotation: named entity recognition

https://explosion.ai/demos/displacy-ent

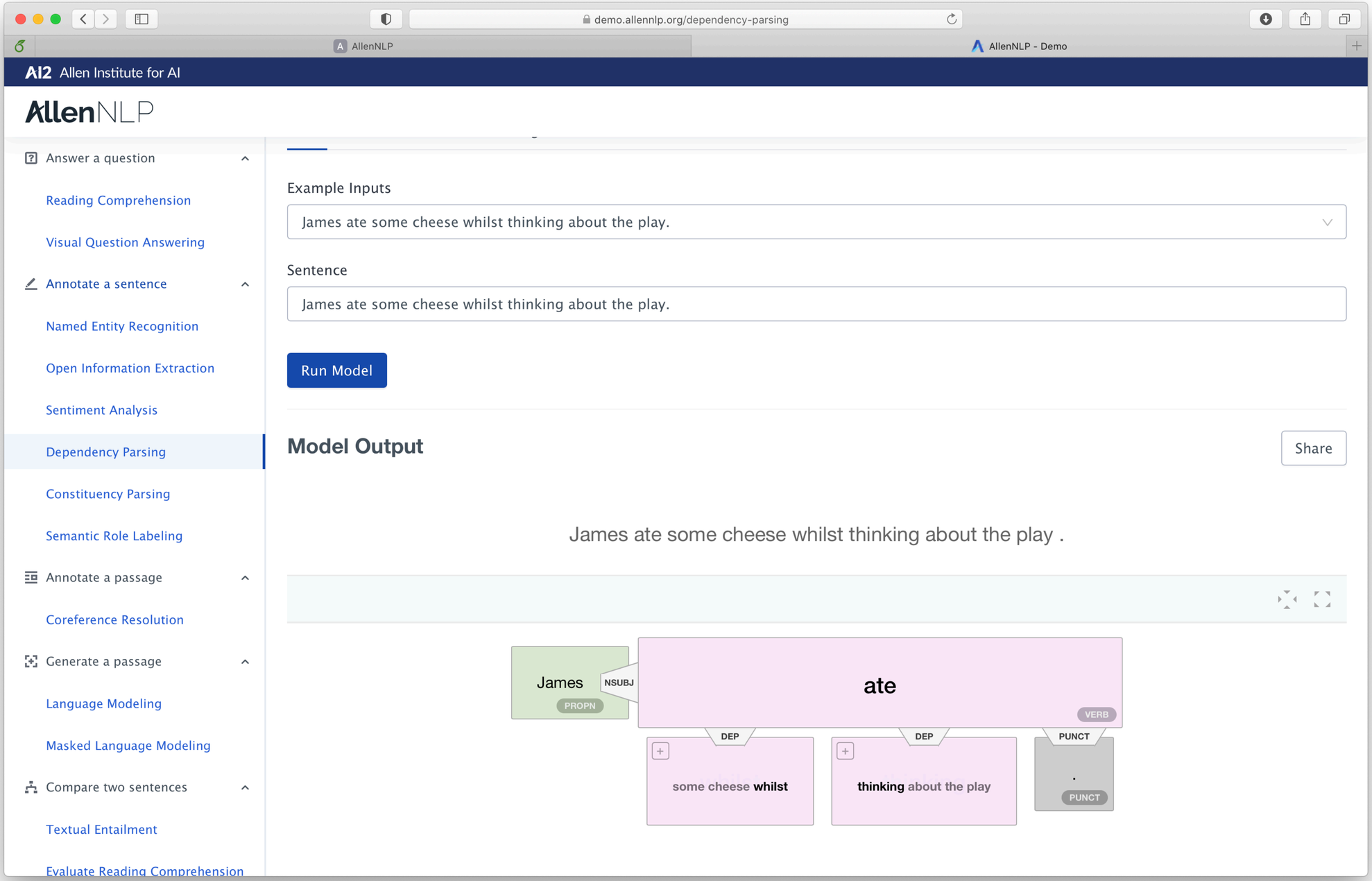
Dependency parsing

https://explosion.ai/demos/displacy



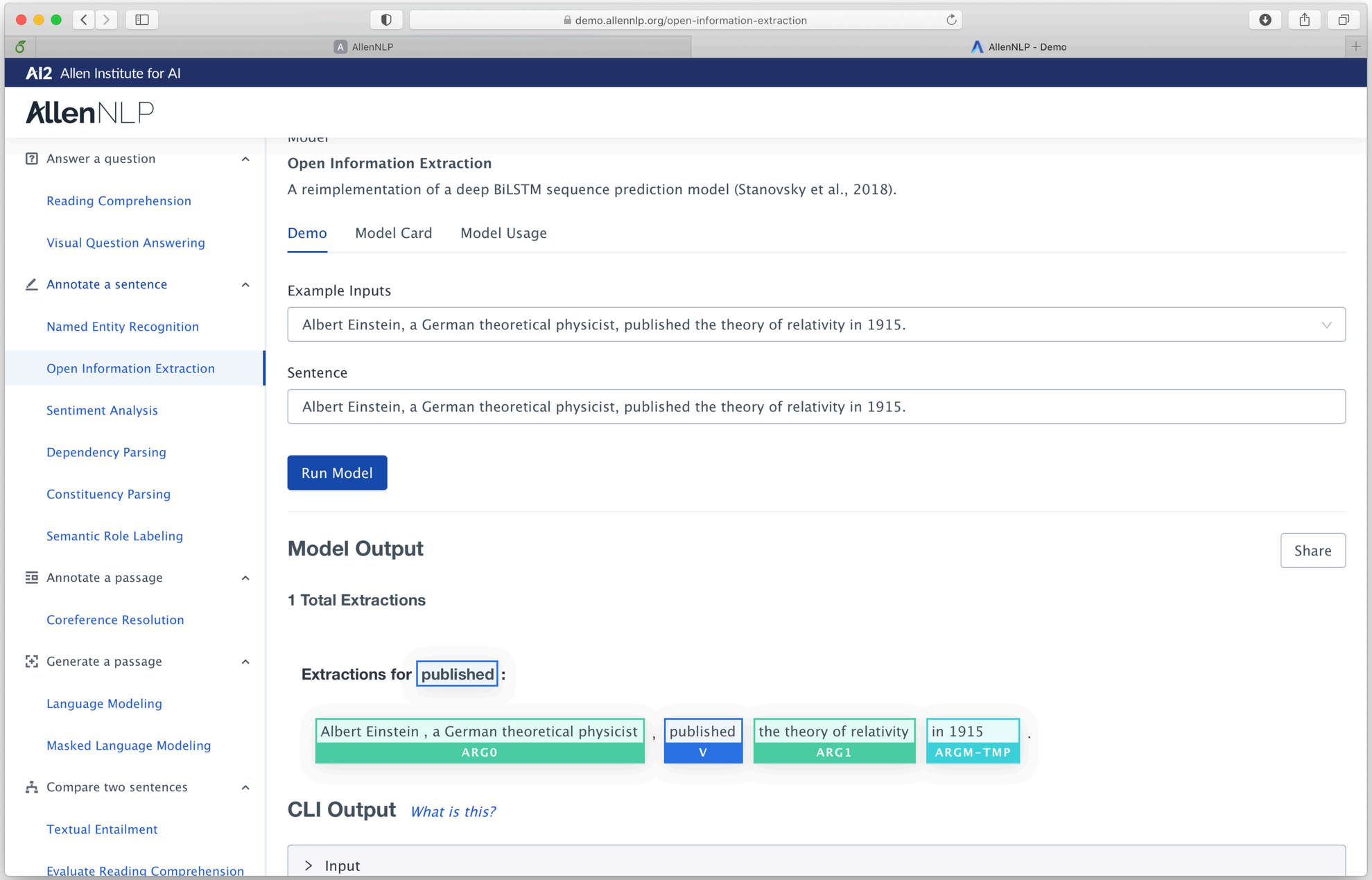
Dependency parsing

https://demo.allennlp.org



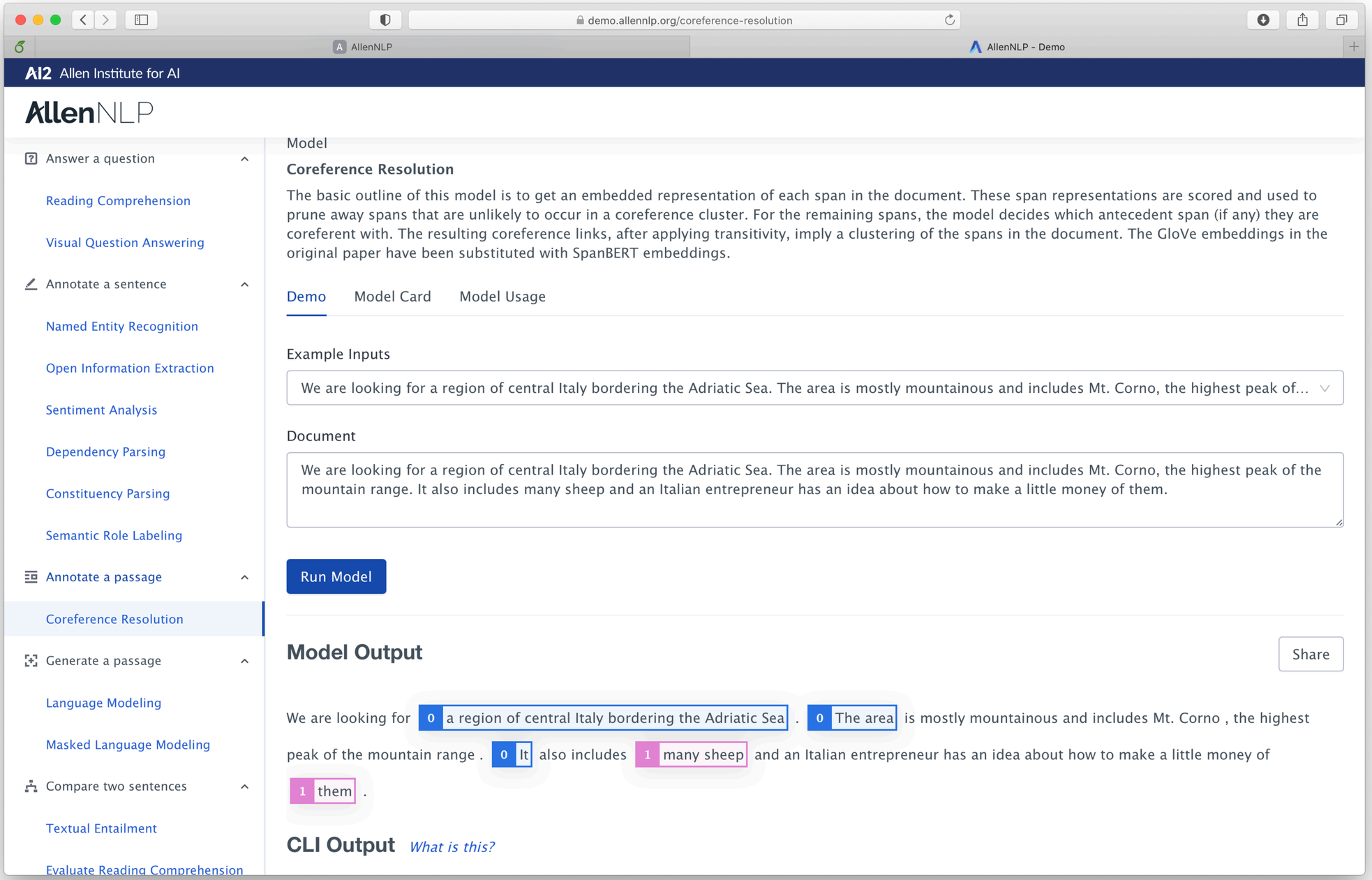
Information extraction

https://demo.allennlp.org



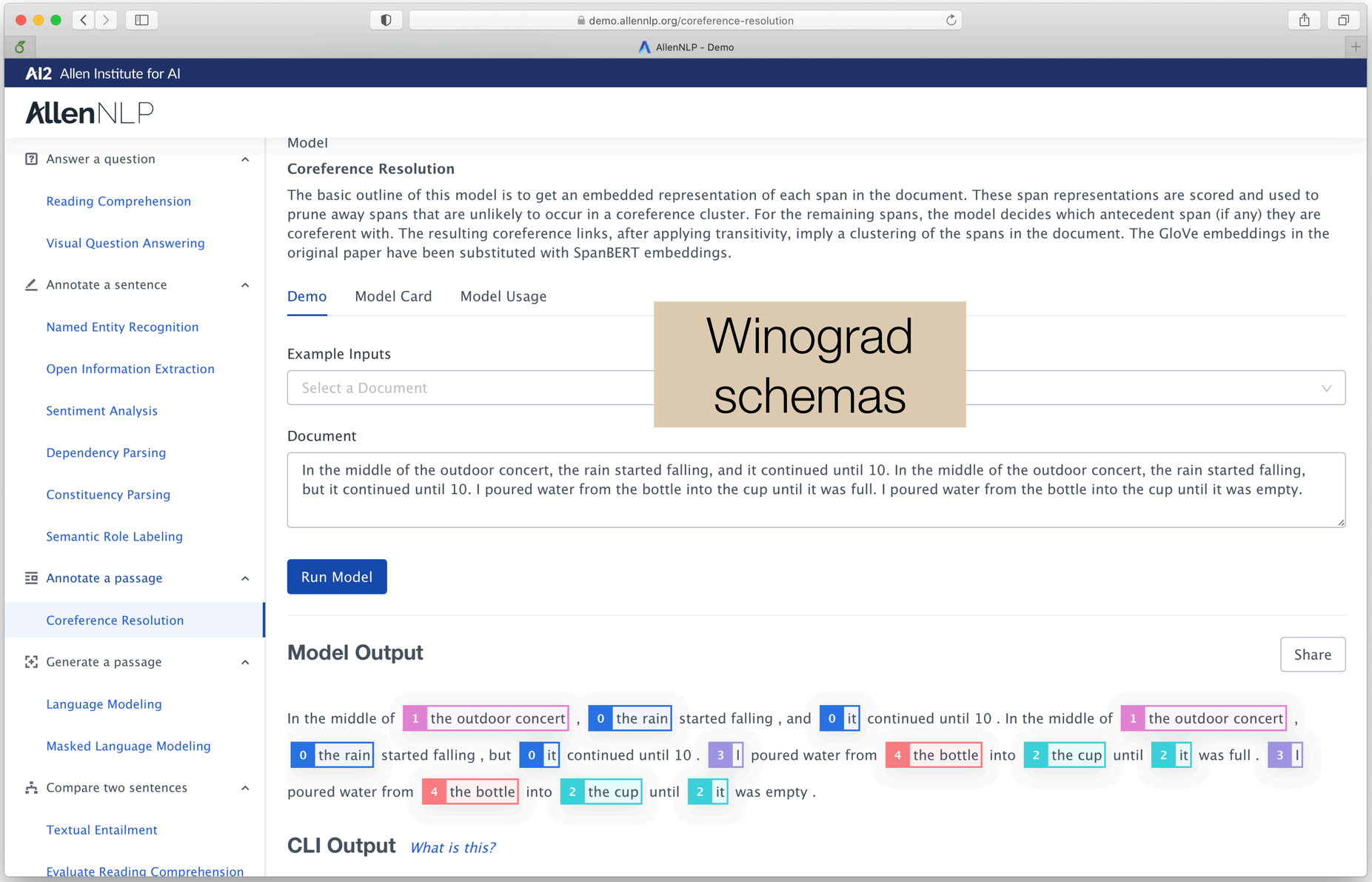
Coreference resolution: annotation?

https://demo.allennlp.org



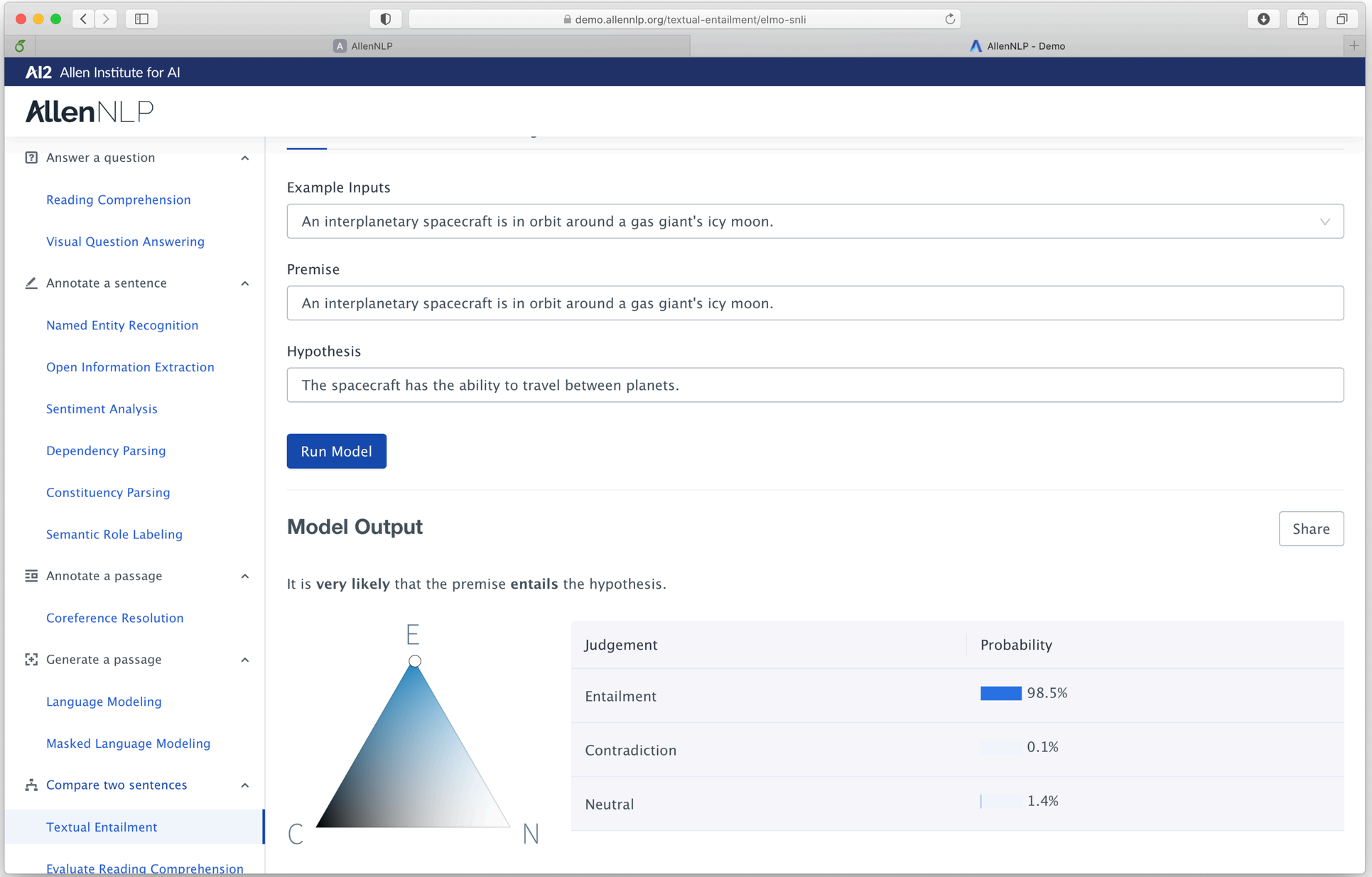
Coreference resolution: NLU?

https://demo.allennlp.org



Textual Entailment

https://demo.allennlp.org



The path we'll take through NLP

- Traditionally, abstract NLP tasks were preceded by pipeline annotation. This differs from, or is only loosely approximated by, the conventional "preprocessing" we've seen in text-as-data tasks. Tomorrow (DAY 2) we'll explore what those pipelines look like, what problems they solve, and how/why we might incorporate them into our text-as-data analyses.

- Day 3 will focus on an area that overlaps TADA and NLP: word embeddings. Word embeddings share a lot of similarities with unsupervised topic model and scaling methods.

- Contemporary NLP, as with AI generally, is dominated now by neural network / deep learning approaches. These often sidestep the traditional NLP pipeline, using minimally preprocessed tokenized text as inputs and training directly on the task at hand, for "end-to-end" analysis. We'll spend most of our time, from Day 4 to the end, on neural NLP.

Tomorrow

- NLP Pipelines / core NLP tasks
  - Tokenization / segmentation
  - Normalization / lemmatization / stemming / morphology
  - Sequence labeling — part-of-speech tagging, named entity recognition
  - Dependency parsing
- Demos: NLP pipelines in R and Python

Software / demos

- Intro to RStudio Cloud
- Intro to Google Colab
- We'll look (briefly) at some R and Python notebooks covering TADA material that you would have seen, more or less, in a prior class.
  - Log in to https://rstudio.cloud via SSO with your Essex account. The class should be open to you and the project for Day 1 should be accessible as well.
  - Google Colab notebook linked from course information page: https://burtmonroe.github.io/TextAsDataCourse/Essex/