Filtering Junk Mail with A Maximum Entropy Model

Zhang Le

www.nlplab.cn

Natural Language Processing Lab

Northeastern University, P.R. China

Organization

- Junk mail problem on the Internet
- Previous work on junk mail filtering
- Maximum Entropy Model
- Feature Selection
- Evaluation
- Compare with Naive Bayes
- Conclusion & Future Work

Junk Mail Problem

The increasing volume of junk mail (spams) has become the main problem concerned by email users. Junk mail has caused several problems:

- Money and time to sort through junk mails
- Causing network traffic, server overload, crashed mail-servers
- Social problems (pornography pic, unwanted adverts)

The task of junk mail filtering is to rule out unsolicited bulk mail automatically from a user's mail stream.

Previous Work

Since junk mail filtering can be re-casted as a Text Categorization task it is nature to apply known machine learning technologies to the task (Decision Tree, SVMs, Maximum Entropy Model etc.).

- RIPPER rule learning algorithm (Cohen, 1996)
- Bayes classifier (Sahami et al, 1998)
- Memory Based Learner (Androutsopoulos et al, 2000)
- Ada Boost algorithm (Carrera and Mrquez, 2001)

All these machine learning methods achieves a high junk precision & recall ($> 95\%$). The work presented here will focus on applying Maximum Entropy Model to the spam filtering task.

Maximum Entropy Model

Maximum Entropy (ME) Model is a general purpose machine learning framework that has been successfully applied to various NLP tasks:

- POS Tagging
- Text Categorization
- Text Chunking
- Shallow Parsing
- Statistical Language Modeling
- Statistical Machine Translation.

Maximum Entropy Model

Maximum Entropy (ME) Model is a general purpose machine learning framework that has been successfully applied to various NLP tasks:

- POS Tagging
- Text Categorization
- Text Chunking
- Shallow Parsing
- Statistical Language Modeling
- Statistical Machine Translation.

Given a set of features, and a set of constraints, ME model seeks for a model that minimizes the relative entropy (in the sense Divergence of Kullback-Leibler) $D(p||p_0)$.

ME Model (cont)

In general, a conditional ME model is an exponential (log-linear) model has the form:

$$p(y|x) = \frac{1}{Z(x)} \exp\left[\sum_{i=1}^{k} \lambda_i f_i(x,y)\right]$$

$$Z(x) = \sum_{y} \exp\left[\sum_{i=1}^{k} \lambda_i f_i(x,y)\right]$$

where k is the number of features and $Z(x)$ is a normalization factor to ensure that $\sum_y p(y|x) = 1$, also called partition function.

Features in ME model

Under ME framework, constraints imposed on a model are represented by features known as feature function in the form:

$$f(x,y) = \begin{cases} 1 & \text{if (x,y) satisfies certain constraint} \\ 0 & \text{otherwise} \end{cases}$$

Features in ME model

Under ME framework, constraints imposed on a model are represented by features known as feature function in the form:

$$f(x,y) = \begin{cases} 1 & \text{if (x,y) satisfies certain constraint} \\ 0 & \text{otherwise} \end{cases}$$

For example:

$$f_{free}(x,y) = \begin{cases} 1 & \text{if document } x \text{ contains word free} \\ 0 & \text{otherwise} \end{cases}$$

$$f_{javascript}(x,y) = \begin{cases} 1 & \text{if } x \text{ has a malicious javascript} \\ 0 & \text{otherwise} \end{cases}$$

Parameter Estimation of ME models

Several known methods exist for estimating the parameters (λ_i) of ME models:

- Iterative Scaling (GIS, IIS)
- First order methods (Steepest Ascent, Conjugate Gradient)
- Second order methods (Limited-Memory Variable Metric (L-BFGS))

Parameter Estimation of ME models

Several known methods exist for estimating the parameters (λ_i) of ME models:

- Iterative Scaling (GIS, IIS)
- First order methods (Steepest Ascent, Conjugate Gradient)
- Second order methods (Limited-Memory Variable Metric (L-BFGS)) most effective (Moulf, 2002)

Parameter Estimation of ME models

Several known methods exist for estimating the parameters (λ_i) of ME models:

- Iterative Scaling (GIS, IIS)
- First order methods (Steepest Ascent, Conjugate Gradient)
- Second order methods (Limited-Memory Variable Metric (L-BFGS)) most effective (Moulf, 2002)

Overfitting:

- held-out data
- smoothing (Gaussian Prior) $\frac{1}{\sqrt{2\pi\sigma_i^2}} \exp\left(-\frac{\lambda_i^2}{2\sigma_i^2}\right)$

Selecting Features

- Term Feature:
  - do not stem word
  - special HTML tags are preserved (url, ip address. . .)
  - take account of term position

Selecting Features

- Term Feature:
  - do not stem word
  - special HTML tags are preserved (url, ip address. . .)
  - take account of term position
- Domain Specific Feature:
  - mail header fields (X-Mailer)
  - non-textual features (Java Script, Color, Font. . .) (spamassassin.org)

Selecting Features

- Term Feature:
  - do not stem word
  - special HTML tags are preserved (url, ip address. . .)
  - take account of term position
- Domain Specific Feature:
  - mail header fields (X-Mailer)
  - non-textual features (Java Script, Color, Font. . .) (spamassassin.org)
- χ^2 Tests

Evaluation

We performed experiments on a public spam corpus, which contains 9351 messages of which: 2400 are labeled as spam and 6951 are marked as legitimate (ham), with a spam rate 25.7%.

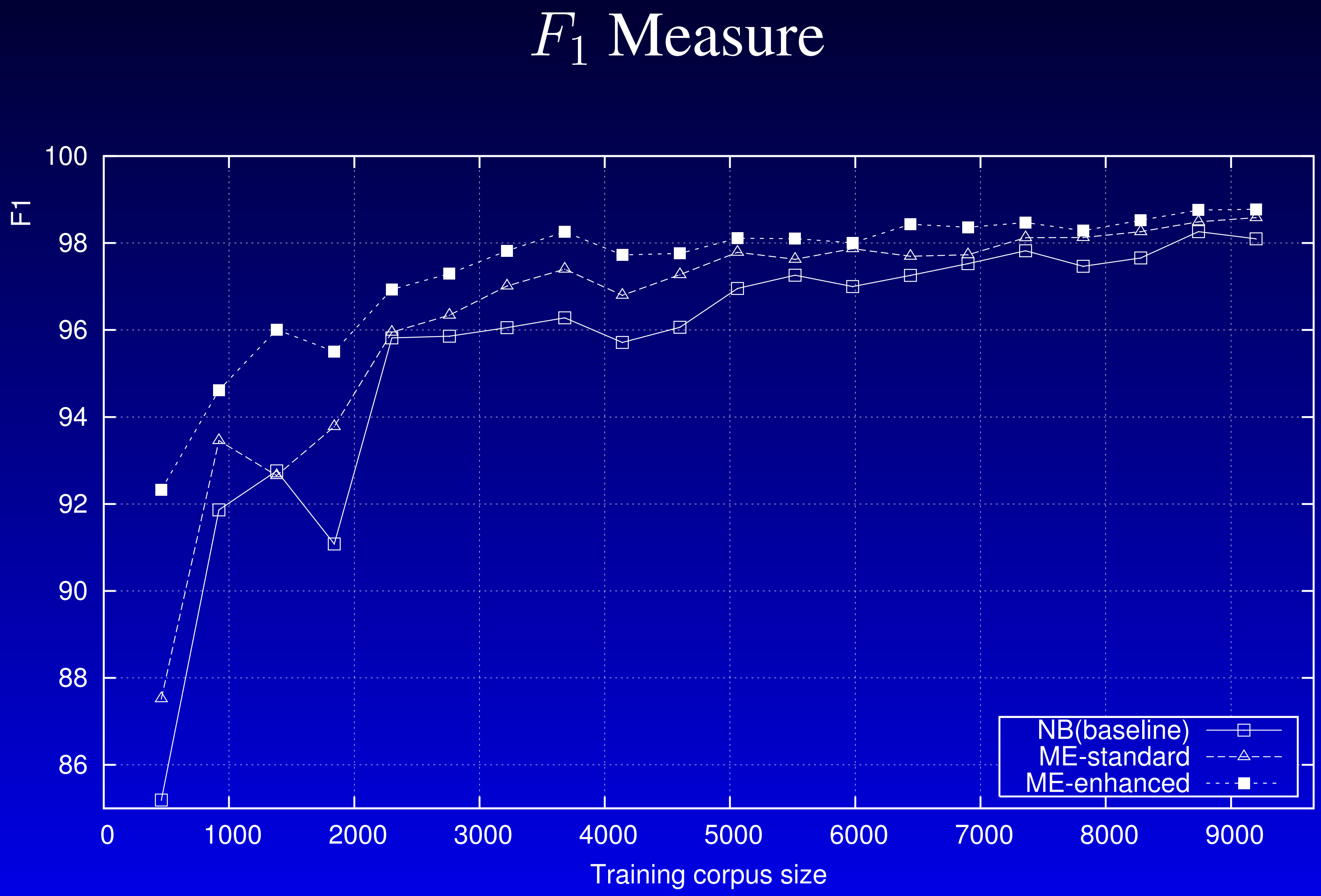
model	junk precision	junk recall	error-rate	F_1
NB(baseline)	99.67%	96.58%	0.98%	98.09%
ME	99.83%(0.16%)	97.37%(0.82%)	0.73%(-25.51%)	98.59%(0.51%)
ME-enhanced	99.83%(0.16%)	97.74%(1.20%)	0.63%(-35.71%)	98.77%(0.69%)

Table 0: Filtering performance of different models

(the number in parenthesis indicates improvements over baseline NB model)

Performance (Error Rate)

Performance (F_1 Measure)



Why Better Than Naive Bayes

Bayes Law:

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

Why Better Than Naive Bayes

Bayes Law:

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

Conditional independence assumption:

$$p(x|y) = p(x_1, x_2, \ldots, x_n|y) \approx \prod_{i=1}^{n} p(x_i|y)$$

Why Better Than Naive Bayes

Bayes Law:

$$p(y|x) = \frac{p(x|y)p(y)}{p(x)}$$

Conditional independence assumption:

$$p(x|y) = p(x_1, x_2, \ldots, x_n|y) \approx \prod_{i=1}^{n} p(x_i|y)$$

The ME model's ability of freely incorporating evidence from different sources makes it perform better than Naive Bayes classifier, which suffers from strong conditional independence assumptions.

Conclusion

Strength of ME model:

- knowledge-poor features
- reusable software
- free incorporation of overlapping and interdependent features

Conclusion

Strength of ME model:

- knowledge-poor features
- reusable software
- free incorporation of overlapping and interdependent features

Weakness of ME model:

- slow training procedure
- can not do increment learning (like Bayes and MBL)
- no explicit controls on parameter variance (like SVMs), to control false positive rate

Future Work

- More sophisticated features (variable length n-gram sequence, triggers. . .)
- Shallow parsing model
- Compare with other ML framework (Ada Boost, SVMs)

The End