Using Maximum Entropy for Text Classification

Kamal Nigam, John Lafferty, Andrew McCallum
(IJCAI-99)
presented by Ari Frank

1

Talk Outline

- What is the Maximum Entropy Principle?
- Maximum Entropy models for text classification.
- Learning Maximum Entropy Models.
- Experimental results.
- Conclusion.

2

The Essence of Maximum Entropy

- Maximum Entropy is a technique for learning probability distributions from data.
- "Don't assume anything about your probability distribution other than what you have observed."
- Always choose the most uniform distribution subject to the observed constraints.

3

Simple Classification Example

- An expert can classify documents into 4 classes: economics, sports, politics, art.
- The training data is a set of documents; each document is represented by a vector of words.
- We want to construct a probability distribution that represents the documents.

4

First Model

- Each document must be classified into one of the classes, so :
  P(economics) + P(sports) + P(politics) + P(art) = 1
- Without additional information, choose the model that makes the least assumptions.
  P(economics)=P(sports)=P(politics) = P(art) = 0.25
- Least assumptions = Most Uniform.

5

Example Cont.

- Suppose that if the word "ball" appears in the text, then p(sports | ball) = 0.7 .
- How do we adjust the distribution?
  - p(sports | ball) =0.7
  - P(politics | ball)=0.1
  - P(economics | ball)=0.1
  - P(art | ball)=0.1

6

What about More Observations?

- How do we factor in additional constraints? P(politics | Bush) =0.8, P(sports | game) =0.6, P(economic | stock)=0.5, ...
- Maximum Entropy modeling lets us create a distribution that abides by all these constraints, while being as uniform as possible.

7

Why Try to be Uniform?

- Most uniform = Maximum Entropy.
- By making the distribution as uniform as possible, we don't make any additional assumptions to what is supported by the data.
- Matches intuition of how probability distributions should be estimated from data.
- Abides by the principle of Occam's Razor (least assumptions made = simplest explanation).

8

Maximum Entropy Modeling for Text Classification

- Our training data is N pairs $\{(d_1,c_1),\ldots,(d_N,c_N)\}$
  - $c_i \in C$ classes of documents.
  - $d_i \in D$ – Set of documents. Each document is represented as a vector of word counts.
- The training set is renamed the "empirical" distribution

$$\tilde{p}(d,c) \equiv \frac{1}{N} \times \text{the number of times (d,c) appears}$$

- We want to create a stochastic model for $p(c \mid d) = \dfrac{p(d,c)}{p(d)}$

9

Feature Functions

- Features are used to capture relevant aspects of the training data.
- For example, a binary feature describing the appearance of the word ball in a sports document:

$$f_{sprots,ball}(d,c) = \begin{cases} 1 & \text{if c = sports and 'ball' appears in d} \\ 0 & \text{otherwise} \end{cases}$$

10

Feature Functions cont.

- In this paper scaled real valued features are used:

$$f_{w,c'}(d,c) = \begin{cases} 0 & \text{if } c \neq c' \\ \frac{N(d,w)}{N(d)} & \text{Otherwise} \end{cases}$$

- Gives better results than binary features.

11

Statistics

- The expected values of a feature with respect to the empirical distribution is

$$\tilde{p}(f) \equiv \sum_{d,c} \tilde{p}(d,c) \cdot f(d,c)$$

- Likewise, the expected value of a feature with respect to our model p is

$$p(f) \equiv \sum_{d,c} \tilde{p}(d)\, p(c \mid d) \cdot f(d,c)$$

12

Constraints

- Important statistics are used to shape the model, by forcing the model to comply with them:

$$p(f) = \tilde{p}(f)$$

- This means that the following should hold for every feature *f*:

$$\sum_{d,c} \tilde{p}(d)\,p(c \mid d) \cdot f(d,c) = \sum_{d,c} \tilde{p}(d,c) \cdot f(d,c)$$

13

Selecting a Model

- There can be an infinite number of models that satisfy a set of constraints.
- The maximum entropy principle dictates we select the most uniform model that satisfies the constraints.
- Uniformity is measured in terms of the conditional entropy of p(c | d):

$$H(p) \equiv -\sum_{d,c} \tilde{p}(d)\,p(c \mid d) \log p(c \mid d)$$

14

The Maximum Entropy Model

- From the set of allowed probability distributions, we select the model p* that maximizes H(p).
- p* can always be expressed in an exponential form:

$$p(c \mid d) = \frac{1}{Z(d)} \exp\left(\sum_i \lambda_i f_i(d,c)\right)$$

  - λ_i - weight parameters to be estimated
  - Z(d) – normalizing constant:

$$Z(d) = \sum_c \exp\left(\sum_i \lambda_i f_i(d,c)\right)$$

15

Properties of p*

- When p* is selected from the exponential family of distributions, we are guaranteed that:
  - p* is always well-defined, and unique.
  - p* also maximizes the conditional log likelihood of the data:

$$L_{\tilde{p}}(p) \equiv \sum_{d,c} \tilde{p}(d,c) \log p(c \mid d)$$

  - The likelihood space for the parameters of p is **convex with one global maximum** (unlike the typical likelihood surface for EM).

16

Improved Iterative Scaling

1. Start with $\lambda_i=0$ for all $i \in \{1,2,...,n\}$
2. Do for each $i \in \{1,2,...,n\}$
   a. Let $\Delta\lambda_i$ be the solution to

$$\left[\sum_{d,c} \tilde{p}(d)\,p(c \mid d)\,f_i(d,c)\right] \exp(\Delta\lambda_i f^{\#}(d,c)) = \tilde{p}(f_i)$$

$p_\lambda(f)$ ← Where $f^{\#}(d,c) \equiv \sum_i f_i(d,c)$ → rescaling factor

   b. $\lambda_i = \lambda_i + \Delta\lambda_i$
3. Go to Step 2 if not all λ_i have converged

17

Feature Functions: Reminder

- In this paper scaled real valued features are used:

$$f_{w,c'}(d,c) = \begin{cases} 0 & \text{if } c \neq c' \\ \frac{N(d,w)}{N(d)} & \text{Otherwise} \end{cases}$$

- Has the useful property that

$$f^{\#}(d,c) \equiv \sum_{w,c'} f_{w,c'}(d,c) = 1 = \frac{N(d)}{N(d)}$$

18

IIS cont.

- When $f^{\#}(d,c)=1$ for all c,d (as is the case in our model), $\Delta\lambda_i$ can be calculated in closed form:

$$\Delta\lambda_i = \log \frac{\tilde{p}(f_i)}{p_\lambda(f_i)}$$

- However, in the general case we must use numerical methods to calculate $\Delta\lambda_i$.

19

Toy IIS Example

- 2 classes (c_1=politics, c_2=art) :
  - politics: d_1=<the>
  - art: d_2=<the Monet>, d_3=<the Monet Painting>
- empirical distribution:

$$\tilde{p}(d,c) = \begin{cases} \frac{1}{3} & \text{if } (d_1,c_1) \text{ or } (d_2,c_2) \text{ or } (d_3,c_2) \\ 0 & \text{otherwise} \end{cases}$$

20

Example cont.

- Six features (3 words x 2 classes).
- Empirical expected feature values $\tilde{p}(f_{w,c'})$

	the	Monet	Painting
politics	1*1/3= **1/3**	**0**	**0**
art	1/2*1/3+ 1/3*1/3= **5/18**	1/2*1/3+ 1/3*1/3= **5/18**	1/3*1/3= **1/9**

21

Results of IIS iterations

Round 0:	$\lambda_{w,c'}$			$p(f_{w,c'})$		
	the	Monet	painting	the	Monet	painting
politics	0	0	0	0.1666	0	0
art	0	0	0	0.1388	0.1388	0.0555

Round 1:	$\lambda_{w,c'}$			$p(f_{w,c'})$		
	the	Monet	painting	the	Monet	painting
politics	0.087	0	0	0.295	0	0
art	-0.062	0.726	0.601	0.3158	0.1615	0.066

22

Round 2:	$\lambda_{w,c'}$			$p(f_{w,c'})$		
	the	Monet	painting	the	Monet	painting
politics	0.208	0	0	0.2973	0	0
art	-0.147	1.306	1.069	0.313	0.176	0.073

Round 5:	$\lambda_{w,c'}$			$p(f_{w,c'})$		
	the	Monet	painting	the	Monet	painting
politics	0.506	0	0	0.3097	0	0
art	-0.368	2.58	2.04	0.3013	0.203	0.086

Round 500:	$\lambda_{w,c'}$			$p(f_{w,c'})$		
	the	Monet	painting	the	Monet	painting
politics	2.574	0	0	0.333	0	0
art	-2.56	14.73	8.45	0.278	0.276	0.1108

23

Classification

- At convergence, the weights $\lambda_{w,c'}$ yield the following distribution for $p(c|d)$:

	<the>	<the Monet>	<the Monet painting>
politics	0.99985	0.00021	0.00005
art	0.00015	0.99979	0.99995

24

Adding a Prior

- Maximum Entropy models can suffer from overfitting.
- With sparse data the observed feature statistics can be far from the true values.
- A $N\sim(0,\sigma^2)$ prior probability for the weights λ_i is added to the model as a regularization term.
- With sparse data a small variance is used (so feature weights are forced towards 0).

25

Experiments

- The performance was tested on 3 datasets.
- Both with and without the Gaussian prior.
- Results are compared to multinomial Naïve Bayes classifiers (both regular and scaled).

26

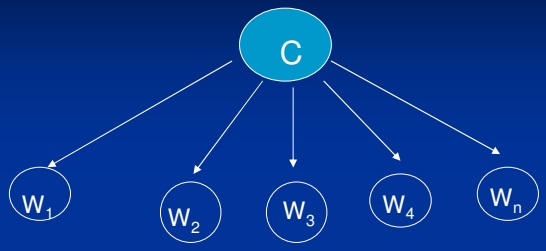
Feature Selection

- For each experiment, results with the optimal number of features are reported.
- Features are ranked and added according to mutual information with the class label:

$$I(c, w_i) = \sum_j \sum_{t=0,1,..} \tilde{p}(w_i = t, c_j) \log \frac{\tilde{p}(w_i = t, c_j)}{\tilde{p}(w_i = t) \cdot \tilde{p}(c_j)}$$

27

Naïve Bayes Classifiers



- $p(c_j)$, and $p(w_i \mid c_j)$ are estimated from the data.
- Classification according to Bayes' rule:

$$p(c_j \mid w_1, ..., w_n) = \frac{p(c_j) \prod_{i=1}^{n} p(w_i \mid c_j)}{\sum_j p(c_j) \prod_{i=1}^{n} p(w_i \mid c_j)}$$

28

Naïve Bayes cont.

- The values w_i are the word counts in the Naïve Bayes.
- In the scaled Naïve Bayes, all word counts are scaled so all documents have the same number of words.
- Naïve Bayes makes the assumption of independence between features.

29

Data Sets

Name	Samples	Classes	Vocabulary
WebKB	4199	4	23830
Industry Sector	6440	71	29964
Newsgroup	20000	20	57040

- For WebKB 30% is held out for testing, for Industry and Newsgroup 35% is held out (for these datasets the extra 5% is used as a validation set to terminate the training of the IIS).

30

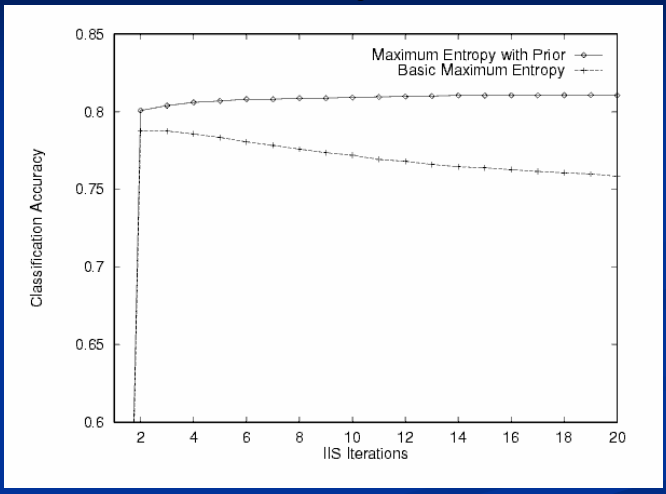
Error Rates on Holdout Sets

Data Set	NB	Scaled NB	Basic ME	ME w/prior
WebKB	13.69 (2000)	13.10 (5000)	7.92 (2000)	8.08 (2000)
Industry	28.97 (20000)	20.21 (29964)	21.14 (29964)	18.90 (29964)
Newsgroups	16.15 (57040)	14.43 (57040)	15.77 (57040)	15.14 (57040)

The number of features used is in parentheses.

31

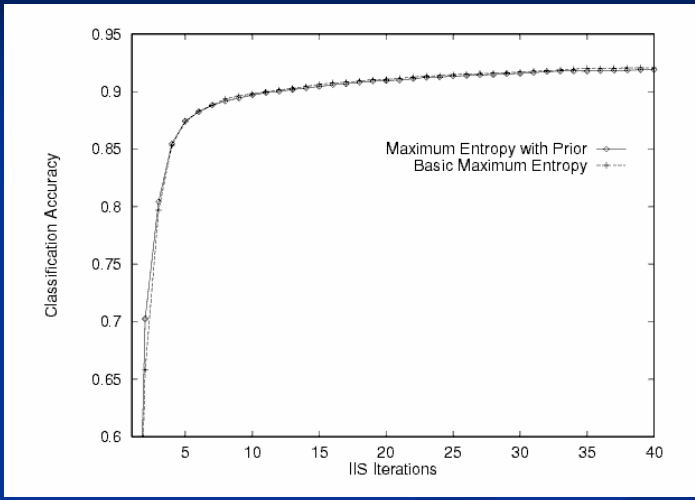
Industry Data



- The performance of ME without a prior deteriorates with increases IIS round due to overfitting.

32

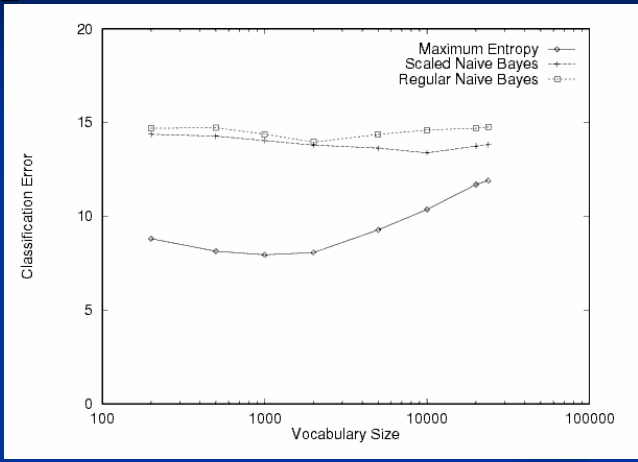
WebKB data



- No overfitting by the IIS with the WebKB data.

33

Importance of Feature Selection



- The maximum Entropy classifier without a prior is more prone to overfitting than Naïve Bayes.

34

Concluding Remarks

- Maximum Entropy is a method for learning distributions.
- The ME distribution is the most uniform one that complies with constraints determined from the training data, and makes no assumptions beyond them.
- The ME distribution is well defined and unique.
- It is the single maximum in a convex likelihood space, which makes it easy to find the optimal parameters.

35

More..

- The Maximum Entropy method has been shown to work well for text classification.
- In all cases it performed better than Naïve Bayes, and had mixed results comparable to scaled Naive Bayes.
- Simple greedy feature selection was used; more sophisticated methods can be employed.

36

Remarks about Features

- This experiment used only simple single word features.
- The Maximum Entropy framework allows for complex context dependent features:
  - Word pairs "Buenos Aires"
  - Boolean formulas – has "stock" but not "market"
- Maximum Entropy doesn't assume independence; it can accommodate overlapping and "redundant" features.

37

References

- Using Maximum Entropy for Text Classification
  Kamal Nigam, John Lafferty, Andrew McCallum. In IJCAI-99 Workshop on Machine Learning for Information Filtering, pages 61--67, 1999 (1999)
- A maximum entropy approach to natural language processing
  Adam Berger, Stephen Della Pietra, and Vincent Della Pietra. *Computational Linguistics*, (22-1), March 1996

38