Proj #7 Text Classification Report

Alexei Betin and Brent Payne

April 28, 2009

1 Bayesian classifier

The principle behind the Bayesian classifier is Bayes' rule, $p(y|x) = \frac{p(x|y) \cdot p(y)}{p(x)}$. In our setup, we take x to be word counts from a document and y to be a label. Then, $p(y)$ can be estimated by the frequency of y in the training data. $p(x)$ can be estimated as $\sum_y p(y|x)p(y)$. $p(y|x)$ can be estimated by any distribution that take multiple discrete values. We use the simplest such distribution, the multinomial distribution, Eqn. 1.

$$\frac{n!}{\Pi_{i=1}^m x_i!} \Pi_{i=1}^m \theta_i^{x_i}, \mathbf{x} \in \mathbb{R}^m, \|\mathbf{x}\|_1 = n \tag{1}$$

We fit θ_y to the maximum-likelihood of a training set of topic y documents. We add a smoothing constant, c, to each x_i to ensure no θ_i is zero. We set $c = 1/m$.

$$\theta_i^y = \sum_x \frac{x_i + c}{\|x\|_1 + m \cdot c} \tag{2}$$

Do to issue with underflow, we do not dirstly compute $p(y|x)$. Instead, we compute the $log[p(x|y)p(y)]$. Since the log function is monotonically increasing and $p(x)$ is constant across y, this metric can be used comparison between different labels. Additionally, we drop the multinomial coefficient from $p(x|y)$ since it is constant given an x.

This model performs very well recieving 99% accuracy on ten fold cross validation across the whole data set. Table 1 shows the union of the top five words per topic across the ten validation runs. The θ_i^y tell you how significant an occurance of a word x_i is to the topic y, thus is used for selecting the top five words per topic.

topic 1	topic 2	topic 3
patients	system	boundary
cases	scientific	mach
fatty	retrieval	layer
acids	research	wing
ventricular	science	supersonic
nickel	language	
left		
blood		

Table 1: Union of top five words per topic

2 SVM Classification

For data preparation, we transform the raw document counts in the input data to log counts using the formulae $x = log(1 + x)$. This gives a little higher accuracy for all the classifiers we can build, but matters especially when applying stronger regularization and/or feature reduction. The numeric results below are obtained on the transformed input data.

For SVM classification model, since we have to deal with 3 classes, we first dichotomize the target to create 3 new target variables$y_k = 1$ if $y = k$, 0 otherwise where $k = 1, 2, 3$, and y is the original target. We then build 3 SVM binary classifiers for each of the new targets y_k. We use RapidMiner FastLargeMargin learning algorithm with L2 SVM Primal Solver since this appears to be the fastest SVM learner available in RapidMiner. We first train the 3 binary classifiers on the non-transformed original input dataset using 10-fold Cross-Validation. We use GridParameterOptimization to find the optimal value of C=2. The binary classifiers achieve the following average accuracies using 10-fold Cross-Validation: For final classification, we apply all 3 classifiers to the

class 1:	$98.25\% \pm 1.60\%$
class 2:	$96.75\% \pm 2.75\%$
class 3:	$97.50\% \pm 2.24\%$

Table 2: one-sided Accuracy of SVMs

input dataset and then choose the class whose respective classifier produces the highest confidence. We achieve 100% accuracy on the input dataset. The model is likely to be overfit since it has more parameters than training examples.

To reduce overfitting and build a more robust generalized model, we then apply variable reduction by SVM Weighting before we train each of the binary classifiers. After experimenting with different numbers of attributes to keep, we find that we can still achieve very high classification accuracy using only 60 input variables to build all 3 binary classifiers. We also find that with fea-

ture reduction, it becomes much more advantageous to transform raw counts into log-counts according as described above. Again, we do GridParameterOptimization to find the optimal value of C, in this case we end up with C=10 so we use somewhat smaller regularization in the case of fewer model parameters. Table 2 displays the binary classifiers average accuracy following 10-fold Cross-Validation. The final classifier using 60 input variables (words) achieves

class 1:	$93.25\% \pm 4.48\%$
class 2:	$94.50\% \pm 2.69\%$
class 3:	$94.25\% \pm 2.97\%$

Table 3: one-sided accuracy of SVMs after feature selection

the accuracy of 96.5% on the input dataset.

	true 1	true 2	true 3	class precision
pred. 1	99	6	7	88.39%
pred. 2	0	94	0	100.00%
pred. 3	1	0	193	99.48%
class recall	99.00%	94.00%	96.50%	

Table 4: SVM confusion matrix

Looking at the final models, we can deduce what kind of documents is represented by each class. Namely, Class 1 appears to contain documents about clinical studies, Class 2 appears to contain documents about aeronautics, whereas Class 3 probably contains documents related to computer or information science.

The final reduced models are as follows:

topic 1	topic 2	topic 3
clinical	naca	technical
nickel	loading	semantic
patient	scientific	publications
flat	fluid	clinical
layer	measurements	science

Table 5: Top five words based on SVM weights

3 Bayesian classifier applied to CSE291 message board

In an attempt to apply this technique to a harder dataset, we apply a Bayesian classifier to authorship of messages on the CSE291 message board. We took the

first 184 messages and partitioned them into two groups: professor, student. The professor has written 97 of these messages, so the base rate of professor authorship is 52.7%. This is a much harder problem. It is complicated by professor and student messages that share the same topics and the professor quoting student messages. Thus stylistic information is more important. For this reason, we neither remove stop words or punctuation when tokenizing, nor do we smooth counts to ofset burstiness. After removing tokens that only occur in one message, we are left with 1370 unique tokens.

Directly applying the procedure used on the classic400 dataset, we achieve accuracy that is worse than baserate. To improve performance we do two things, select a feature set based on rank one decomposition, use bigram features. Since the messages are typically small, unigram frequency of tokens do not store enough style information to distinguish students and professor messages. Instead, we use bigrams of contiguous tokens as our feature set. The message board data has 2453 unique bigrams that appear in multiple messages. This gives us more than ten times the number of features as data.

To limit our feature set, we use a rank 1 decomposition of each topic to select features. We construct a n x m matrix A_y. The rows are the n messages associated with author y and the columns are the m bigrams. Then we choose a vector W that minimizes the squared error of, $A_y \cdot W - B$, where B is a ones vector. W is then the rank one decomposition of A_y. Since the topics are similar between professor and student documents, we want to choose the features that maximally distinguish them. Thus we do not want to chose features that are highly correlated with both the student and the professor messages. We take this into account by choosing the top 20[1] tokens based on $|W_{prof} - W_{student}|$. Table 8 show the list of selected bigrams.

This setup gives us a micro-average accuracy of 62.0% on ten fold validation with the combined confusion matrix of Table 6. Table 7 is the union of the top 5 most significant bigrams to our Bayesian classifiers.

	true prof.	true student	precision
pred. prof.	72	45	61.5%
pred. student.	25	42	62.7%
recall	74.2%	48.3%	acc. 62.0%

Table 6: message board confusion matrix

[1]twenty was choosen because it would give us roughly ten times the amount of training per parameter

professor	student
of—the	to—the
’—s	i—’
1—|	of—the
to—the	i—am
for—a	1—|
i—am	test—set

Table 7: significate bigrams

it—to	a—reasonable	?—/	will—the
and—i	kind—of	by—author	pm—/
as—the	would—be	can—we	of—the
i—am	i—’	of—ram	appears—to
as—i	the—notes	1—|	what—i
to—do	for—each	the—quiz	you—can
training—examples	p—=	on—a	don—’
see—also	for—a	seem—to	it—is
the—model	to—the	”—a	.—(
test—set	?—i	’—s	)—for

Table 8: 40 selected bigrams