Workshop: NLP and Computational Linguistics in Economics

Princeton University – March 2024

Benjamin W. Arold, ETH Zurich

Welcome

- This workshop focuses on applications of **natural language processing and computational linguistics** in **applied economics**.
- Part 1: Methods Overview
  - Convert natural language texts – e.g. legal and political documents – to data.
  - Relate text data to metadata to understand economic/political/social forces.
- Part 2: Paper presentation: "When Words Matter: The Value of Collective Bargaining Agreements" (Arold, Ash, MacLeod, Naidu, mimeo)
  - Application of methods introduced in Part 1 to question in labor economics

Text is high-dimensional

- Sample of documents, each n_L words long, drawn from vocabulary of n_V words.
- The unique representation of each document has dimension $n_V^{n_L}$.
  - e.g., a sample of 30-word Twitter messages using only the one thousand most common words in the English language
    - $\rightarrow$ dimensionality $= 1000^{30} = 10^{32}$

Part 1: Syllabus

- Dictionary-Based Methods
- Tokenization
- Measuring Document Distance
- Machine Learning with Text
- Word Embeddings
- Linguistic Parsing
- Large-Language Models: Overview
- Large-Language Models: GPT and BERT

Dictionary-Based Methods

Tokenization

Measuring Document Distance

Machine Learning with Text

Embeddings

Linguistic Parsing

Large-Language Models: Overview

Large-Language Models: GTP and BERT

Overview of Dictionary-Based Methods

- Dictionary-based methods one way to reduce dimensionality
- Use a pre-selected list of words or phrases to analyze a corpus.
- Corpus-specific: counting sets of words or phrases across documents
  - (e.g., number of times a judge says “justice” vs “efficiency”)
  - in practice: use regular expressions for this task
- General dictionaries: WordNet, LIWC, MFD, etc.

Measuring uncertainty in macroeconomy

Baker, Bloom, and Davis (QJE 2016)

For each newspaper on each day since 1985, submit the following query:

1. Article contains “uncertain” OR “uncertainty”, AND
2. Article contains “economic” OR “economy”, AND
3. Article contains “congress” OR “deficit” OR “federal reserve” OR “legislation” OR “regulation” OR “white house”

Normalize resulting article counts by total newspaper articles that month.

Measuring uncertainty in macroeconomy

Baker, Bloom, and Davis (QJE 2016)

For each newspaper on each day since 1985, submit the following query:

1. Article contains "uncertain" OR "uncertainty", AND
2. Article contains "economic" OR "economy", AND
3. Article contains "congress" OR "deficit" OR "federal reserve" OR "legislation" OR "regulation" OR "white house"

Normalize resulting article counts by total newspaper articles that month.

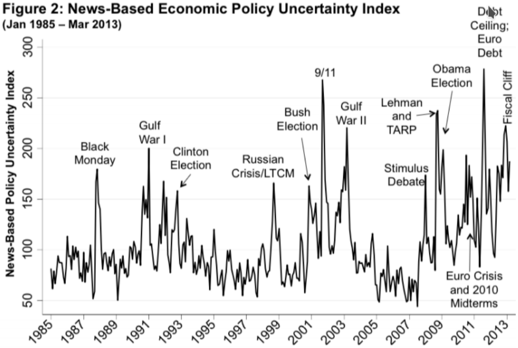


Figure 2: News-Based Economic Policy Uncertainty Index
(Jan 1985 – Mar 2013)

Measuring uncertainty in macroeconomy

Baker, Bloom, and Davis (QJE 2016)

For each newspaper on each day since 1985, submit the following query:

1. Article contains “uncertain” OR “uncertainty”, AND
2. Article contains “economic” OR “economy”, AND
3. Article contains “congress” OR “deficit” OR “federal reserve” OR “legislation” OR “regulation” OR “white house”

Normalize resulting article counts by total newspaper articles that month.

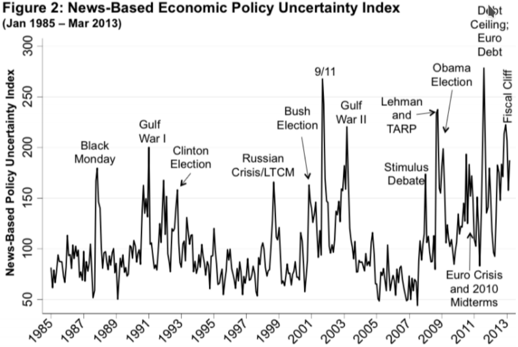
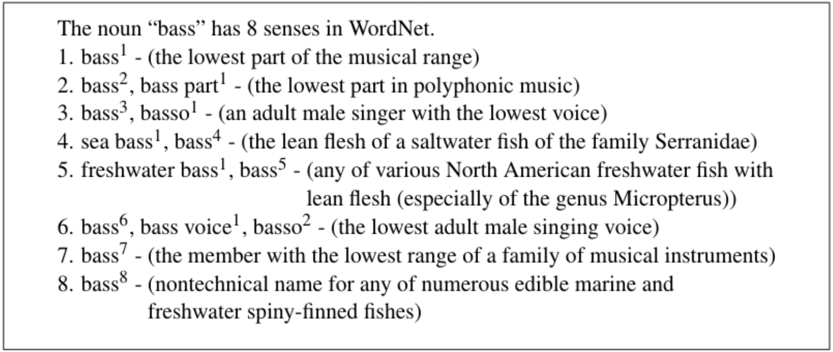


Figure 2: News-Based Economic Policy Uncertainty Index
(Jan 1985 – Mar 2013)

- but see Keith et al (2020), showing some problems with this measure (`https://arxiv.org/abs/2010.04706`).

WordNet

- English word database: 118K nouns, 12K verbs, 22K adjectives, 5K adverbs



The noun “bass” has 8 senses in WordNet.

1. bass^1 - (the lowest part of the musical range)
2. bass^2, bass part^1 - (the lowest part in polyphonic music)
3. bass^3, basso^1 - (an adult male singer with the lowest voice)
4. sea bass^1, bass^4 - (the lean flesh of a saltwater fish of the family Serranidae)
5. freshwater bass^1, bass^5 - (any of various North American freshwater fish with lean flesh (especially of the genus Micropterus))
6. bass^6, bass voice^1, basso^2 - (the lowest adult male singing voice)
7. bass^7 - (the member with the lowest range of a family of musical instruments)
8. bass^8 - (nontechnical name for any of numerous edible marine and freshwater spiny-finned fishes)

Figure 19.1 A portion of the WordNet 3.0 entry for the noun *bass*.

- Synonym sets (synsets) are a group of near-synonyms, plus a gloss (definition).
  - also contains information on antonyms (opposites), holonyms/meronyms (part-whole).
- Nouns are organized in categorical hierarchy (hence “WordNet”)
  - “hypernym” – the higher category that a word is a member of.
  - “hyponyms” – members of the category identified by a word.

General Dictionaries

- Function words (e.g. *for*, *rather*, *than*)
  - also called stopwords
  - can be used to get at non-topical dimensions, identify authors.
- LIWC (pronounced “Luke”): Linguistic Inquiry and Word Counts
  - more than 70 lists of category-relevant words, e.g. “emotion”, “cognition”, “work”, “family”, “positive”, “negative” etc.
- Mohammad and Turney (2011):
  - code 10,000 words along four emotional dimensions: joy–sadness, anger-fear, trust-disgust, anticipation-surprise
- Warriner et al (2013):
  - code 14,000 words along three emotional dimensions: valence, arousal, dominance.

Dictionary-Based Methods

Tokenization

Measuring Document Distance

Machine Learning with Text

Embeddings

Linguistic Parsing

Large-Language Models: Overview

Large-Language Models: GTP and BERT

Tokenization

- Input:
  - A set of documents (e.g. text files), D.
- Pre-processing:
  - removing page numbers, capitalization, punctuation, etc.
- Output:
  - Tokens: A sequence, $\boldsymbol{w}$, containing a list of tokens (words/n-grams) in document i, for use in natural language processing
  - n-grams: learn a vocabulary of phrases and tokenize those: "Princeton University → princeton_university"
  - Counts/Frequencies: A **document-term matrix**, X, containing statistics about word/phrase frequencies in each document.
  - "Bag-of-words" representation, a row of X is just the frequency distribution over words (n-grams, syntax features) in the document corresponding to that row.

Pre-processing

- An important piece of the “art” of text analysis is deciding what data to throw out.
  - Uninformative data add noise and reduce statistical precision.
  - They are also computationally costly
  - For sequence data, e.g. language modeling, need to keep everything.
- Pre-processing choices can affect down-stream results, especially in unsupervised learning tasks (Denny and Spirling 2017).
  - some features are more interpretable: “judge has” / “has discretion” vs “judge has discretion”.

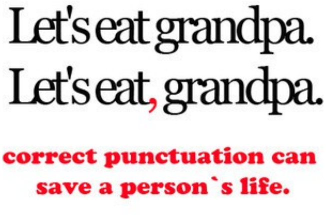
Segmenting paragraphs/sentences

- Many tasks should be done on sentences, rather than corpora as a whole.
  - spaCy does a good (but not perfect) job of splitting sentences, while accounting for periods on abbreviations, etc.
- There isn't a grammar-based paragraph tokenizer.
  - most corpora have new paragraphs annotated.
  - or use line breaks.

Capitalization

- Removing capitalization is a standard corpus normalization technique
  - usually the capitalized/non-capitalized version of a word are equivalent – e.g. words showing up capitalized at beginning of sentence
  - $\rightarrow$ capitalization not informative.
- Also: what about “the first amendment” versus “the First Amendment”?
  - Compromise: include capitalized version of words not at beginning of sentence.
- For some tasks, capitalization is important
  - needed for sentence splitting, part-of-speech tagging,named entity recognition, syntactic/semantic parsing

Punctuation



Let's eat grandpa.

Let's eat, grandpa.

correct punctuation can
save a person`s life.

Source: Chris Bail text data slides.

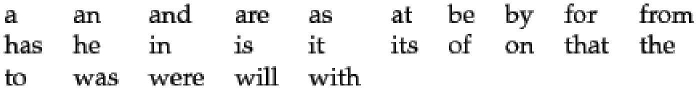
Inclusion of punctuation depends on your task:

- if you are vectorizing the document as a bag of words or bag of n-grams, punctuation won't be needed.
- like capitalization, punctuation is needed for annotations (sentence splitting, parts of speech, syntax, roles, etc) or for text generators.

Numbers

- for bag of words/phrases:
    - drop numbers, or replace with a special character (e.g. #)
- for language models:
    - just treat them like letters.

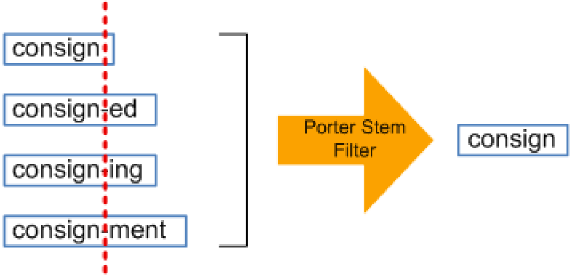
Drop Stopwords?



a	an	and	are	as	at	be	by	for	from
has	he	in	is	it	its	of	on	that	the
to	was	were	will	with					

- What about “not guilty”?
- Legal “memes” often contain stopwords:
  - “beyond a reasonable doubt”
  - “with all deliberate speed”
- can drop stopwords by themselves, but keep them as part of phrases.

Stemming/lemmatizing



- Effective dimension reduction with little loss of information.
- Lemmatizer produces real words, but N-grams won't make grammatical sense
  - e.g., "judges have been ruling" would become "judge have be rule"

Word frequency

- Inspect low-frequency words and determine a minimum document threshold.
    - e.g., 10 documents, or .25% of documents.
- Inspect high-frequency words and determine a maximum document threshold.
    - e.g., 10000 documents, or 90% of documents.

Dictionary-Based Methods

Tokenization

Measuring Document Distance

Machine Learning with Text

Embeddings

Linguistic Parsing

Large-Language Models: Overview

Large-Language Models: GTP and BERT

Document-Term Matrix

The **document-term matrix $\boldsymbol{X}$**:

- each row d represents a **document,** while each column w represents a word (or term more generally, e.g. n-grams).
  - A matrix entry $\boldsymbol{X}_{[d,w]}$ quantifies the strength of association between a document and a word, generally its count or frequency
- each document/row $\boldsymbol{X}_{[d,:]}$ is a distribution over terms
  - these vectors have a **spatial interpretation** → geometric distances between document vectors reflect semantic distances between documents in terms of shared terms.
- each word/column $\boldsymbol{X}_{[:,w]}$ is a distribution over documents.
  - these vectors also have a spatial interpretation! geometric distances between word vectors reflect semantic distances between words in terms of showing up in the same documents.

Cosine Similarity

- Each document is a vector x_d, e.g. term counts or TF-IDF frequencies.
- $\rightarrow$ Each document is a non-negative vector in an n_x-space, where $n_x =$ vocabulary size.
    - that is, documents are rays, and similar documents have similar vectors.
- Can measure similarity between documents i and j by the cosine of the angle between x_i and x_j :
    - With perfectly collinear documents (that is, $x_i = \alpha x_j$, $\alpha > 0$), $\cos(0) = 1$
    - For orthogonal documents (no words in common), $\cos(\pi/2)$=0

Cosine similarity is computable as the normalized dot product between the vectors:

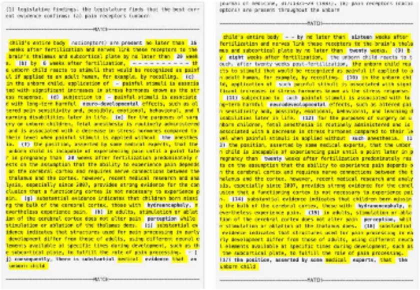
$$\text{cos\_sim}(x_1, x_2) = \frac{x_1 \cdot x_2}{||x_1||||x_2||}$$

Burgess et al, “Legislative Influence Detectors”

- Compare bill texts across states in two-step process:
  (1) find candidates using elasticsearch (tf-idf similarity);
  (2) compare candidates using text reuse score.

Burgess et al, “Legislative Influence Detectors”

- Compare bill texts across states in two-step process:
  (1) find candidates using elasticsearch (tf-idf similarity);
  (2) compare candidates using text reuse score.



(1) legislative findings. the legislature finds that the best current evidence confirms: (a) pain receptors (unborn

MATCH

child's entire body nociceptors) are present no later than 16 weeks after fertilization and nerves link these receptors to the brain's thalamus and subcortical plate by no later than 20 weeks. (b) by 8 weeks after fertilization, - - - - - - - - - - - the unborn child reacts to stimuli that would be recognized as painful if applied to an adult human, for example, by recoiling. (c) in the unborn child, application of - painful stimuli is associated with significant increases in stress hormones known as the stress response. (d) subjection to - painful stimuli is associated with long-term harmful neuro-developmental effects, such as altered pain sensitivity and, possibly, emotional, behavioral, and learning disabilities later in life. (e) for the purposes of surgery on unborn children, fetal anesthesia is routinely administered and is associated with a decrease in stress hormones compared to their level when painful stimuli is applied without the anesthesia. (f) the position, asserted by some medical experts, that the unborn child is incapable of experiencing pain until a point later in pregnancy than 20 weeks after fertilization predominately rests on the assumption that the ability to experience pain depends on the cerebral cortex and requires nerve connections between the thalamus and the cortex. however, recent medical research and analysis, especially since 2007, provides strong evidence for the conclusion that a functioning cortex is not necessary to experience pain. (g) substantial evidence indicates that children born missing the bulk of the cerebral cortex, those with hydranencephaly, nevertheless experience pain. (h) in adults, stimulation or ablation of the cerebral cortex does not alter pain perception while stimulation or ablation of the thalamus does. (i) substantial evidence indicates that structures used for pain processing in early development differ from those of adults, using different neural elements available at specific times during development, such as the subcortical plate, to fulfill the role of pain processing. - (j) consequently, there is substantial medical evidence that an unborn child

MATCH

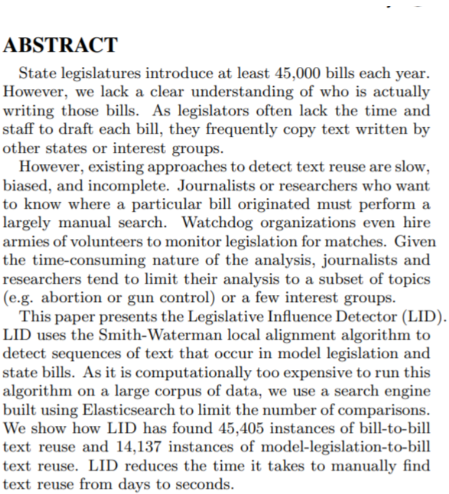
journal of medicine, 317:1321-29 (1987). (8) pain receptors (nociceptors) are present throughout the unborn

MATCH

child's entire body - - by no later than sixteen weeks after fertilization and nerves link these receptors to the brain's thalamus and subcortical plate by no later than twenty weeks. (9) by eight weeks after fertilization, the unborn child reacts to touch. after twenty weeks post-fertilization, the unborn child reacts to stimuli that would be recognized as painful if applied to an adult human, for example, by recoiling. (10) in the unborn child, application of such painful stimuli is associated with significant increases in stress hormones known as the stress response. (11) subjection to such painful stimuli is associated with long-term harmful neurodevelopmental effects, such as altered pain sensitivity and, possibly, emotional, behavioral, and learning disabilities later in life. (12) for the purposes of surgery on unborn children, fetal anesthesia is routinely administered and is associated with a decrease in stress hormones compared to their level when painful stimuli is applied without such anesthesia. (13) the position, asserted by some medical experts, that the unborn child is incapable of experiencing pain until a point later in pregnancy than twenty weeks after fertilization predominately rests on the assumption that the ability to experience pain depends on the cerebral cortex and requires nerve connections between the thalamus and the cortex. however, recent medical research and analysis, especially since 2007, provides strong evidence for the conclusion that a functioning cortex is not necessary to experience pain. (14) substantial evidence indicates that children born missing the bulk of the cerebral cortex, those with hydranencephaly, nevertheless experience pain. (15) in adults, stimulation or ablation of the cerebral cortex does not alter pain perception, while stimulation or ablation of the thalamus does. (16) substantial evidence indicates that structures used for pain processing in early development differ from those of adults, using different neural elements available at specific times during development, such as the subcortical plate, to fulfill the role of pain processing. (17) the position, asserted by some medical experts, that the unborn child

MATCH

Figure 10: Match between Scott Walker's bill and a highly similar bill from Louisiana. For a detailed view, please visit http://dssg.uchicago.edu/lid/.



ABSTRACT

State legislatures introduce at least 45,000 bills each year. However, we lack a clear understanding of who is actually writing those bills. As legislators often lack the time and staff to draft each bill, they frequently copy text written by other states or interest groups.

However, existing approaches to detect text reuse are slow, biased, and incomplete. Journalists or researchers who want to know where a particular bill originated must perform a largely manual search. Watchdog organizations even hire armies of volunteers to monitor legislation for matches. Given the time-consuming nature of the analysis, journalists and researchers tend to limit their analysis to a subset of topics (e.g. abortion or gun control) or a few interest groups.

This paper presents the Legislative Influence Detector (LID). LID uses the Smith-Waterman local alignment algorithm to detect sequences of text that occur in model legislation and state bills. As it is computationally too expensive to run this algorithm on a large corpus of data, we use a search engine built using Elasticsearch to limit the number of comparisons. We show how LID has found 45,405 instances of bill-to-bill text reuse and 14,137 instances of model-legislation-to-bill text reuse. LID reduces the time it takes to manually find text reuse from days to seconds.

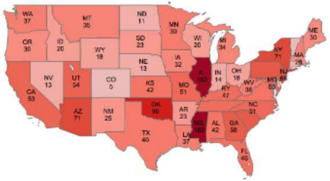


Figure 7: Introduced bills by state from ALEC model legislation

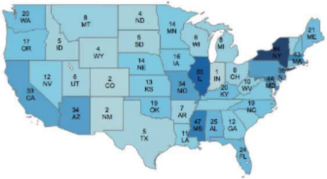


Figure 8: Introduced bills by state from ALICE model legislation

Dictionary-Based Methods

Tokenization

Measuring Document Distance

Machine Learning with Text

Embeddings

Linguistic Parsing

Large-Language Models: Overview

Large-Language Models: GTP and BERT

Different Goals, Different Methods

- Supervised Learning
  - pursuing a known goal, e.g., predicting whether a political speech is from a Democrat or a Republican.
  - machine learns to replicate labels for new data points
- Unsupervised Learning
  - algorithm discovers themes/patterns in text (or other high-dimensional data)
  - human interprets the results (e.g. inspect content of topics or clusters)
- Both strategies amplify human effort, each in different ways.

What do ML Algorithms do? Minimize a cost function

- A typical cost function (or loss function) for regression problems is Mean Squared Error (MSE):

$$\text{MSE}(\theta) = \frac{1}{n_D} \sum_{i=1}^{n_D} (h(x_i;\theta) - y_i)^2$$

  - n_D, the number of rows/observations
  - x, the matrix of predictors, with row x_i
  - y, the vector of outcomes, with item y_i
  - $h(x_i;\theta) = \hat{y}$ the model prediction (hypothesis)

The **data** (x, y) are taken as given, and the ML algorithm searches for **parameters** θ to minimize the cost function.

Linear Regression and Numerical Optimization

- Ordinary Least Squares Regression (OLS) assumes the functional form $f(x;\theta) = x_i'\theta$ and minimizes the mean squared error (MSE)

$$\min_{\hat{\theta}} \frac{1}{n_D} \sum_{i=1}^{n_D} (x_i'\hat{\theta} - y_i)^2$$

- This minimand has a closed form solution

$$\hat{\theta} = (\boldsymbol{x}'\boldsymbol{x})^{-1}\boldsymbol{x}'\boldsymbol{y}$$

- Most machine learning models do **not** have a closed form solution → use numerical optimization instead (gradient descent).

$$\text{MSE}(\theta) = \frac{1}{n_D} \sum_{i=1}^{n_D} (h(\theta;\boldsymbol{x}_i) - y_i)^2$$

- The partial derivative for feature j is

$$\frac{\partial \text{MSE}}{\partial \theta_j} = \frac{2}{n_D} \sum_{i=1}^{n_D} (\underbrace{h(\theta;\boldsymbol{x}_i) - y_i}_{\text{error for this obs}}) \underbrace{\frac{\partial h(\theta;\boldsymbol{x}_i)}{\partial \theta_j}}_{\text{how } \theta_j \text{ shifts } h(\cdot)}$$

  - → estimates how changing θ_j would change the error across the whole dataset

Machine Learning with Text Data

- We have a corpus (or dataset) D of $n_D \geq 1$ documents d_i (or data points).
- Each document i has an associated outcome or label $\boldsymbol{y}_i$ with dimensions $n_y \geq 1$
- Some documents are labeled and some are unlabeled $\rightarrow$
  - we would like to learn a function $\hat{\boldsymbol{y}}(d_i)$ based on the labeled data ...
  - ... to machine-classify the unlabeled data.

First Problem

- Each document is a sequence of symbols d_i, while (standard) ML algorithms work on numbers.
- The solution: all the methods from previous workshop parts for extracting informative numerical information from documents:
  - style features
  - counts over dictionary patterns
  - tokens
  - n-grams
  - (principal components)
  - (topic shares)
  - etc.
- documents can thus be **featurized** – represented as a matrix of vectors $\boldsymbol{x}$ with $n_x \geq 1$ features.

A sample baseline for machine learning using text

1. For example, take dimension-reduced bigrams as inputs X.
2. Select a machine learning model for predicting outcome y:
   - For classification (y is discrete), e.g., L2 (Lasso/Ridge)-penalized logistic regression
   - For regression (y is continuous), e.g., elastic net
   - *(If y is more complicated, e.g. a sequence of words, we use deep learning.)*
3. Use cross-validation grid search in training set to select model hyperparameters.
   - For classification, use cross entropy; for regression, use mean squared error.
4. Evaluate model in held-out test set:
   - For classification, use balanced accuracy, confusion matrix, and calibration plot.
   - For regression, use R squared and binscatter plot.
5. Interpret the model predictions:
   - for gradient boosting, use feature importance ranking.
   - for linear models, examine coefficients
   - look at highest and lowest ranked documents for $\hat{y}$
6. Answer the research question!

Unsupervised ML with Text: Topic Models

- Core methods for topic models were developed in computer science and statistics
  - summarize unstructured text
  - use words within document to infer subject
  - useful for dimension reduction
- Social scientists use topics as a form of measurement
  - how observed covariates drive trends in language
  - tell a story not just about what, but how and why
  - **topic models are more interpretable** than other dimension reduction methods, such as PCA.

Standard Topic Model

- Latent Dirichlet Allocation (LDA):
  - Each topic is a distribution over words.
  - Each document is a distribution over topics.
- Input: $N \times M$ document-term count matrix X
- Assume: there are K topics (tunable hyperparameter, use coherence).
- Like PCA, LDA works by factorizing X into:
  - an $N \times K$ document-topic matrix
  - an $K \times M$ topic-term matrix.
- LDA discovers topics based upon co-occurrence of individual words (though labeling is up to the user)

Using an LDA Model

Once trained, can easily get topic proportions for a corpus.

- for any document – doesn't have to be in training corpus.
- main topic is the highest-probability topic
- documents with highest share in a topic work as representative documents for the topic.

Can then use the topic proportions as variables in a social science analysis.

Dictionary-Based Methods

Tokenization

Measuring Document Distance

Machine Learning with Text

Embeddings

Linguistic Parsing

Large-Language Models: Overview

Large-Language Models: GTP and BERT

Overview

- So far: Focus on global document counts
- An influential line of work in NLP, known as "word embedding", reframes text analysis
- now: represent the meaning of words by neighboring words – their **local contexts**.
- move from high-dimensional sparse representations to low-dimensional dense representations
- "You shall know a word by the company it keeps"

GloVe Embeddings

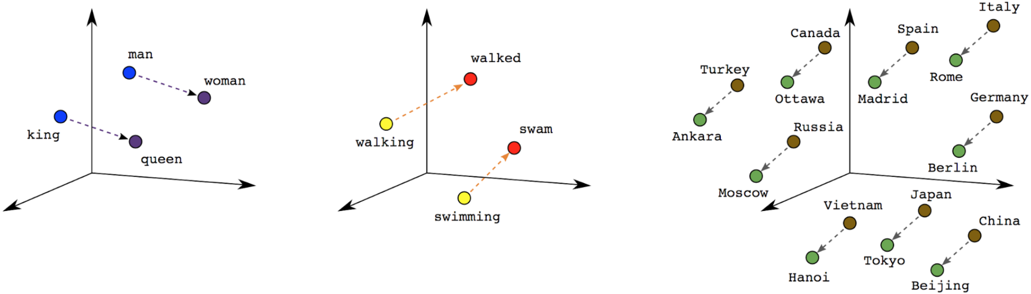
- Define a co-occurrence matrix W, with W_{ij} = local co-occurrence counts between words i, j
  - that is, within some co-occurence window, typically 10 words.
- Define word vectors $\boldsymbol{v} = (v_1, ..., v_i, ..., v_{n_w})$, where $v_i \in (-1, 1)^{n_E}$,
  - initialized randomly
  - n_E typically ≈ 200
- then use gradient descent to solve

$$\min_{\boldsymbol{v}} \sum_{i,j} f(W_{ij}) \left(v_i^T v_j - \log(W_{ij}) \right)^2$$

  - $f(\cdot)$ is a non-negative, increasing, concave weighting function
- Minimizes **squared difference** between**:**
  - **dot product of word vectors,** $v_i^T v_j$
  - **empirical co-occurrence,** $\log(W_{ij})$
- Intuitively: words that co-occur should have high correlation (dot product)

Dimensions

- Geometric dimensions in space equal semantic dimensions in language:



Male-Female

Verb Tense

Country-Capital

- Once words are represented as vectors $\{v_1, v_2, \ldots\}$, we can use linear algebra to understand the relationships between words:
  $vec(king) - vec(man) + vec(woman) \approx vec(queen)$
- Can also apply to sentences instead of words (“sentence embeddings”)

Dictionary-Based Methods

Tokenization

Measuring Document Distance

Machine Learning with Text

Embeddings

Linguistic Parsing

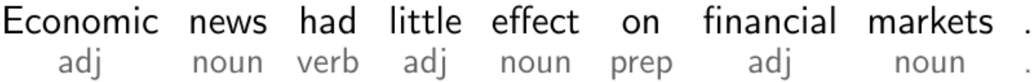
Large-Language Models: Overview

Large-Language Models: GTP and BERT

Dependency Parsing

- The models we have seen so far have counted tokens, now we also incorporate grammatical concepts
- The basic idea:
  - **Syntactic structure** consists of **words**, linked by binary directed relations called **dependencies**.
  - Dependencies identify the grammatical relations between words.

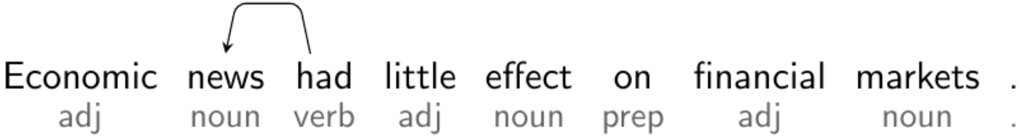
Dependencies: Binary Directed Relations Between Words (Head and Dependent)



Economic	news	had	little	effect	on	financial	markets	.
adj	noun	verb	adj	noun	prep	adj	noun	.

- dependency trees are mostly determined by the ordering of POS tags.

Dependencies: Binary Directed Relations Between Words (Head and Dependent)

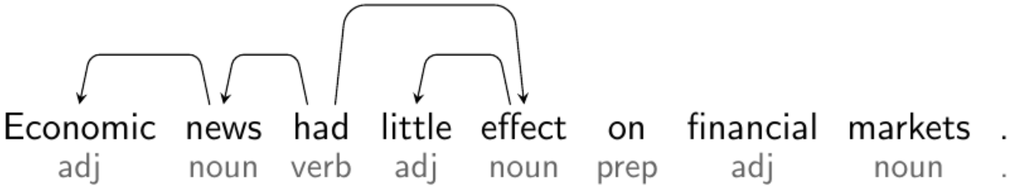


Economic news had little effect on financial markets .

adj noun verb adj noun prep adj noun .

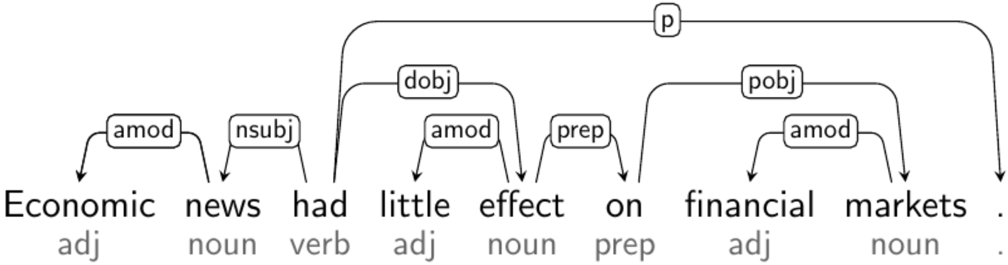
- the “root” of a sentence is the main verb (for compound sentences, the first verb).

Dependencies: Binary Directed Relations Between Words (Head and Dependent)



- directed arcs indicate dependencies: a one-way link from a "head" token to a "dependent" token.
- A word can be "head" multiple times, but "dependent" only one.

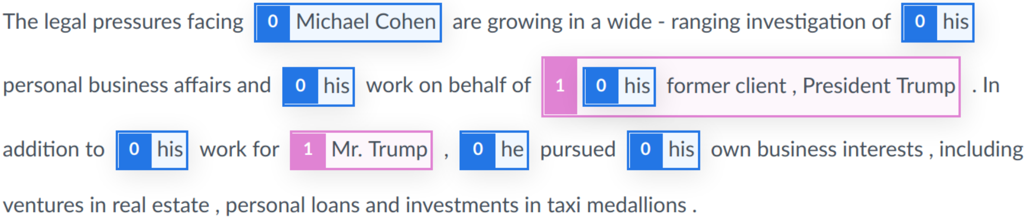
Dependencies: Binary Directed Relations Between Words (Head and Dependent)



- arc labels indicate functional relations, e.g.:
  - nsubj: verb → subject doing the verb
  - dobj: verb → object targeted by the verb
  - amod: noun → attribute of the noun
- spaCy dependency visualizer: `https://explosion.ai/demos/displacy`
- Allows to extract grammar-based information from text

Co-Reference Resolution

Finding all expressions that refer to the same entity in a text.



The legal pressures facing 0 Michael Cohen are growing in a wide - ranging investigation of 0 his personal business affairs and 0 his work on behalf of 1 0 his former client , President Trump . In addition to 0 his work for 1 Mr. Trump , 0 he pursued 0 his own business interests , including ventures in real estate , personal loans and investments in taxi medallions .

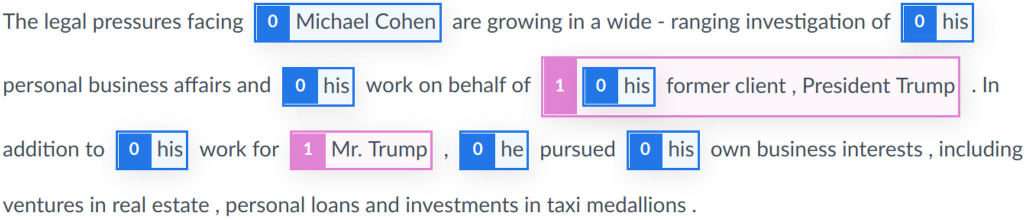
"My sister has a cat. Her name is Roberta."

↓

[Cat's] name is Roberta ↔ [Sister's] name is Roberta

Co-Reference Resolution

Finding all expressions that refer to the same entity in a text.



The legal pressures facing Michael Cohen are growing in a wide - ranging investigation of his personal business affairs and his work on behalf of his former client , President Trump . In addition to his work for Mr. Trump , he pursued his own business interests , including ventures in real estate , personal loans and investments in taxi medallions .

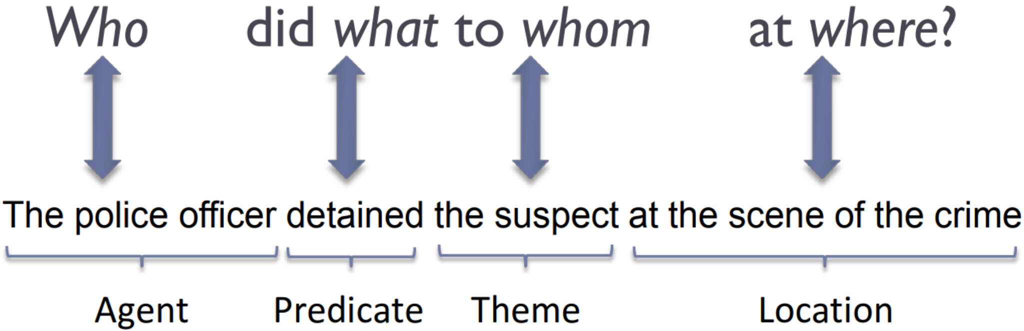
“My sister has a cat. Her name is Roberta.”

↓

[Cat’s] name is Roberta ↔ [Sister’s] name is Roberta

`https://demo.allennlp.org/coreference-resolution`

Addendum: Semantic Parsing


Who did *what* to *whom* at *where?*
The police officer detained the suspect at the scene of the crime
Agent
Predicate
Theme
Location

Source: Jurafsky-Martin slides.

Dictionary-Based Methods

Tokenization

Measuring Document Distance

Machine Learning with Text

Embeddings

Linguistic Parsing

Large-Language Models: Overview

Large-Language Models: GTP and BERT

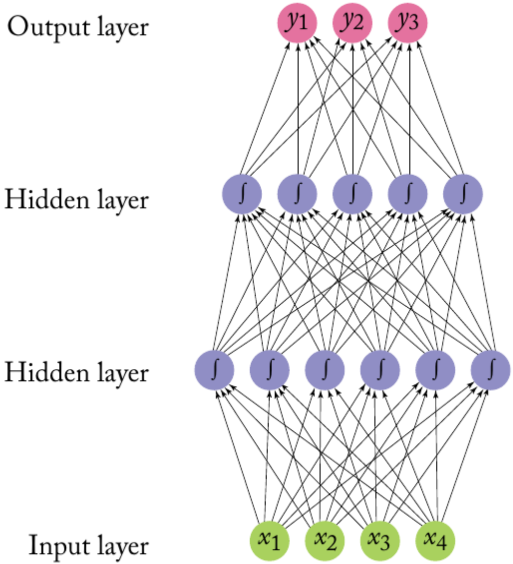
What are LLMs? (Korinek, 2023)

- Type of generative AI that produces text
- AI systems trained to predict the next word given preceding text
- Typically fine-tuned to follow human instructions and generate responses aligned with human preferences
- Based on deep neural networks with billions of parameters
- Built on transformer models (with attention mechanisms, which endogenously assign varying degrees of importance to different words)

Step 1: Pre-Training

- Calculate conditional probability distribution over words given the preceding words, based on its training data ("Next Token Prediction")
    - Self-Supervised Learning: Model is fed text fragments; parameters are adjusted to predict continuation
    - Terabytes of data (Wikipedia, scientific articles, books, etc.)
    - Neural nets learns language structure: syntactic structures, relationships between words and concepts they represent, context of sentences and how different words interact in that context, and how different sentences are related

Neural Net Example (Multi-Layer Perceptron)



- A multilayer perceptron (also called a feed-forward network or sequential model) stacks neurons horizontally and vertically.
- alternatively, think of it as a stacked ensemble of logistic regression models.
- this vertical stacking is the “deep” in “deep learning”!

Step 2: Instruction Fine-Tuning

- Improves model to follow human instructions
  - According to pretrained model, a likely continuation of “What’s your name?” may be “And how old are you?-> This is not the answer we want
  - Instruction fine-tuning makes model learn how to respond to user’s instructions
  - Supervised learning: Feeding the model millions of examples for how to respond to thousands of different instructions for tasks like summarization, answering questions, brainstorming, etc

Step 3: Reinforcement learning

- Improves model by incorporating human feedback
  - Feedback from human raters tells the model how different responses compare
  - Makes model better aligned with human preferences, in particular along domains that are difficult to define via instruction fine-tuning (for example, to penalize hateful responses)
  - Noisy process (for example, it is part of the reason why LLMs have learned to sound authoritative even when they hallucinate)

Limitations

- Hallucination
- Biases (from training data and human feedback)
- Privacy concerns (from training data and human feedback)
- Reproducibility (vs. diversity of text)
- Data limitations

Dictionary-Based Methods

Tokenization

Measuring Document Distance

Machine Learning with Text

Embeddings

Linguistic Parsing

Large-Language Models: Overview

Large-Language Models: GTP and BERT

Autoregressive vs Autoencoding Language Models

- **Autoregressive models**:
  - e.g. **GPT = "Generative Pre-Trained Transformer"**:
  - pretrained on classic language modeling task: guess the next token having read all the previous ones.
  - during training, attention heads only view previous tokens, not subsequent tokens.
  - ideal for text generation.

GPT

- GPT-1: the first autoregressive transformer model (2018)
  - trained on BookCorpus (around 7000 books)
- GPT-2 (2019)
  - trained on all articles linked from Reddit with at least 3 upvotes (8 million documents, 40 GB of text)
- GPT-3 (2020)
  - even bigger corpus (Common Crawl, WebText2, Books1, Books2 and Wikipedia)
- GPT-3.5 (2022)
  - subclass of GPT-3 trained on data up to June 2021
  - incorporates the base model on which ChatGPT is fine-tuned and after optimized for chat
- GPT-4 (2023)
  - multimodal model: can take also images as inputs (i.e., can describe the humor in unusual images, summarize text from screenshots, and answer exam questions that contain diagrams)
  - trained in two stages: token prediction (like other GPT models), and reinforcement learning with human feedback
  - much, much larger model (details not public)

Autoregressive vs Autoencoding Language Models

- **Autoencoding models**
  - e.g. **BERT = “Bidirectional Encoder Representations from Transformers”**
  - pretrained by dropping/shuffling input tokens and trying to reconstruct the original sequence (random masking).
  - usually build bidirectional representations and get access to the full sequence.
  - can be fine-tuned and achieve great results on many tasks, e.g. text classification.

BERT

- BERT = Bidirectional Encoder Representations from Transformers
- Task: Masked language modeling:
  - 15% of words masked (randomly)
  - if masked: replace with [MASK] 80% of the time, a random token 10% of the time, and left unchanged 10% of the time.
  - model has to predict the original word.
- Unlike GPT, BERT attention observes all tokens in the sequence, reads backwards and forwards (bidirectional).
- Corpus:
  - 800M words from English books (modern work, from unpublished authors), by Zhu et al (2015).
  - 2.5B words of text from English Wikipedia articles (without markup).
  - Architecture: The largest BERT model has $\approx$ 340M parameters to learn (a stack of transformer blocks with a self-attention layer and an MLP.)

End of Part 1: Main Python packages for NLP

- nltk – broad collection of pre-neural-nets NLP tools
- scikit-learn – ML package with nice text vectorizers, clustering, and supervised learning
- xgboost – gradient-boosted machines for supervised learning
- gensim – topic models and embeddings
- spaCy – tokenization, NER, parsing, pre-trained vectors
- huggingface – source for pre-trained transformer models
- More coding material: https://github.com/BenjaminArold/

Thank you!

Benjamin Arold | aroldb@ethz.ch