Text classification using Expectation-maximization and Semi-supervised Learning

Group S5

Shijie Li, Yifan Guo

04/25/2017

Objective

- Semi-supervised Text Classification
- Naive Bayes (NB) Classifier as baseline
- Expectation-Maximization using Unlabeled Text Data
- Investigate key factors affecting NB Classifier Accuracy & Performance (i.e. labeled vs unlabeled ratio)

Related Work

A Comparison of Event Models for Naive Bayes Text Classification

Andrew McCallum[‡†]
mccallum@justresearch.com

[‡]Just Research
4616 Henry Street
Pittsburgh, PA 15213

Kamal Nigam[†]
knigam@cs.cmu.edu

[†]School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Text Classification by Bootstrapping with Keywords, EM and Shrinkage

Andrew McCallum[‡†]
mccallum@justresearch.com

[‡]Just Research
4616 Henry Street
Pittsburgh, PA 15213

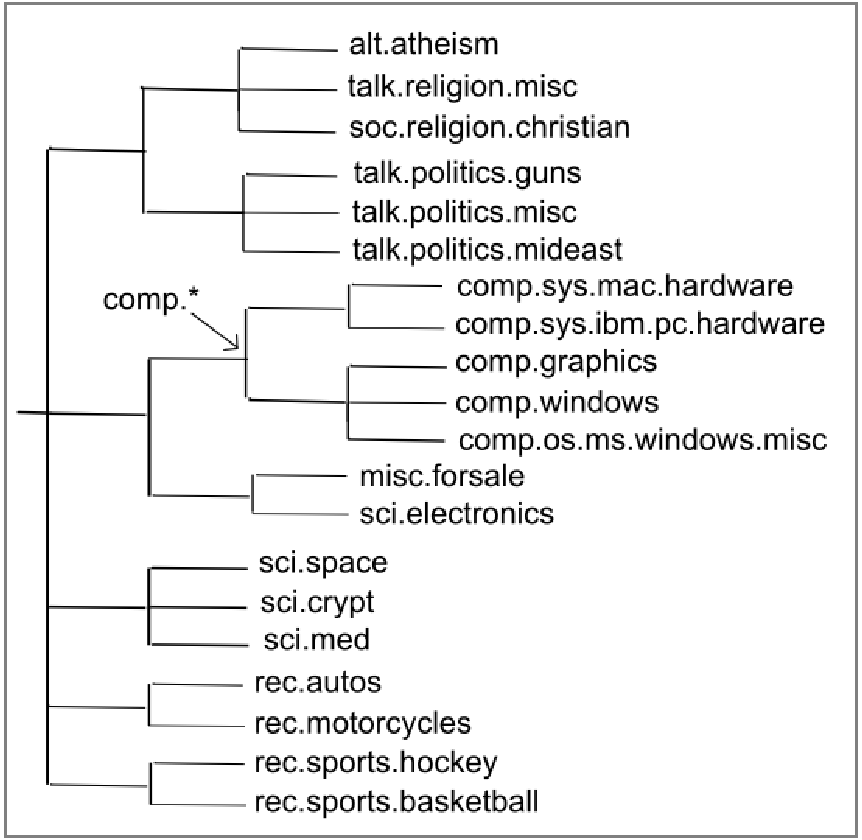
Kamal Nigam[†]
knigam@cs.cmu.edu

[†]School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Dataset Description

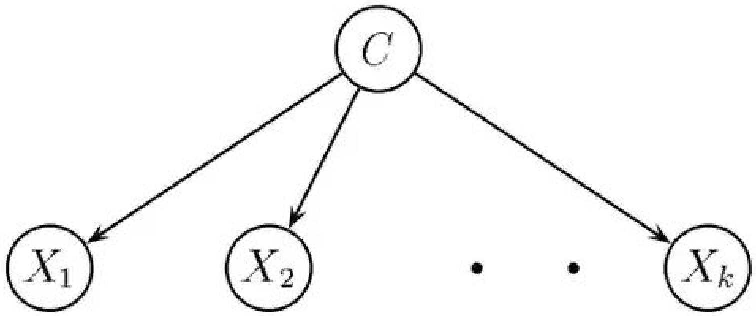
- 20,000 articles
- Hierarchical clustered category
- 20 categories at leaf level
- Evenly 1000 articles per category

- Train vs Test ratio: 80% vs 20%
- Randomly split training data into labeled and unlabeled data set.
- Labeled doc size: 200 ~ 5076
- Unlabeled doc fixed size: 10000



Mechanics - Naive Bayes Classifier

Multinomial Naive Bayes (with m-estimate smoothing)



$$\begin{aligned} p(C_k \mid x_1, \dots, x_n) &\propto p(C_k, x_1, \dots, x_n) \\ &\propto p(C_k)\, p(x_1 \mid C_k)\, p(x_2 \mid C_k)\, p(x_3 \mid C_k) \cdots \\ &\propto p(C_k) \prod_{i=1}^{n} p(x_i \mid C_k)\,. \end{aligned}$$

Training

$$\frac{\quad + \sum_{x_i \in X} \delta_{ij} x_{it}}{|\mathcal{X}| + \sum_{s=1}^{|\mathcal{X}|} \sum_{x_i \in X} \delta_{ij} x_{is}},$$

$$\frac{1 + \sum_{i=1}^{|X|} \delta_{ij}}{M + |X|}.$$

Tune parameter α using Grid Search

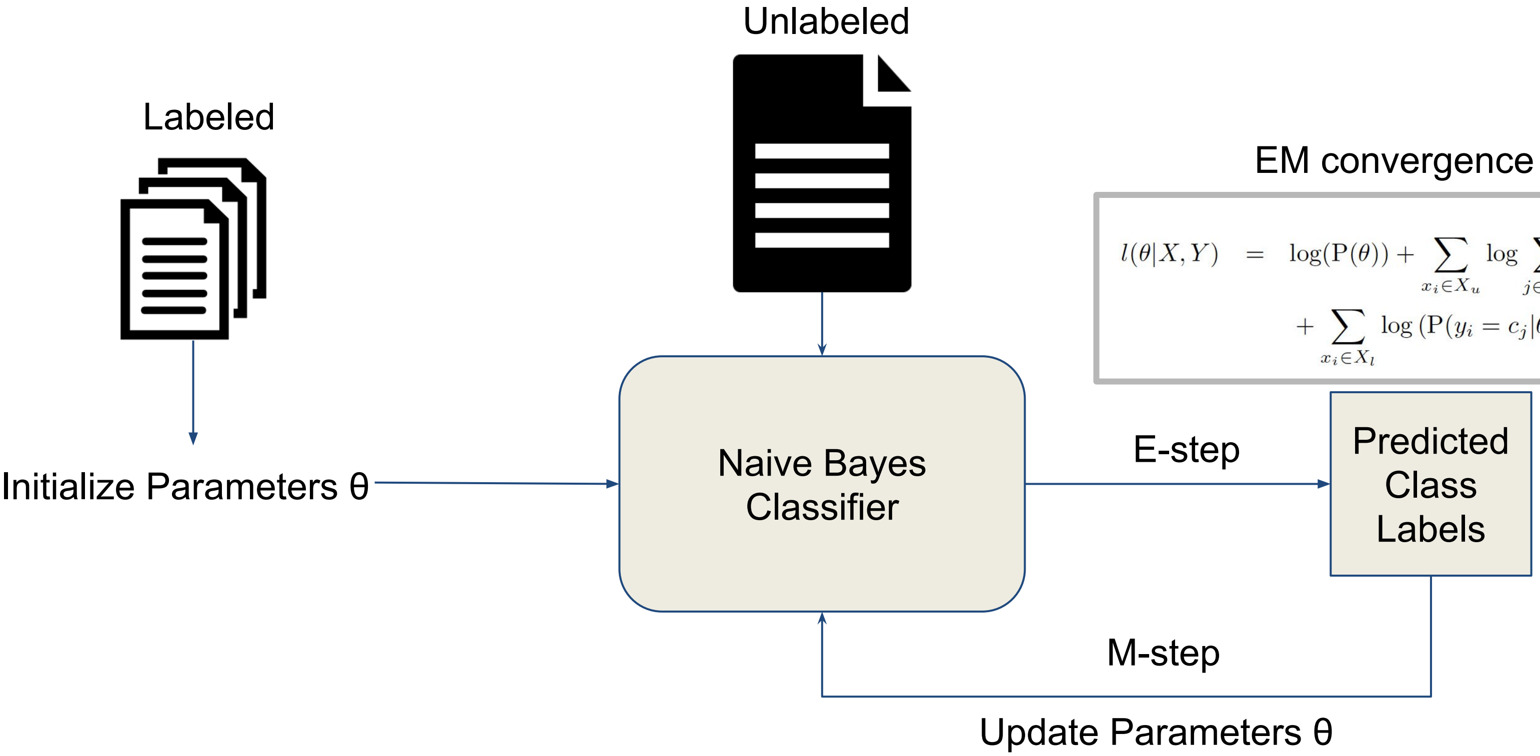
Mechanics - EM Algorithm

- **Inputs:** Collections X_l of labeled documents and X_u of unlabeled documents.
- Build an initial naive Bayes classifier, $\hat{\theta}$, from the labeled documents, X_l, only. Use maximum a posteriori parameter estimation to find $\hat{\theta} = \arg\max_\theta \mathrm{P}(X_l|\theta)\mathrm{P}(\theta)$ (see Equations 1.5 and 1.6).
- Loop while classifier parameters improve, as measured by the change in $l(\theta|X, Y)$ (the log probability of the labeled and unlabeled data, and the prior) (see Equation 1.8):
  - **(E-step)** Use the current classifier, $\hat{\theta}$, to estimate component membership of each unlabeled document, *i.e.*, the probability that each mixture component (and class) generated each document, $\mathrm{P}(c_j|x_i; \hat{\theta})$ (see Equation 1.7).
  - **(M-step)** Re-estimate the classifier, $\hat{\theta}$, given the estimated component membership of each document. Use maximum a posteriori parameter estimation to find $\hat{\theta} = \arg\max_\theta \mathrm{P}(X, Y|\theta)\mathrm{P}(\theta)$ (see Equations 1.5 and 1.6).
- **Output:** A classifier, $\hat{\theta}$, that takes an unlabeled document and predicts a class label.

Table 1.1 The basic EM algorithm for semi-supervised learning of a text classifier.

EM convergence condition:

$$
\begin{aligned}
l(\theta|X,Y) \quad &= \quad \log(\mathrm{P}(\theta)) + \sum_{x_i \in X_u} \log \sum_{j \in [M]} \mathrm{P}(c_j|\theta)\mathrm{P}(x_i|c_j;\theta) \\
&\quad + \sum_{x_i \in X_l} \log\left(\mathrm{P}(y_i = c_j|\theta)\mathrm{P}(x_i|y_i = c_j;\theta)\right).
\end{aligned}
$$



continue mechanics ...

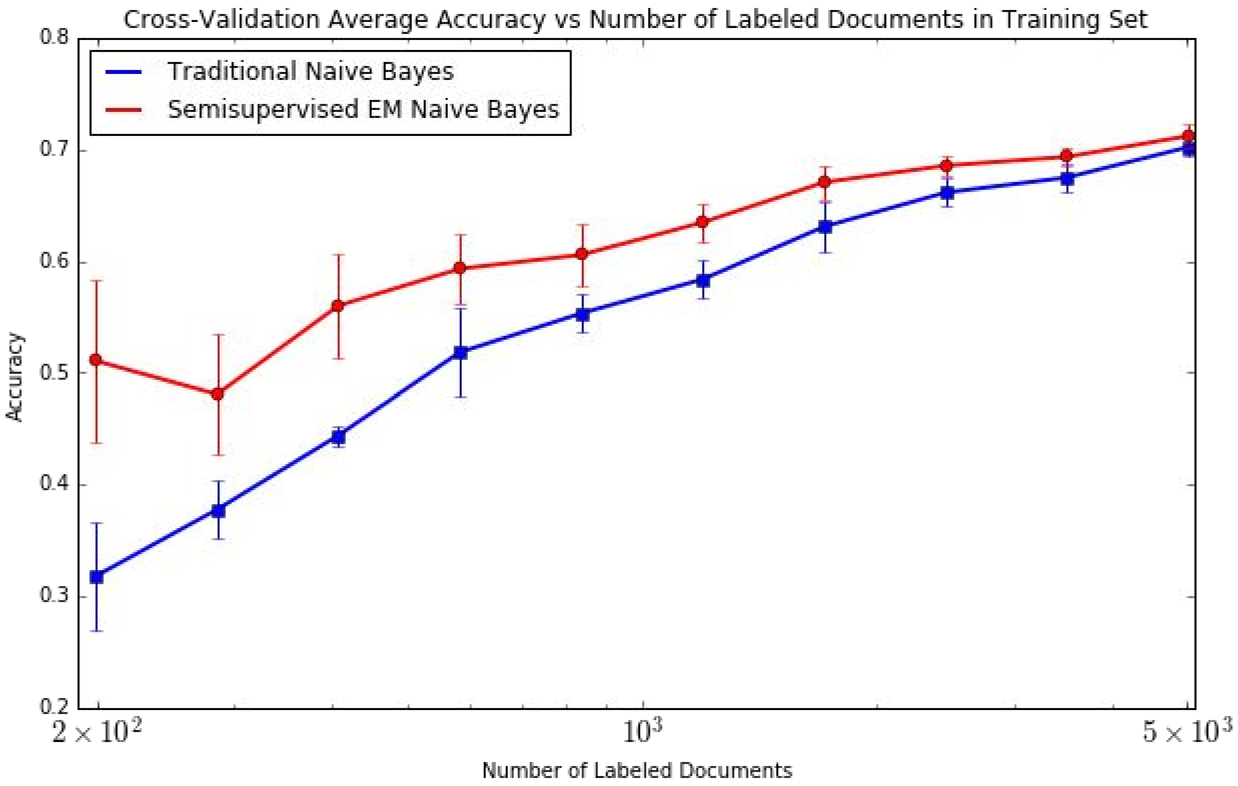
Mechanics - Text Preprocessing Dimension Reduction

Use NLTK library and Regular Expression:

- Remove Noisy Symbols (punctuation, digit, web links, etc.)
- Stemming (reduce derived words to their root form)
- Lemmatization (reduce inflected word forms to single item)
- Remove stopwords and rare words

Result and Analysis

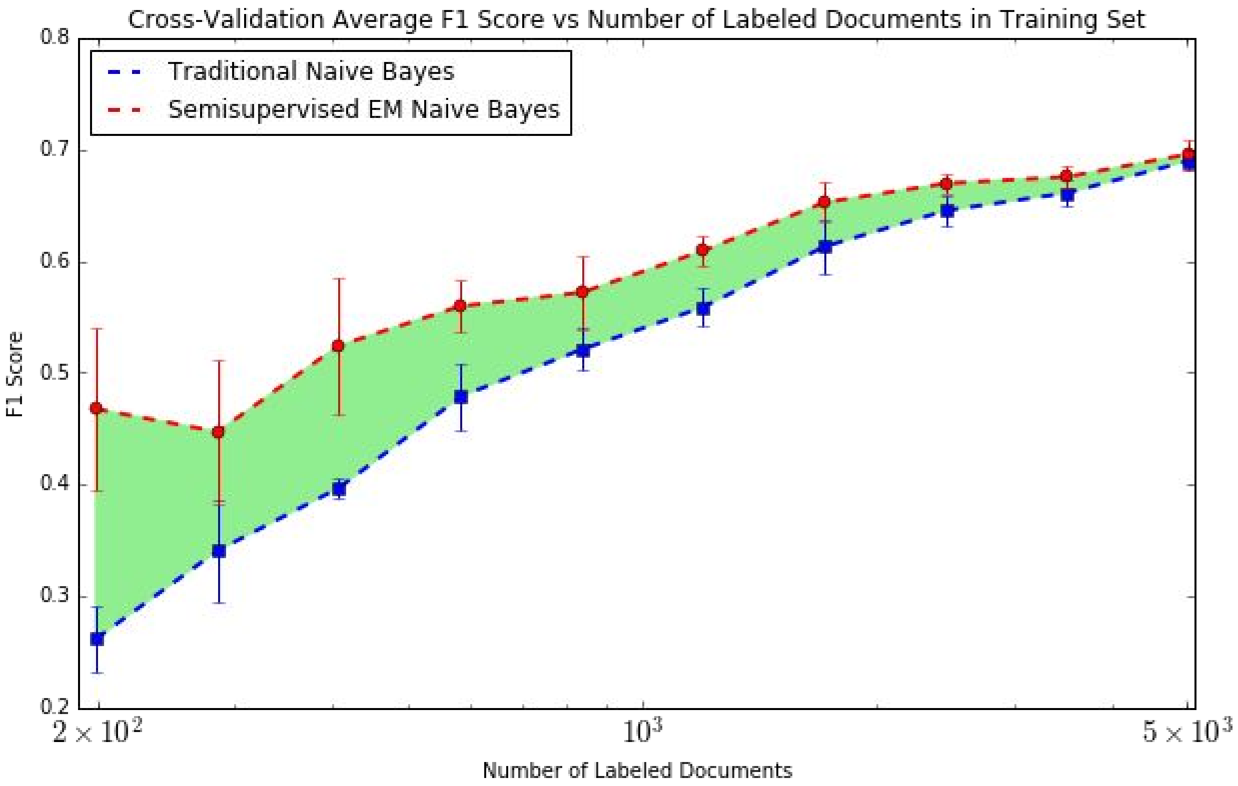
Huge improvement for small labeled dataset



Unlabeled data cannot help once the data sizes are at the same order.

Result and Analysis

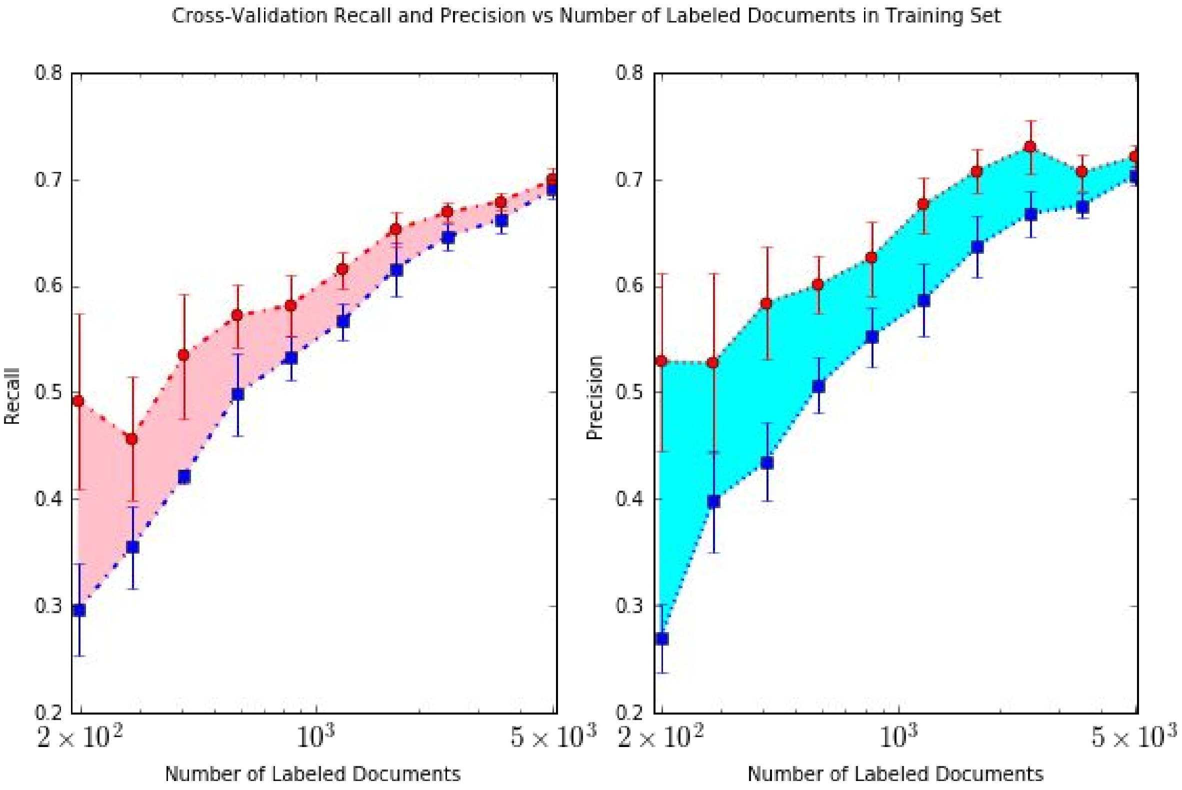
Stability is improved as labeled dataset increases. (Better EM initialization)



Performance is not stable at small labeled size

Result and Analysis

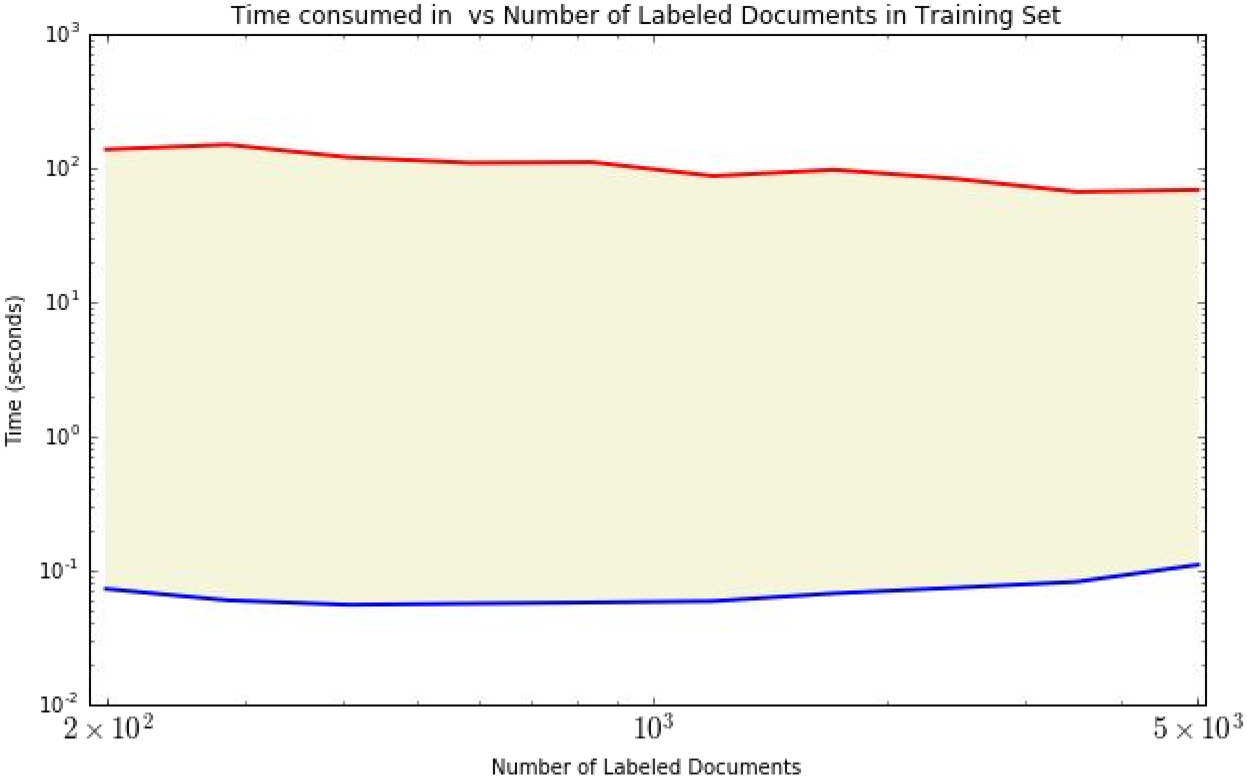
Both recall and precision perform better.



Still problem of instability

Result and Analysis

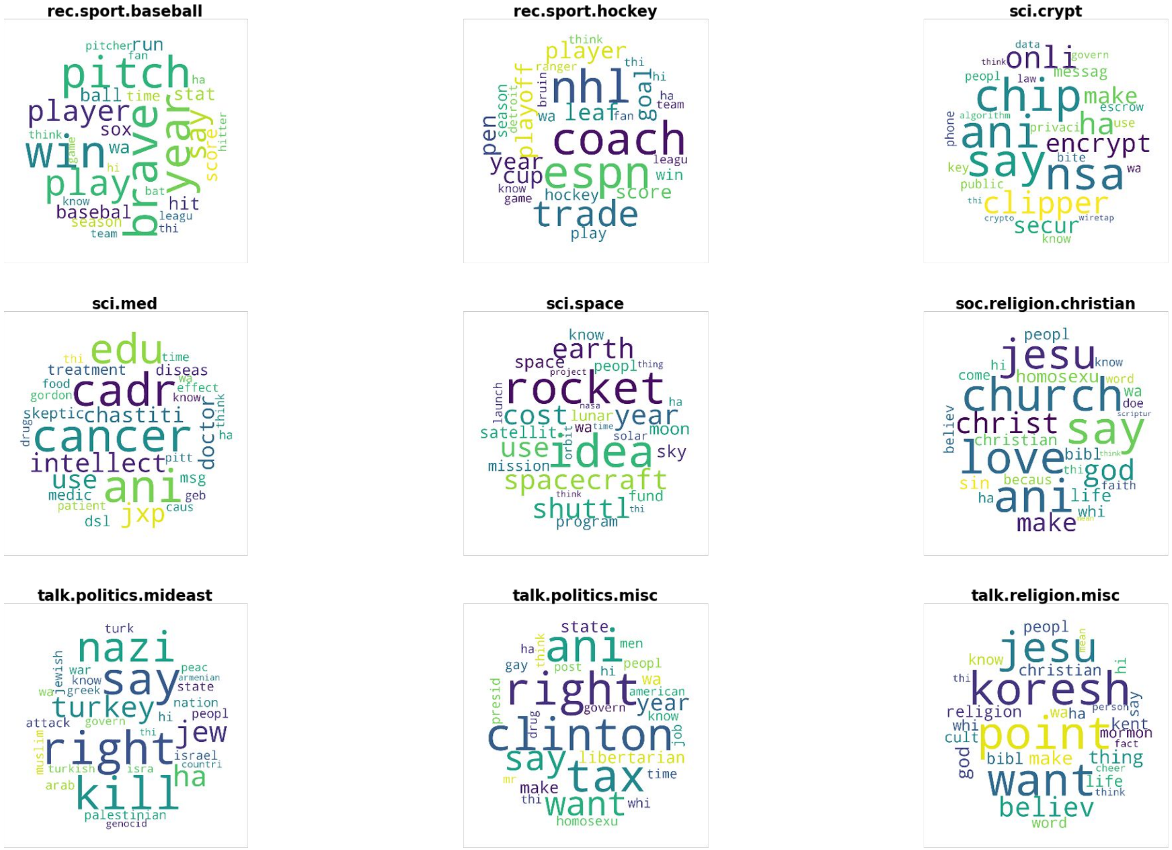
Time cost decreases as labeled dataset increases.



Really time consuming for huge unlabeled dataset

Result and Analysis

Explicit interpretability by most probable word feature.



Loss of context information and derived word form

Conclusion

Advantage:

- Massive cheap dataset in real case
- Significantly improve accuracy given small labeled dataset
- Labeled data for parameter initialization
- Simple implementation and parameter tuning
- Good scalability (online learning)
- Works well for categorical data (i.e. text in word vector)

Limitation:

- Strong assumption of feature independence
- Suffer noisy data distribution and highly correlated features
- Time consuming in computation of log likelihood over EM iterations

Thank You!
END

Natural Language Tool Kit (NLTK) Basic Text Analytics