SOSC 4300/5500: Text Analysis; Supervised Machine Learning

Han Zhang

Outline

Logistics

- Begin to think about literature review and final projects?
- Assignment 1. Questions?

Review: text classification

- Goal: classify documents into pre existing categories
  - e.g. sentiment of tweets, ideological position of politicians, whether an email is a spam email or not
- We have seen how dictionary method works
- But dictionary methods have many shortcomings
  - Polysemy
  - Many arbitrary decisions in constructing dictionary

Supervised Machine Learning

- We use supervised machine learning to classify documents into categories
- Typically setting: there is 1M documents that we want to get their labels (e.g. pos or neg)
  - But we do not have time / resources to read every document and decide their labels
- Training data: a sample of documents with labels (e.g., sentiment labels), typically much smaller than the size of entire corpus (e.g., 10,000 documents)
  - We further split the training data into:
    - training
    - validation; to select the tuning parameter
- Apply the trained ML algorithm on the rest (1M - 10,000) data
  - to generate predictions as the measurement

A social science example

- Tamar Mitts, *From Isolation to Radicalization: Anti-Muslim Hostility and Support for ISIS in the West*, American Political Science Review **113** (2019), no. 1, 173–194
- Research question: do offline anti-Muslim intensity increase online pro-ISIS attitudes
- Independent variable: local-level voting share for the far-right parties in Western European countries
- Dependent variable: local-level percentage of tweets that are pro-ISIS
- Data collection: followers of ISIS-affiliated accounts who live in Western Europe
- Problem: over 100M tweets posted by these users; need a quick way to identify
  - which tweet is pro-ISIS
  - which is not

Mitts, examples

TABLE 2. Examples of Tweets in Different Topics

Sympathy with ISIS

1. Jihad is the greatest of all deeds #IslamicState
2. Show everything from the Islamic State and other groups in Syria. It's important to hear all sides of the story
3. Assalam o Alaikom to All Islamic State Brothers
4. In sha Allah we will have honor again #IslamicState

Procedures

- Random sample tweets in 4 languages: English (9926), Arabic (10631), French (6158) and German (3011);
- Artificially created balanced training data: roughly half of them are pro-ISIS and the other half are not relevant to ISIS-related topics
  - Recruit over 1,000 coders (crowd sourcing approach; more later)
  - Each coder is assigned to three tweets, and label as
    - pro-ISIS
    - not relevant
  - Ultimate label is the majority decision of the three coders
- Train an Elastic Net (LASSO + Ridge) on the training data
  - use cross-validation to select tuning parameters (λ)
- Apply the trained model on the most unlabelled data

Procedures

- Ultimately, get a measure based on the Elastic Net's prediction of whether the post is pro-ISIS or not
- And use this measure to answer her question: does offline anti-Muslim intensity increase online pro-ISIS attitudes?
  - Through regression
  - Dependent variable is pro-ISIS attitudes (from ML predictions)
  - Independent variable is offline anti-Muslim intensity (from voting)
- This is a good example of using ML predictions for measurement

Exercises:

- How can you use dictionary methods to study the same research question: do offline anti-Muslim intensity increase online pro-ISIS attitudes
- Recall: we need to generate measures for the dependent variable: local-level percentage of tweets that are pro-ISIS

Supervised Machine Learning vs Dictionary Method

- Supervised machine learning can be conceptualized as a generalization of dictionary methods
  - Think about document-term matrix
  - Dictionary methods basically only keeps the columns whose words are in the dictionary;
  - Supervised methods keeps all words, and learn the weights of each column from data
    - Irrelevant words will then be assigned with lower weights
- Theoretically, supervised machine learning will outperform dictionary methods in classification tasks, as long as training set is large enough

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

Supervised Machine Learning vs Dictionary Method

- Dictionary Method; if you use an off-the-shelf dictionary
  - Advantage: not corpus-specific, cost to apply to a new corpus is trivial
  - Disadvantage: not corpus-specific ; performance will be bad
- Dictionary Method; if you construct your own dictionary
  - Advantage: performance will be better than using existing dictionary
  - Disadvantage: time cost
- Supervised learning
  - Advantage: performance will theoretically be the best
  - Disadvantage: time cost

Steps in supervised methods

Supervised	Dictionary
Collect corpus and preprocess	Collect corpus and preprocess
Collect training data	Find/construct dictionaries
train algorithms	Apply dictionary on corpus
Validation	Validation
Apply trained algorithm	

Various algorithms you will commonly used

- We have introduced how do you transform text data into a matrix X in the last lecture
- And we have labels (Y)
- Then we can use the algorithms that you have used for Assignment 1 to make prediction for texts
  - Linear/logistic regression
  - LASSO and Elastic Net: linear regression
  - Tree and Forests
  - SVM

Named entity recognition

- Named entity recognition (NER) is a specific supervised task that aims to recognize proper nouns
- Input: texts
- Output: a list of named entiries, such as
  - name of people
  - name of organizations
  - name of countries/cities
  - dates

NER example

- using Spacy package

F.B.I. Agent Peter Strzok PERSON , Who Criticized Trump PERSON in Texts, Is Fired GPE - The New York Times ORG SectionsSEARCHSkip to contentSkip to site indexPoliticsSubscribeLog InSubscribeLog InToday's PaperAdvertisementSupported ORG byF.B.I. Agent Peter Strzok PERSON , Who Criticized Trump PERSON in Texts, Is FiredImagePeter Strzok, a top F.B.I. GPE counterintelligence agent who was taken off the special counsel investigation after his disparaging texts about President Trump PERSON were uncovered, was fired. CreditT.J. Kirkpatrick PERSON for The New York TimesBy Adam Goldman ORG and Michael S. SchmidtAug PERSON . 13 CARDINAL , 2018WASHINGTON CARDINAL — Peter Strzok PERSON , the F.B.I. GPE senior counterintelligence agent who disparaged President Trump PERSON in inflammatory text messages and helped oversee the Hillary Clinton PERSON email and Russia GPE investigations, has been fired for violating bureau policies, Mr. Strzok PERSON 's lawyer said Monday DATE .Mr. Trump and his allies seized on the texts — exchanged during the 2016 DATE campaign with a former F.B.I. GPE lawyer, Lisa Page — in PERSON assailing the Russia GPE investigation as an illegitimate "witch hunt." Mr. Strzok PERSON , who rose over 20 years DATE at the F.B.I. GPE to become one of its most experienced counterintelligence agents, was a key figure in the early months DATE of the inquiry.Along with writing the texts, Mr. Strzok PERSON was accused of sending a highly sensitive search warrant to his personal email account.The F.B.I. GPE had been under immense political pressure by Mr. Trump PERSON to dismiss Mr. Strzok PERSON , who was removed last summer DATE from the staff of the special counsel, Robert S. Mueller III PERSON . The president has repeatedly denounced Mr. Strzok PERSON in posts on Twitter EVENT , and on Monday DATE expressed satisfaction that he had been sacked.Mr. Trump's ORG victory traces back to June DATE , when Mr. Strzok PERSON 's conduct was laid out in a wide-ranging inspector general's report on how the F.B.I. GPE handled the investigation of Hillary Clinton's PERSON emails in the run-up to the 2016 DATE election. The report was critical of Mr. Strzok PERSON 's conduct in sending the

Two different solutions

- Dictionary methods (rule-based methods)
  - Have a dictionary of names that you want to capture
  - Drawback: if the NER you want to capture are constantly changing, needs to constantly update the dictionary
- ML approach: collect training data -> train algorithms -> apply algorithms on new data
- E.g., for Chinese names:
  - training data: a corpus that contains:
    - existing names you can get (positive examples)
    - other two or three character words that are not names (negative examples)
  - More complex than the document classification tasks we have just learned, because:
  - One document may have multiple outcomes (from 0 organization names to many)

NER in practice

- For common entities (such as names for people, countries), there are many mature packages that deal with the task
  - For English, NLTK, Spacy (more basic) and Flair (more advanced)
  - For Chinese, Baidu's PaddlePaddle
- If your tasks is very specific, then these general purpose packages may not be useful
  - E.g., we want to identify mentions of company names in Weibo.
  - If people use short names

How do we obtain a labeled training set?

- Given some texts, how can we get labels?
- Try to find external sources first
- Jake M. Hofman, Amit Sharma, and Duncan J. Watts, *Prediction and explanation in social systems*, Science **355** (2017), no. 6324, 486–488
- Y is cascade size: number of total retweets (including retweets of retweets, and so on so fort)
- X is text of tweets, user info, past number of retweets.
- In other words, Y is automatically obtained by some external process
- Other examples?

Expert annotation

- If you cannot find existing labeled training data that fits your need
- Expert annotation
- E.g., the Comparative Manifesto Project
  - Texts: parties' election manifestos in major electoral democracies
  - Outcomes: a bunch of variables related to the party's policy preferences reflected in the texts.
  - 4,000 manifestos from nearly 1,000 parties in 50 countries and then organized political scientists to systematically code them. Each sentence in each manifesto was coded by an expert using a 56-category scheme
  - `https://manifestoproject.wzb.eu/down/tutorials/primer.html`

Step 1: decide on a codebook

- Codebook: instruction manual for coders to
- `https://manifesto-project.wzb.eu/datasets`

Step 2: select coders

- Usually, at least two coders: to ensure you can calculate some intercoder reliability (more later)
- Inter coder reliability:
  - Cohen's Kappa
  - Cronbach's alpha

Step 3: how many documents to code?

- The number of categories: more categories, more documents needed
  - E.g, if you code 1,000 documents and with two categories (pos / neg), then each category has 500 documents
  - E.g, if you code 1,000 documents on one outcome (support level for liberalism policy) and with 5 levels (strongly agree, agree, neutral, disagree, strongly disagree), then each level has only 200 documents
    - ML algorithm may easily overfit on 200 documents
- With low level of reliability, you need to code more

Step 3: how many documents to code?

- No rule of thumb on when you need to stop
- First collect several hundreds or a thousand, if your Y is binary
- Then start to fit some models and see performances
- And see if you need to code more

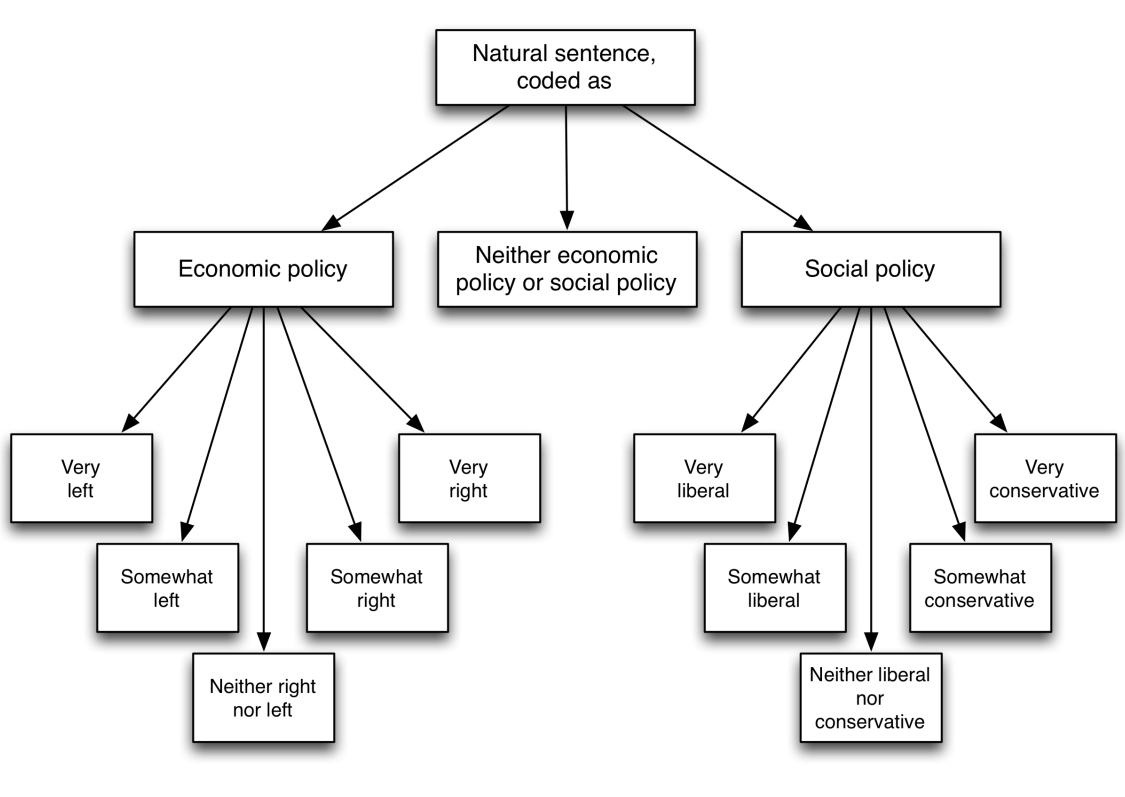
Step 4: train and start coding

- Typically needs some training / pilot tests
- Then calculate intercoder reliability on first round results
- Then re-train coders; sit down together and resolve the disagreements
  - This means that you need to resolve the disagreements and make sure each coder agrees with on how future similar cases should be coded
- Then work independently to finish the coding

Crowd-sourced coding

- Crowd-sourced coding:
  - Wisdom of crowds: aggregated judgments of non-experts converge to judgments of experts at much lower cost
- E.g., crowd-sourced coding of the Comparative Manifesto Project
- Kenneth Benoit, Drew Conway, Benjamin E. Lauderdale, Michael Laver, and Slava Mikhaylov, *Crowd-sourced Text Analysis: Reproducible and Agile Production of Political Data*, American Political Science Review **110** (2016), no. 2, 278–295
  - crowd-source workers were asked to classify each sentence as referring to economic policy (left or right), to social policy (liberal or conservative), or to neither
  - Key here: simplify the burden for coder! 56 categories are too much for non-experts.

Crowd-sourced coding



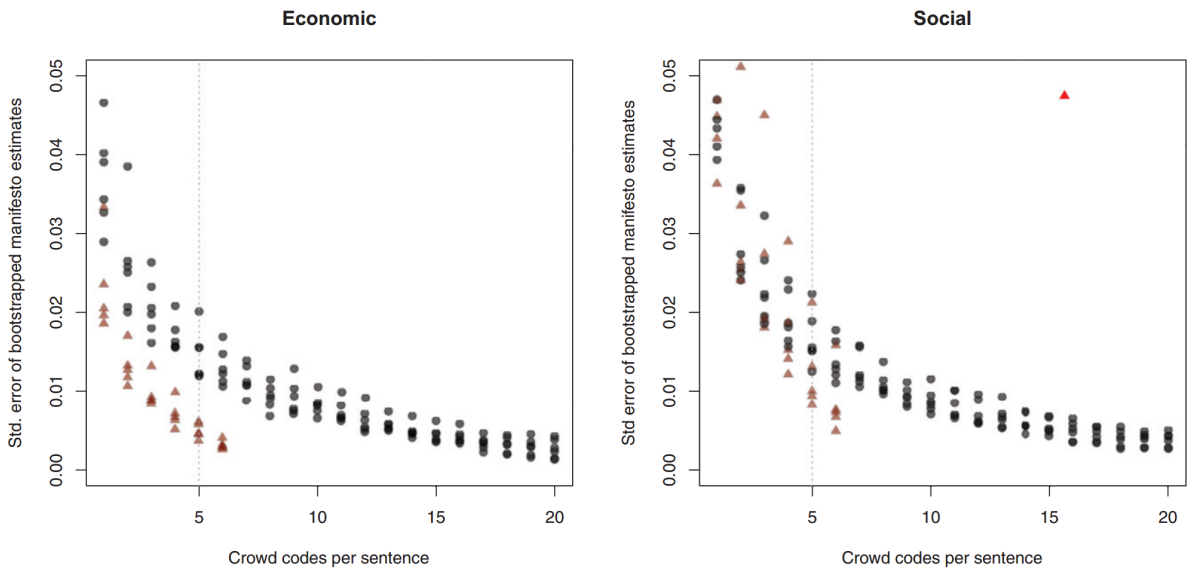
Crowd-sourced coding vs expert coding

	crowd-sourced	expert
instruction	limited	detailed
task difficulty	very easy	can be complex
No. of coders	a few	hundreds
error for one coder	high	low

Crowd-sourced coding

- Error of coding decreases with more coders on the same document,

FIGURE 5. Standard Errors of Manifesto-level Policy Estimates as a Function of the Number of Workers, for the Oversampled 1987 and 1997 Manifestos



Note: Each point is the bootstrapped standard deviation of the mean of means aggregate manifesto scores, computed from sentence-level random n subsamples from the codes.

Crowd-sourced coding

- The crowd-source coding produce high-quality results, on par with expert coding
- And it is quick
- E.g., Benoit et al. want to code a new variable related to immigrants
  - "Within 5 hours of launching their project, the results were in. They had collected more than 22,000 responses at a total cost of $360", based on around 51 coders
- Common platforms: Amazon Mechanical Turk, CrowdFlower