Workshop: social media and text analysis applied to the study of international courts

Pablo Barberá
Center for Data Science
New York University

March 30, 2016

materials: github.com/pablobarbera/icourts-workshop

Overview of text as data methods

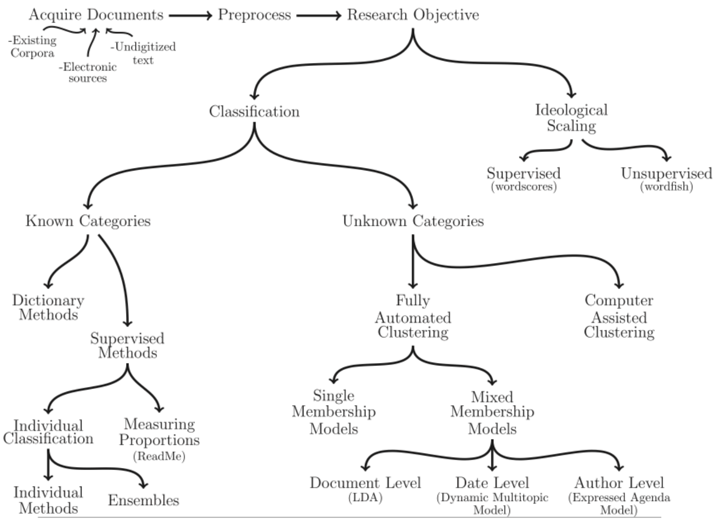


Fig. 1 in Grimmer and Stewart (2013)

Overview of text as data methods

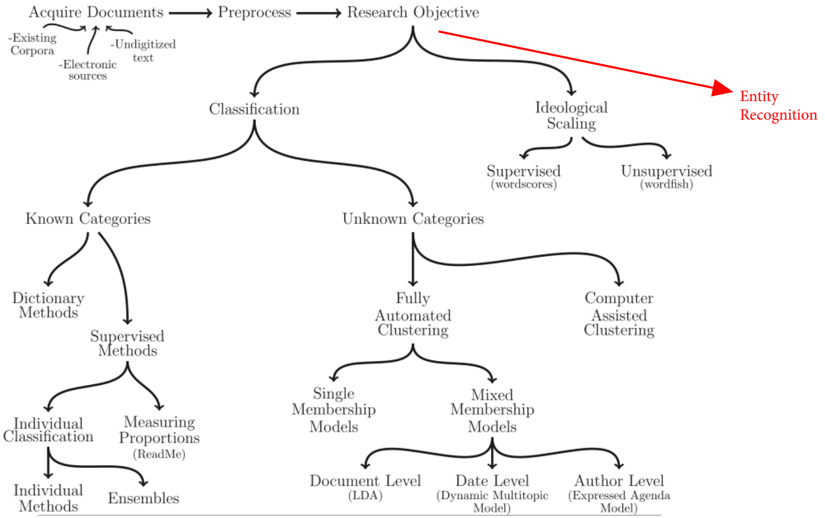


Fig. 1 in Grimmer and Stewart (2013)

Overview of text as data methods

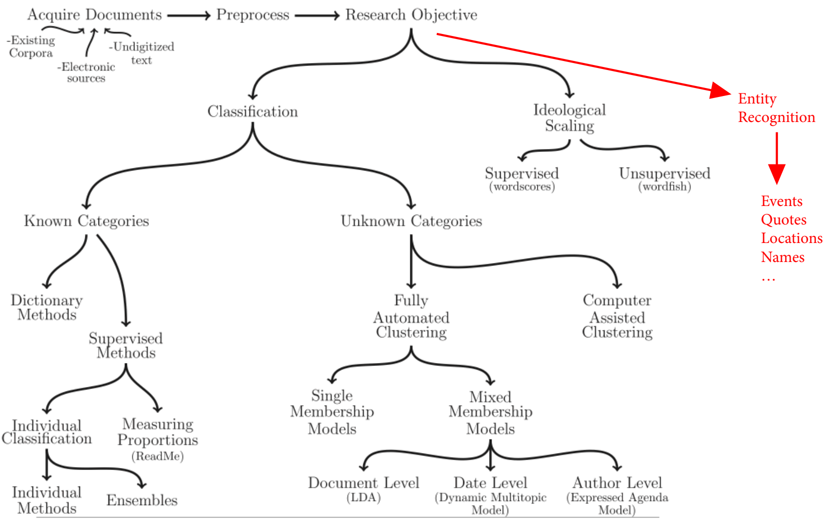


Fig. 1 in Grimmer and Stewart (2013)

Overview of text as data methods

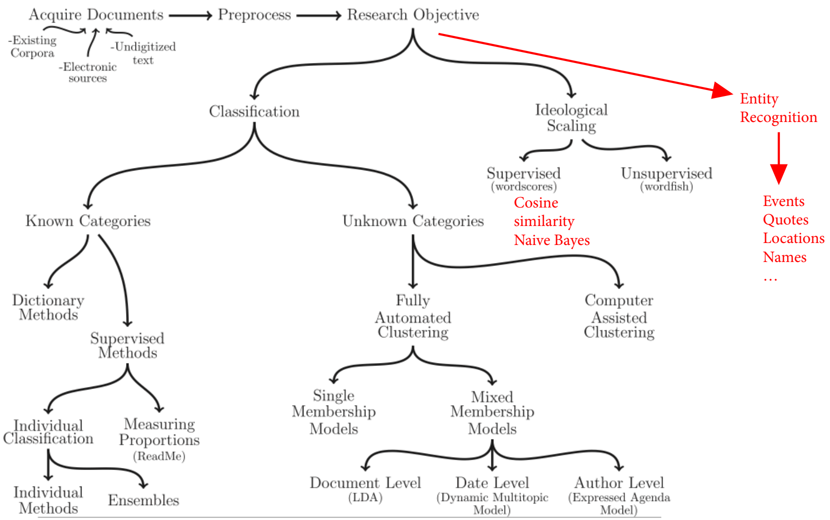


Fig. 1 in Grimmer and Stewart (2013)

Overview of text as data methods

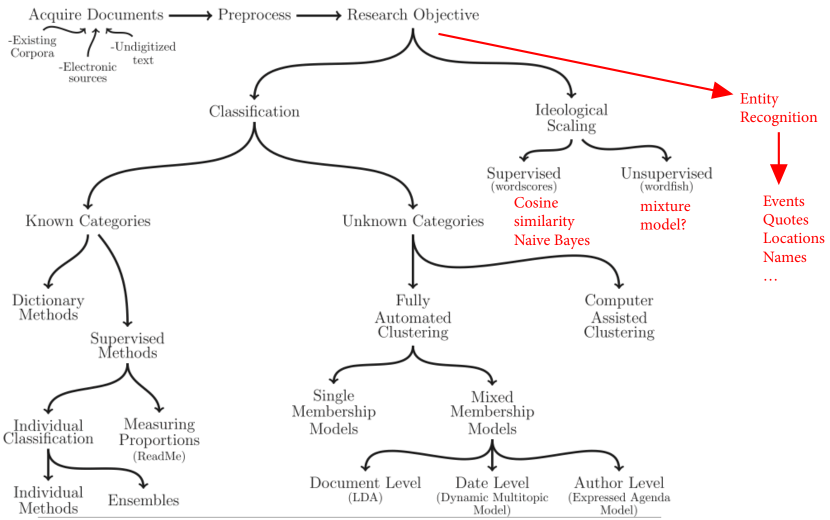


Fig. 1 in Grimmer and Stewart (2013)

Overview of text as data methods

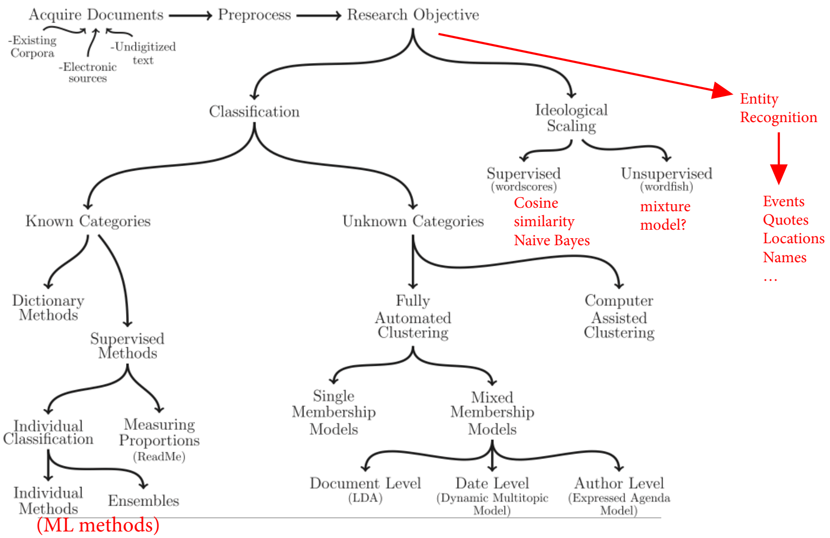


Fig. 1 in Grimmer and Stewart (2013)

Overview of text as data methods

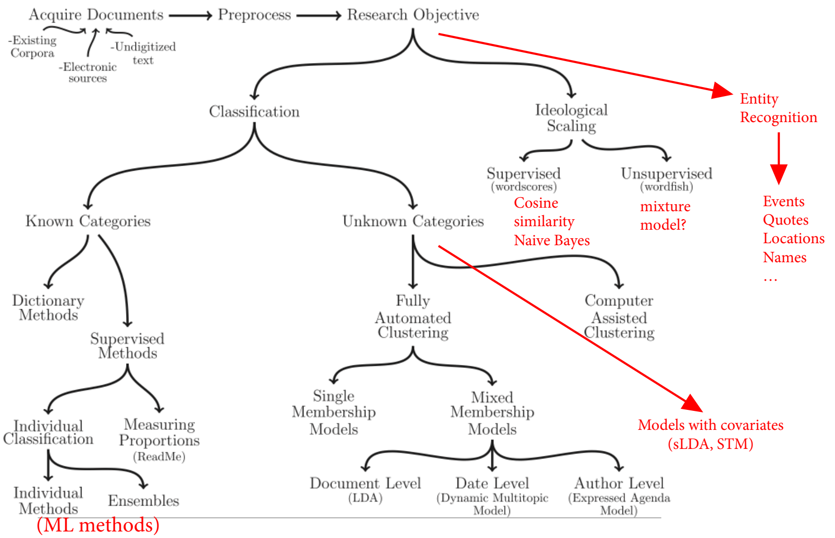


Fig. 1 in Grimmer and Stewart (2013)

Text preprocessing

From words to numbers

1. Preprocess text:

“@MEPcandidate thank you and congratulations, you’re the best #EP2014”

“@MEPcandidate You’re an idiot, I would never vote for you”

Text preprocessing

From words to numbers

1. Preprocess text: lowercase,

“@mepcandidate thank you and congratulations, you’re the best #ep2014”

“@mepcandidate you’re an idiot, i would never vote for you”

Text preprocessing

From words to numbers

1. Preprocess text: lowercase, remove stopwords and punctuation,

“@mepcandidate thank ~~you~~ ~~and~~ congratulations~~,~~ you’re ~~the~~ best #ep2014”

“@mepcandidate you’re ~~an~~ idiot~~,~~ ~~i~~ ~~would~~ never vote ~~for~~ ~~you~~”

Text preprocessing

From words to numbers

1. Preprocess text: lowercase, remove stopwords and punctuation, stem,

“@ thank congratul~~ations~~, you’r~~e~~ best #ep2014”

“@ you’r~~e~~ idiot never vote”

Text preprocessing

From words to numbers

1. Preprocess text: lowercase, remove stopwords and punctuation, stem, tokenize into unigrams and bigrams (bag-of-words assumption)

   [@, thank, congratul, you'r, best, #ep2014, @ thank, thank congratul, congratul you'r, you'r best, best, best #ep2014]

   [@, you'r, idiot, never, vote, @ you'r, you'r idiot, idiot never, never vote]

Text preprocessing

From words to numbers

1. Preprocess text: lowercase, remove stopwords and punctuation, stem, tokenize into unigrams and bigrams (bag-of-words assumption)

   [@, thank, congratul, you'r, best, #ep2014, @ thank, thank congratul, congratul you'r, you'r best, best, best #ep2014]

   [@, you'r, idiot, never, vote, @ you'r, you'r idiot, idiot never, never vote]

2. Document-term matrix:
   - **W**: matrix of N documents by M unique words
   - W_{im}= number of times m-th words appears in i-th document.

	@	thank	congratul	you'r	#ep2014	@ thank	…	M words
Document 1	1	1	1	1	1	1	…	
Document 2	1	0	0	1	0	0	…	
…								
Document n n	0	1	1	0	0	0	…	

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods
    - Dictionary of positive and negative words (issue-specific)

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods
    - Dictionary of positive and negative words (issue-specific)
    - Count number of times they appear

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods
    - Dictionary of positive and negative words (issue-specific)
    - Count number of times they appear
2. Machine learning

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods
   - Dictionary of positive and negative words (issue-specific)
   - Count number of times they appear
2. Machine learning
   - Training data: random sample of posts manually labelled as positive or negative

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods
   - Dictionary of positive and negative words (issue-specific)
   - Count number of times they appear
2. Machine learning
   - Training data: random sample of posts manually labelled as positive or negative
   - *Classifier* learns from data and predicts sentiment of unseen posts

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods
   - Dictionary of positive and negative words (issue-specific)
   - Count number of times they appear
2. Machine learning
   - Training data: random sample of posts manually labelled as positive or negative
   - *Classifier* learns from data and predicts sentiment of unseen posts
   - Many different classifiers: regularized logistic regression, Naive Bayes, SVM, BART, ensemble methods...

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods
   - Dictionary of positive and negative words (issue-specific)
   - Count number of times they appear
2. Machine learning
   - Training data: random sample of posts manually labelled as positive or negative
   - *Classifier* learns from data and predicts sentiment of unseen posts
   - Many different classifiers: regularized logistic regression, Naive Bayes, SVM, BART, ensemble methods...
   - Accuracy, precision, recall, test data, cross-validation.

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

Two main approaches:

1. Dictionary methods
   - Dictionary of positive and negative words (issue-specific)
   - Count number of times they appear
2. Machine learning
   - Training data: random sample of posts manually labelled as positive or negative
   - *Classifier* learns from data and predicts sentiment of unseen posts
   - Many different classifiers: regularized logistic regression, Naive Bayes, SVM, BART, ensemble methods...
   - Accuracy, precision, recall, test data, cross-validation.

Challenges: sarcasm, intensity, subject identification.

Sentiment analysis

Sentiment, tone, valence, affective score... = % positive messages about a subject

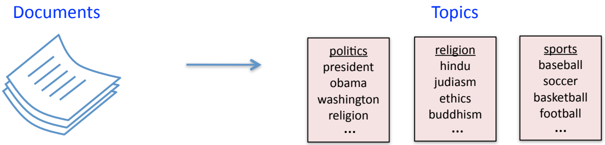
Two main approaches:

1. Dictionary methods
   - Dictionary of positive and negative words (issue-specific)
   - Count number of times they appear
2. Machine learning
   - Training data: random sample of posts manually labelled as positive or negative
   - *Classifier* learns from data and predicts sentiment of unseen posts
   - Many different classifiers: regularized logistic regression, Naive Bayes, SVM, BART, ensemble methods...
   - Accuracy, precision, recall, test data, cross-validation.

Challenges: sarcasm, intensity, subject identification.
In practice, accuracy over 80% is impossible (and that's ok)

Latent Dirichlet allocation (LDA)

- **Topic models** are powerful tools for exploring large data sets and for making inferences about the content of documents



Latent Dirichlet allocation (LDA)

- **Topic models** are powerful tools for exploring large data sets and for making inferences about the content of documents

Documents

Topics

politics	religion	sports
president	hindu	baseball
obama	judiasm	soccer
washington	ethics	basketball
religion	buddhism	football
...	...	...

- Many applications in information retrieval, document summarization, and classification

New document

Words $w_1, \ldots, w_N$

What is this document about?

weather	.50
finance	.49
sports	.01

Distribution of topics θ

Latent Dirichlet allocation (LDA)

- **Topic models** are powerful tools for exploring large data sets and for making inferences about the content of documents

Documents

Topics

politics	religion	sports
president	hindu	baseball
obama	judiasm	soccer
washington	ethics	basketball
religion	buddhism	football
...	...	...

- Many applications in information retrieval, document summarization, and classification

New document

Words $w_1, \ldots, w_N$

What is this document about?

weather	.50
finance	.49
sports	.01

Distribution of topics θ

- LDA is one of the simplest and most widely used topic models

Latent Dirichlet Allocation

- Document = random mixture over latent topics
- Topic = distribution over n-grams

Probabilistic model with 3 steps:

1. Choose $\theta_i \sim \text{Dirichlet}(\alpha)$
2. Choose $\beta_k \sim \text{Dirichlet}(\delta)$
3. For each word in document i:
    - Choose a topic $z_m \sim \text{Multinomial}(\theta_i)$
    - Choose a word $w_{im} \sim \text{Multinomial}(\beta_{i,k=z_m})$

where:

α=parameter of Dirichlet prior on distribution of topics over docs.

θ_i=topic distribution for document i

δ=parameter of Dirichlet prior on distribution of words over topics

β_k=word distribution for topic k