Discovering Word Associations in News Media via Feature Selection and Sparse Classification

Brian Gawalt

UC Berkeley
Departments of EECS

Based on joint work with:

Jinzhu Jia (UC Berkeley)
Luke Miratrix (UC Berkeley)
Laurent El Ghaoui (UC Berkeley)
Bin Yu (UC Berkeley)
Sophie Clavier (SFSU)

Motivating Question

- There exist techniques from machine learning capable of reliably classifying documents based on lexical content. They are a promising resource for greater understanding of large document collections.

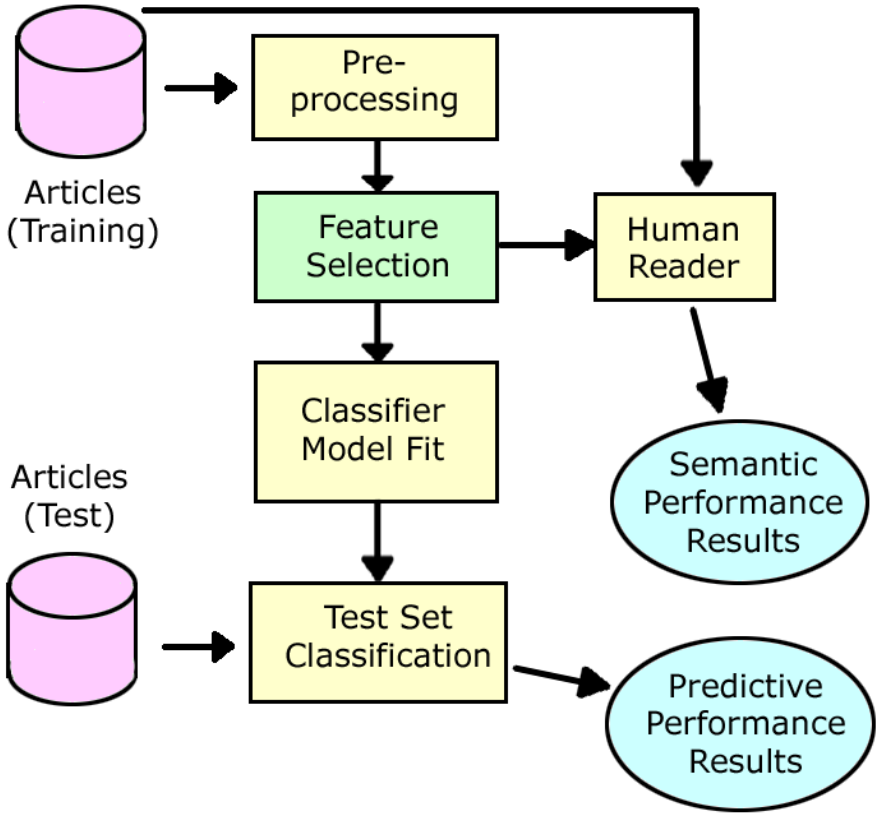
- However, nearly all these lack an interpretable path from word use to document class:
  - Classifier structure of some, e.g. decision trees, produce complicated, nested rules
  - Nearly all models consider "too many" words in the decision process – human interpretation requires sparsity of a classifier

- We ask:
  - To what degree does human interpretability sacrifice classification capability?
  - Which (interpretable) models & methods best aid human understanding of document corpora?

Case study: word image in the New York Times

- For any query word q, we can split a corpus of document units into those containing q and those not
- We can then establish interpretable classifer models which provide a best guess as to whether a hypothetical document unit would or would not contain q given its word use pattern.
- We investigate feature selection processes which drop all but $k \approx 15$ words from inclusion prior to establishing such a classification rule
  - Consider this list of k words under feature selection process P the image of q under P
  - Evaluate these images for (comparative) ability both to lead to good classifiers and ability to convey semantic meaning to a human reader
- Experiments conducted with a body of articles from the *New York Times* International section
  - Dates run contiguously from December 15, 2008 through October 18, 2009
  - Articles broken into 109,686 paragraphs, each taken as a document unit
  - 79,494 distinct word tokens

Our approach: feature selection for logistic regression models



- Independent variable: Feature selection process
- Dependent variables: Semantic/predictive performance
- Experiment is conducted repeatedly across many queries (adjusting the training/testing datasets for each) in order to broadly test the effects of the choice of feature selection process alone.

Pre-processing

- White-space maintained while punctuation and numerals omitted
- Characters reduced to lower case
- Stop words omitted
  - Words known *a priori* to be uninteresting
  - E.g., `the`, `is`, `a`, `in`; majority prepositions and linking verbs
- No stemming implemented
  - `iraqis` denotes the citizens, as distinct from the more general `iraqi`
- Bag-of-words vectorization
  - Each document unit encoded as a vector with integral elements
  - Each element of vector corresponds to one distinct (non-stop) word token found in the corpus
  - Value of element corresponds to number of times token appears in unit (ergo, vectors are sparse)
- For each experiment, choose one token as the classification label
  - Positive example vectors have non-zero values as that token's element

Classifier Model: Logistic Regression

- Logistic regression model: The log-odds ratio of a document unit belonging to the positive class equals the inner product of the document's word count vector with the model parameters

$$\log \frac{P(y_i = 1 \mid x_i, \beta)}{P(y_i = 0 \mid x_i, \beta)} = \log \frac{P(y_i = 1 \mid x_i, \beta)}{1 - P(y_i = 1 \mid x_i, \beta)} = \beta_0 + x_1\beta_1 + \ldots + x_n\beta_n \quad (1)$$

$$P(y_i \mid x_i, \beta) = \frac{1}{1 + \exp(-y_i(\beta_0 + x^T\beta))} \quad (2)$$

- We can assemble a log-likelihood for parameter set β taking the log of the product of all above terms, $i = 1, \ldots, m$:

$$\mathcal{L}(\beta) = \sum_{i=1}^{m} \log P(y_i \mid x_i, \beta) = -\sum_{i=1}^{m} \log \left(1 + \exp(-y_i(\beta_0 + x^T\beta))\right) \quad (3)$$

- This can be maximized for β efficiently ($O(m^3n^3)$ or better), and preserves a meaningful/interpretable pathway from each word token $j = 1, \ldots, n$ to positive class membership (i.e., mentioning the query term)