Text as Data: The Promise and Pitfalls of Automatic Content Analysis Methods for Political Texts[1]

Jasmine. Hao[1]

[1]University of Hong Kong

ECON 6083: Machine Learning

[1]This section is based on [Grimmer and Stewart, 2013]

Outline

Language and Politics

- Political Discourse:
  - Candidates **debate** policies during campaigns.
  - Elected officials engage in **legislative writing** and debates.
  - Bureaucrats seek public input on **regulations** post-legislation.
  - Nations utilize language in **peace treaties**.
  - Media coverage of **international relations**.
  - Political entities express ideologies through **platforms** and manifestos.
  - Terrorist groups communicate objectives through **media**.

Challenges and Advances in Political Text Analysis

Challenges in Political Text Analysis

- Volume Problem:
  - Inundation with texts, manual analysis **impractical**.
  - Costs for manual coding are **prohibitive**.

Advances in Automated Content Analysis

- Automated Methods:
  - Enable **systematic analysis** of large text collections.
  - Require careful **validation** due to language complexity.

Overview of Automated Content Analysis

- Automated vs. Manual Analysis:
  - Automated methods **amplify** and **augment** detailed examination.
  - **Validation** is crucial as methods are **imperfect models** of language.
- Text Collection:
  - Accessible electronic texts have spurred **interest** in automated methods.
- Classification and Scaling Tasks:
  - **Classification** organizes texts into categories.
  - **Scaling** estimates positions in policy space.
  - Both require **validation** to determine appropriateness.

Scope and Limitations

- Focused Review:
  - Concentrates on **document-level analysis**.
  - Natural language processing has **limited adoption** in political science.
- Further Learning:
  - Supplementary materials for **in-depth exploration** of methods.
  - Discussion distribution does not reflect **method prevalence**.

An overview of text as data methods

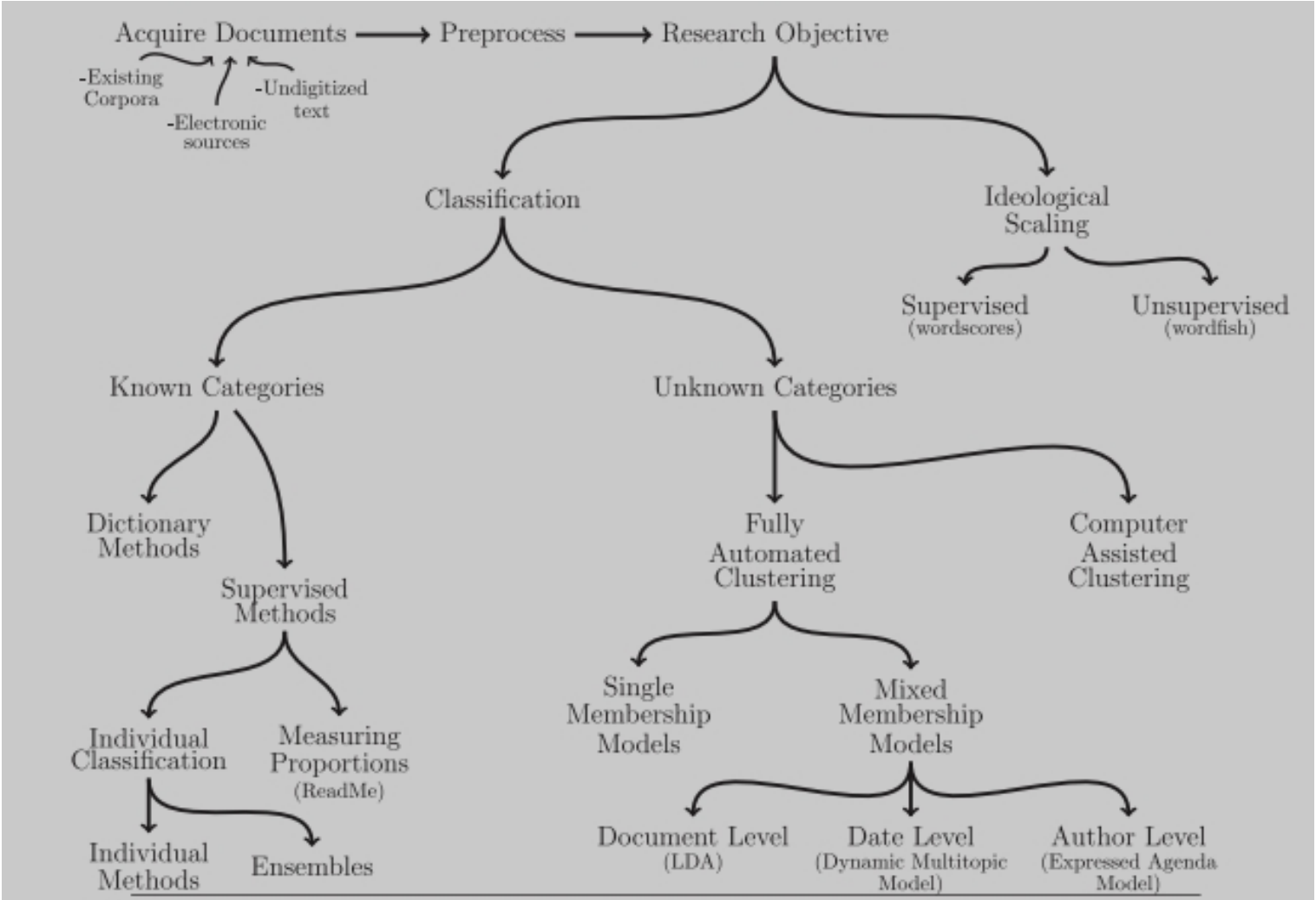

Fig. 1 An overview of text as data methods.

Principles of Automated Text Analysis

Volume, Validation, Complexity, and Method Adaptation

- **Principle 1: Volume and Scalability**
  - Designed for large volumes of data.
  - Scalable to analyze extensive corpora.
  - Vital for the growing digital text in political science.
- **Principle 2: Validation**
  - Requires rigorous validation for reliability.
  - Cross-validation with manually coded data is key.
  - Must account for political discourse nuances.
- **Principle 3: Language Complexity**
  - Must navigate inherent complexity and context-dependency.
  - Should capture sarcasm, irony, and rhetorical strategies.
  - Cannot fully replace careful reading and interpretation.
- **Principle 4: Method Development and Adaptation**
  - Not a one-size-fits-all; requires adaptation.
  - Must evolve with language and communication changes.
  - Collaboration between experts is crucial for tailored methods.

Acquiring Text for Automated Analysis

Diverse Texts in Political Science:

- Media archives, legislative speeches, political statements, legislation, and more.

Sources and Formats:

- Electronic databases (e.g., Lexis Nexis, ProQuest, JSTOR).
- Government websites (e.g., U.S. House Resolutions in XML).
- Web scraping and crowdsourcing platforms (e.g., Mechanical Turk).

Preparation for Analysis:

- Conversion of texts to computer-readable formats (UTF-8, XML).
- Archival materials via scanning and OCR technology.

Suitable Texts for Automated Methods:

- Focused texts (for classification or policy positions).
- Longer texts preferred but large volumes of short texts can suffice.

Reducing Complexity: From Words to Numbers

The Recipe for Quantitative Text Representation

The Challenge:

- Language is complex; not all complexity is necessary for analysis.

The Recipe:

1. **Discard Word Order**: Treat documents as a "bag of words".
2. **Use of N-grams**: Optionally include bigrams or trigrams to retain some order.
3. **Stemming**: Simplify vocabulary by reducing words to their roots.
4. **Lemmatization (Optional)**: Reduce words to base forms using context and dictionaries.
5. **Discard Punctuation/Capitalization**: Remove grammar elements that don't convey content.
6. **Remove Stop and Rare Words**: Exclude non-discriminating words.

Outcome:

- Documents are vectors of word counts (Document-Term Matrix).
- The approach retains substantively interesting properties of texts despite information reduction.

Document-Term Matrix and Feature Reduction

Document-Term Matrix (DTM):

- Each document is a vector of word counts: $W_i = (W_{i1}, W_{i2}, ..., W_{iM})$.
- DTM is sparse, mostly zeroes, typically 3,000 - 5,000 features.

Feature Reduction:

- Addresses high dimensionality and sparsity of text data.
- Enhances computational efficiency and model interpretability.

Validation:

- Despite reduction, empirical evidence shows sufficient information is retained.
- Validation is key: Test different approaches and validate results.

Note:

- Adapt steps based on the specific problem and corpus characteristics.
- Balance between information retention and simplification.

Classifying Documents in Political Science

Classifying Documents into Known Categories

- **Political science research** often categorizes texts such as campaigns, legislation, international statements, and media coverage.
- Inference goals:
  - Determine the category of each document.
  - Distribution of documents across categories.

Challenges of Manual Classification

- Time-consuming and resource-intensive.
- Requires coding rules and trained coders.
- Coders must read each text individually.

Automated Methods Advantage

- Reduce classification costs.
- Limit manual classification to a subset of documents.

Dictionary Methods in Automated Classification

Dictionary Methods

- Classify documents based on the frequency of **key words**.
- Intuitive and easy to apply, particularly for measuring categories like **tone**.

Example of Dictionary Method for Tone Analysis

- Lists of words with positive or negative connotations.
- Measure document tone by the relative occurrence of these words.

Formal Measurement

$$t_i = \frac{\sum_{m=1}^{M} s_m W_{im}}{N_i}$$

Pitfalls of Dictionary Methods

- Words must align with context-specific usage.
- Cross-domain application can lead to errors.
- Validation is crucial but often overlooked.

Improving Dictionary Methods

- Simplify classification to binary for validation.
- Apply scrutiny similar to unsupervised methods.

Supervised Learning Methods I

Beyond Dictionary Methods

- Dictionary methods have limitations, particularly when applied outside their original domain.
- Supervised learning methods offer a domain-specific alternative.

The Concept of Supervised Learning

- Human coders categorize a set of documents manually.
- The algorithm learns to categorize new documents based on this training set.

Supervised Learning Methods II

Advantages Over Dictionary Methods

1. Domain specificity - tailored coding rules and definitions.
2. Easier validation - clear statistics summarizing model performance.

Steps in Supervised Learning for Text Classification

1. Construct a training set with hand-coded categories.
2. Train the algorithm to learn the relationship between document features and categories.
3. Validate the model output and classify the remaining documents.

Published Resources

- Jurka et al. 2012 provides excellent software for supervised learning in text classification.

Constructing a Training Set I

The Foundation of Supervised Learning

- The training set's quality is crucial; no model can compensate for a poorly constructed training set.
- A well-coded training set can even mask the faults of simple models.

Creating a Coding Scheme

- Develop coding schemes iteratively to handle language ambiguities and nuanced concepts.
- Start with a concise codebook, refine it through coder feedback, and revise until ambiguities are resolved.
- For further reading on coding schemes and coder agreement, see Krippendorff (2004), Neuendorf (2002), and Weber (1990).

Constructing a Training Set II

Sampling Documents

- The training set should be representative of the entire corpus.
- Use random sampling to capture a representative sample, which can be simple or stratified.
- The number of documents needed is context-dependent, but Hopkins and King (2010) suggest a minimum of 100 to 500.
- The complexity of the coding scheme affects the required size of the training set.

Key Considerations

- Ensure the training set accurately reflects the variety of documents.
- Address challenges when data is not fully available at the time of coding.

Applying a Supervised Learning Model

- After hand classification, the labeled documents train supervised learning methods to classify or measure proportions of categories in a test set.
- Training set with N_{train} documents, each coded into one of K categories.
- Each document i's category is $Y_i \in \{C_1, C_2, \ldots, C_K\}$ represented as $Y_{\text{train}} = (Y_1, \ldots, Y_{N_{\text{train}}})$.
- Document i's features are in an M-length vector W_i, collected in the $N_{\text{train}} \times M$ matrix W_{train}.
- Supervised learning algorithms assume a function f describing the relationship: $Y_{\text{train}} = f(W_{\text{train}})$.
- The model learns to predict the category Y_i of each document based on its word features W_i.
- Validation schemes are used to optimize the number of documents needed for effective training.

Naive Bayes Classifier I

Learning from Text Data

- Utilizes training data to determine word distribution for category k.
- Applies Bayes's rule to classify documents in the test set.

Bayes's Rule

$$p(C_k|W_i) \propto p(C_k)p(W_i|C_k)$$

- $p(C_k)$ estimated as the proportion of training documents in category k.
- $p(W_i|C_k)$ more complex due to vast word count vectors.

Naive Bayes Classifier II

Naive Assumption

- Assumes word independence within a document's category.
- Simplifies $p(W_i|C_k)$ estimation:
  $p(W_{im} = j|C_k) = \frac{\text{Number of documents in category } k \text{ with word } m \text{ used } j \text{ times}}{\text{Number of documents in category } k}$

Classification

- Assigns each document to the category with the highest estimated probability.
- Addresses zero-frequency problem with smoothing.

Naive Bayes Classifier: Addressing Data Sparsity

- Naive Bayes models face challenges with word-specific counts that never occur in the training set (zero-frequency problem).
- The common approach to mitigate this is **smoothing** – adding a small constant to each probability estimate.
- This technique is often justified using a **Bayesian Dirichlet-Multinomial model**.
- With smoothing, Naive Bayes assigns each document to the category with the highest probability, despite the **independence assumption** being inherently incorrect.
- The model's effectiveness across various tasks illustrates that a model doesn't need to be perfect to be useful.
- Naive Bayes is just one part of a diverse set of classification algorithms, including Random Forests, Support Vector Machines, and neural networks.

Study Overview and Methodology

Context:

- Stewart and Zhukov (2009) analyze public statements by Russian elites on foreign policy.
- Corpus of 7920 statements from 1998 to 2008.
- Aim: Classify stances as restrained, activist, or neutral concerning use of force.

Approach:

- Development of a codebook based on close reading.
- Human coders classified a random sample of 300 documents.
- Financial constraints limited sample size due to the cost of Russian-speaking coders.

Machine Learning Application:

- A Random Forest model was trained with human-coded samples.
- The model then classified the remaining uncoded documents.

Validation and Results

Model Validation:

- Ten-fold cross-validation to measure algorithm's performance.
- Confusion matrix compared machine and human classifications.

Performance Metrics:

- Overall accuracy: 65%.
- Precision for 'restrained' category: 65%.
- Recall for 'restrained' category: 75%.

Implications:

- Potential for the supervised method to replicate human coding.
- Options to improve accuracy include other methods, ensembles, or changing the research focus.
- Application to the full dataset revealed more activist stances among military elites than previously thought.

Table 2 Confusion matrix: comparing human and supervised coding

		Training data		
		Restrained	*Activist*	*Neutral*
Machine	Restrained	111	31	28
	Activist	10	17	0
	Neutral	26	9	68

Table 3 Document classifications by Elite type (proportion in parentheses)

		Military	*Political*
Training set	Restrained	27 (0.36)	119 (0.53)
	Activist	25 (0.34)	32 (0.14)
	Neutral	22 (0.30)	74 (0.33)
Test set	Restrained	870 (0.41)	3550 (0.62)
	Activist	500 (0.24)	260 (0.04)
	Neutral	749 (0.35)	1960 (0.34)

Measuring Latent Features in Texts: Scaling Political Actors I

- One of the most promising applications of automated content analysis methods is to **locate political actors** in **ideological space**.
- Estimating locations is often **difficult** or **impossible** using existing data.
- **Roll call votes** are used to scale legislators, but are less reliable outside the **U.S. Congress**.
- Alternative scaling methods are needed for other political actors who do not cast votes, such as **presidents**, **bureaucrats**, and **political candidates**.

- Challenges and Opportunities
  - Current methods often depend on **disclosure institutions** that are absent in many democracies.
  - **Speech** as a form of data is nearly universal among political actors.
  - A method that uses **text** to place actors in political space would greatly benefit the testing of **political theories**.

Measuring Latent Features in Texts: Scaling Political Actors II

- Text-Based Scaling Methods
  We describe two methods for scaling political actors using texts:
  - A **supervised method**similar to dictionary approachesfor situating actors based on their words.
  - An **unsupervised method** for locating actors in space by analyzing speech patterns and word usage.

  These methods contribute to testing **spatial theories** of politics.

Measuring Latent Features in Texts: Scaling Political Actors III

- Advancements and Goals in Scaling Literature
  - Recent technical contributions have improved scaling methods.
  - There is a need for a clearer articulation of **goals** in scaling literature.
  - Text-based methods should aim beyond replicating **expert opinion** or existing scalings.
  - A potential goal is the **prediction of political events**, such as votes in Congress or legislative coalitions.

Measuring Latent Features in Texts: Scaling Political Actors IV

- Validation and Ideological Dominance Assumptions
  - **Validation** of scales is crucial for the reliability of models.
  - Current models rely on the assumption of **ideological dominance** in speech.
  - This assumption may not always apply, as seen in contexts like **Senate press releases**.
  - Improved performance requires methods that can **separate ideological** from non-ideological statements.
  - Future research should focus on **nuanced methods** for this separation.

Unsupervised Methods for Scaling I

- Unsupervised methods **discover words** that distinguish political spectra without reference texts.
- **Monroe and Maeda (2004)**, and **Slapin and Proksch (2008)** introduce item response theory (IRT)-based models for automatic spatial location estimation of parties.
- Politicians/parties are assumed to reside in a **low-dimensional political space**. This is quantified by the parameter θ_i for politician i.
- A politicians usage rate of words in texts is assumed to be affected by their position in this space.

Unsupervised Methods for Scaling II

- **Wordfish method:** a Poisson-IRT model developed by Slapin and Proksch (2008).
  - Each word j from individual i, W_{ij}, is drawn from a Poisson distribution with rate λ_{ij}: $W_{ij} \sim \text{Poisson}(\lambda_{ij})$.
  - λ_{ij} is modeled as: $\lambda_{ij} = \exp(\alpha_i + \psi_j + \pi_j \times \theta_i)$.
  - Parameters represent:
    - α_i: individual is loquaciousness.
    - ψ_j: frequency word j is used.
    - π_j: extent to which a word discriminates the underlying ideological space.
    - θ_i: politicians underlying position.
- Demonstration of model application in following section, with conditions for reliable policy position retrieval.

Applying Unsupervised Methods to Political Texts

- **Strengths of IRT methods:**
  - Can estimate political actors' spatial locations with **minimal resources**.
  - Utilizes primary variation in language across actors.
- **Limitations:**
  - Lacks supervision, potentially capturing non-ideological variations such as policy focus, style, or tone.
  - Outputs require **careful validation** to confirm ideological location.
- **Case of German Party Platforms:**
  - Wordfish algorithm successfully replicated expert assessments (Slapin & Proksch, 2008).
- **Case of Senate Press Releases:**
  - Wordfish algorithm fails to differentiate between parties in polarized contexts.
  - Reveals non-ideological dimensions such as the balance of position taking and credit claiming.

References I


Grimmer, J. and Stewart, B. M. (2013).
Text as data: The promise and pitfalls of automatic content analysis methods for political texts.
Political analysis, 21(3):267–297.