An Evaluation of Text Classification Methods for Literary Study[1]

Bei Yu
Graduate School of Library and Information Science
University of Illinois at Urbana-Champaign

Abstract:

This article presents an empirical evaluation of text classification methods in literary domain. This study compared the performance of two popular algorithms, naïve Bayes and Support Vector Machines in two literary text classification tasks: the eroticism classification of Dickinson's poems and the sentimentalism classification of chapters in early American novels. The algorithms were also combined with three text pre-processing tools, namely stemming, stopword removal, and statistical feature selection, to study the impact of these tools on the classifiers' performance in the literary setting. Existing studies outside the literary domain indicated that SVMs are generally better than naïve Bayes classifiers. However, in this study SVMs were not all winners. Both algorithms achieved high accuracies in sentimental chapter classification, but the naïve Bayes classifier outperformed the SVM classifier in erotic poem classification. Self feature selection helped both algorithms improve their performance in both tasks. However, the two algorithms selected relevant features in different frequency ranges, and therefore captured different characteristics of the target classes. The evaluation results in this study also suggest that arbitrary feature reduction steps such as stemming and stopword removal should be taken very carefully. Some stopwords were highly discriminative features for Dickinson's erotic poem classification. In sentimental chapter classification stemming undermined subsequent feature selection by aggressively conflating and neutralizing discriminative features.

Correspondence:

Bei Yu, Kellogg School of Management, Northwestern University, Evanston, IL 60208. Email: bei-yu@northwestern.edu

[1] This research is supported in part by the Nora project (http://www.noraproject.org/). Generous support from the Nora team led by Professor John Unsworth is gratefully acknowledged. Special thanks to Drs. Mark Olsen and Steve Ramsay for their discussions.

1 Introduction

Text classification is a typical scholarly activity in literary study (Unsworth, 2000; Yu and Unsworth, 2006). Humanist scholars organize and study literary texts according to various classification criteria, such as topics, authors, styles, and genres. For decades computational analysis tools have been used in some literary text classification tasks, such as authorship attribution (Mosteller and Wallace, 1964; Holmes, 1994) and stylistic analysis (Holmes, 1998). Recently, with the development of machine learning and natural language processing techniques, automatic text classification methods[2] provide new approaches to more literary text analysis problems (Argamon and Olsen, 2006), for example the discriminant analysis and cross entropy classification for authorship attribution and stylistic analysis (Craig, 1999; Juola and Bayyen, 2005), the decision tree classification for genre analysis of Shakespeare's plays (Ramsay, 2004), the SVM classification for knowledge class assignment of the Encyclopédie entries (Horton *et al.*, 2007), the naïve Bayes classification for the eroticism analysis of Dickinson's poems (Plaisant *et al.*, 2006), and the naïve Bayes classification for sentimentalism analysis of early American novels (Horton *et al.*, 2006).

With the availability of so many text classification methods, empirical evaluation is important to provide guidance for method selection in literary text classification applications. A number of studies have evaluated popular classification algorithms on a few benchmark data sets outside the literary domain (Dumais *et al.*, 1998; Joachims, 1998; Yang and Liu, 1999). However, these benchmark data sets were limited to news and web documents, which have different

[2] As a supervised learning approach, automatic text classification involves two steps. A classifier is firstly trained on some examples with pre-assigned class membership labels (training examples), and then the classifier is used to predict the classes of new examples (test examples). See (Sebastiani, 2002) for a comprehensive survey of text classification methods.

characteristics from the creative writings in literature. Moreover, in these evaluation studies, all methods were tested on topic classification tasks. In the setting of literary text classification, text documents are categorized by many document properties other than topics. Some target classes, such as authors and genres, are defined in an objective manner, while other classes, such as the sub-genres "eroticism" and "sentimentalism", are subjectively defined by the groups of scholars in these particular fields of study. Prediction is the common purpose of scientific classifiers. Hence classifiers are usually evaluated by the measure of classification accuracy. However, improving classification accuracy is seldom the goal for literary scholars. High classification accuracy provides evidence that some patterns have been inferred to separate the classes. The scholars are more interested in the literary knowledge as represented by these linguistic patterns. In other words the usual purpose of literary classification is to seek suggestive evidence for further scholarly investigation of what texture features characterize the target classes (Ramsay, 2008). Sometimes scholars would also like to have classifiers as example-based retrieval tools to find more documents of a certain kind, such as ekphrastic poems[3] and historicist catalog poems[4] (Yu and Unsworth, 2006). In these cases only a small number of training examples are available, which requires the classifiers to learn fast and accurately. Facing the unique characteristics of literary text classification applications, we have to think about the question whether the existing conclusions on classification method comparison still hold for literary text classification tasks.

This article describes an empirical evaluation of text classification methods in the literary domain. Based on the above use scenarios this study evaluates the classification methods from three perspectives: classification accuracy, literary knowledge discovery, and potential for

[3] Heffernan (2004) defined ekphrasis as the "literary representation of visual art". An ekphrastic poem is written in response to all kinds of artworks, including drawings, paintings, sculpture, dance, movie, *etc.*

4 Professor Ted Underwood describes it as "a speaker looks at an object (the moon or the sea, or whatever) and thinks about all the different civilizations that have seen the same object, imagining what they may have felt, and implicitly contrasting them to the present."

example-based retrieval. Because no benchmark data is available in this domain, the methods are compared on two specific sub-genre classification tasks as case studies, both focusing on identifying certain kinds of emotion, a document property other than topic.

The first task is the eroticism classification of Emily Dickinson's poems. The debate over what counts as and constitutes the erotic in Dickinson has been a primary research problem in Dickinson studies for the last half century (Plaisant *et al.*, 2006). To study the erotic language patterns in Dickinson's poems, a group of Dickinson scholars at University of Maryland at College Park compiled a Dickinson erotic poem collection which consists of 269 XML-encoded letters comprising nearly all the correspondence between the poet Emily Dickinson and Susan Huntington (Gilbert) Dickinson, her sister-in-law. The long letters which involve both erotic and not-erotic contents were excluded from the collection. The scholars assessed the 269 letters as either erotic or not[5]. Eventually 102 letters were labeled as erotic (positive), and 167 not-erotic (negative).

The second task is the sentimentalism classification of chapters in early American novels. Although academic study of sentimental fiction has been well accepted in the past few decades, academic disagreement persists about what constitutes textual sentimentality and how to examine sentimental texts in serious criticism (Horton *et al.*, 2006). To explore what linguistic patterns characterize the subgenre of sentimentalism, two literary scholars at the University of Virginia constructed a collection of five novels in the mid-nineteenth century sentimental period which are generally considered to exhibit sentimental features: Uncle Tom's Cabin, Incidents in the Life of a Slave Girl, Charlotte: a Tale of Truth, Charlotte's Daughter, and The Minister's wooing. The scholars assessed the sentimentality level of each of the 184 chapters as either "high" or "low". Among them 95 chapters were labeled as "high" and 89 as "low".

[5] The scholars discussed and resolved the disagreements during the assessment.

Two popular text classification algorithms, naïve Bayes and Support Vector Machines (SVMs), are chosen as the subjects of evaluation. Existing studies indicate that SVMs are among the best text classifiers to date (Dumais *et al.*, 1998; Joachims, 1998; Yang and Liu, 1999). Naïve Bayes is a simple but effective Bayesian learning method (Domingos and Pazzani, 1997), often used as a baseline algorithm. This study compares the performance of these two algorithms on eroticism classification and sentimentalism classification tasks.

Algorithm selection is not the only factor which affects the classification result. The choice of text representation models and text pre-processing options also influence the classification performance. The simplest bag-of-words (BOW) model is often used for text representation when no prior knowledge is available with regard to specific classification tasks. In fact a number of studies have shown that complex features did not help statistical classifiers gain significant performance improvement (Cohen, 1995; Dumais *et al.*, 1998; Lewis, 1992; Scott and Matwin, 1999). Under the bag-of-words model a text document is converted into a vector of word counts. Without feature reduction, a document vector is often defined in a space of thousands of dimensions, each dimension corresponding to a word feature. In such a high-dimensional space many features are of low relevance. Feature reduction is important in order to train classifiers with good generalizability as well as reducing the computation cost. Stemming, stopword removal, and statistical feature selection are three common feature reduction tools in text classification. Studies have shown that in some situations, these tools could interact with classification methods, and consequently affect the classifiers' performance (McCallum and Nigam, 1998; Mladenic and Grobelnik, 1999; Mladenic *et al.*, 2004; Riloff, 1995; Scott and Matwin, 1999). Based on the above considerations, this study combines naïve Bayes and SVM

algorithms with different choices of feature reduction tools, and then examines whether these choices affect the algorithms' performance in literary text classification tasks.

The rest of this article is organized as follows. Section 2 describes the text classification methods, the feature reduction tools and the evaluation measures used in this study. Section 3 describes the design of the evaluation experiments. Section 4 and 5 report the evaluation results in the eroticism classification and the sentimentalism classification tasks respectively. Section 6 concludes with discussions of the evaluation results across the two case studies.

2 Classification methods, feature reduction tools and evaluation measures

2.1 Naïve Bayes and SVM classifiers

Naïve Bayes is a highly practical Bayesian learning method. It assumes that the feature values are conditionally independent given the target value, and therefore significantly reduces the computation cost (Mitchell, 1997). Although real world data (e.g. text data) often violate this assumption, naïve Bayes classifier can still be optimal under zero-one loss even when the independence assumption is violated by a wide margin (Domingos and Pazzani, 1997). As a simple but effective method, naïve Bayes is often included in comparative evaluation of text classification methods (Dumais *et al.*, 1998; Joachims, 1998; Sebastiani, 2002; Yang and Liu, 1999).

The naïve Bayes algorithm can be implemented in various ways. Two naïve Bayes variations are widely used in text classification; they are called the multi-variate Bernoulli model and the multinomial model (McCallum and Nigam, 1998). The multi-variate Bernoulli model (abbreviated as "nb-bool" in this article) uses word presence or absence (one or zero) as feature

values (Boolean). The multinomial model (abbreviated as "nb-tf") uses word frequencies as feature values. Previous studies on topic classification tasks showed that the multi-variate Bernoulli model is more suitable for data sets with small vocabularies, while the multinomial model is better on larger vocabularies (Lewis, 1998; McCallum and Nigam, 1998). However, recent studies demonstrate that naïve Bayes classifiers with word presence/absence values performed better in predicting opinion polarities of movie reviews (Pang *et al.*, 2002). In this study both target classes (eroticism and sentimentalism) are related to emotion, therefore both naïve Bayes variations are implemented and compared based on the description in Mitchell (1997).

SVMs are a family of supervised learning methods developed by Vapnik *et al.* based on the Structural Risk Minimization principle from statistical learning theory (Vapnik, 1982;1999). As linear classifiers (with linear kernel), SVMs aim to find the hyperplanes that separate data points with the maximal margins between the two decision boundaries. Aiming to minimize the generalization error, SVMs have the advantage of reducing the risk of overfitting. SVMs outperform other text classification methods in a number of comparative evaluations on topic classification tasks (Dumais *et al.*, 1998; Joachims, 1998; Yang and Liu, 1999).

The SVM algorithm also allow for various kinds of word frequency measures as feature values, which results in multiple variations. In this study the SVM algorithm is combined with four candidate text representations. The first one is "svm-bool", which uses word presence or absence as feature value. The second one is "svm-tf", which uses word (term) frequency as feature value. The third one is "svm-ntf", which uses normalized word frequency as feature value.

The last one is "svm-tfidf", which uses term frequency weighted by inverse document frequency as feature value. The SVM-light package[6] and its default parameter settings are used in this study.

Table 1 summarizes the combinations of classification algorithms and text representation models. For each algorithm the variation with the best performance in the initial evaluation experiment will be used in the following experiments.

Table 1: variations of SVM and naïve Bayes classification methods

	Feature values			
Algorithms	word presence/absence	original term frequency	normalized term frequency	idf-weighted term frequency
SVM	svm-bool	svm-tf	svm-ntf	svm-tfidf
naïve Bayes	nb-bool	nb-tf	n/a	n/a

2.2 Stemming

In text classification the stemming process conflates a group of inflected words with the same stems into one single feature, assuming that they bear similar meanings. However, sometimes different forms of the same word contribute to the classification in different ways. For example, distinguishing the singular and plural forms of nouns and different verb tenses improved terrorism document classification (Riloff, 1995). Hence stemming might affect text classification in both positive and negative ways. (Scott and Matwin, 1999; Sebastiani, 2002). This study uses the Porter Stemmer (Porter, 1980) to stem words. Complementary look-up tables for irregular nouns and verbs[7] are also used because the Porter stemmer does not take into consideration irregular nouns and verbs.

[6] This software can be downloaded from http://svmlight.joachims.org/.
[7] An irregular verb list is obtained from http://www.learnenglish.de/Level1/IRREGULARVERBS.htm, and an irregular noun list from http://www.esldesk.com/esl-quizzes/irregular-nouns/irregular-nouns.htm.

2.3 The role of stopwords

In information retrieval, stopwords mean extremely common words, such as "the" and "of", which are considered useless and then removed from the queries and the document (Baeza-Yates and Ribeiro-Neto, 1999). Since common words are mostly function words, the concepts "common words" and "function words" are usually considered as synonyms. But they are actually overlapping but not equivalent concepts. "Common words" are defined and selected based on word frequencies in a specific collection. A common word in one collection might not be common in another one. Function words are "closed-class" word groups with constant members. They do not carry concrete meaning, but they have important role in grammar. Function words proved to be useful for some text classification tasks. For example the pronoun "my" is a very useful word feature to identify student homepages (McCallum and Nigam, 1998). Prepositions help identify joint venture documents (Riloff, 1995). Function words are even the major stylistic markers in genre analysis, stylistic analysis and authorship attribution (Argamon, Biber, 1988, 1995; Holmes, 1994; Saric and Stein, 2003). This study tests the effect of stopwords on literary text classification based on both "common words" and "function words" definitions.

2.4 Statistical feature selection

Stemming and stopword removal are "arbitrary" feature reduction tools regardless of classification tasks. Statistical feature selection methods measure the weights of features based on their relevance to the classes and select the features with heaviest weights. Feature selection methods are often used as pre-processing steps before classification because they are assumed to be independent of classification methods (Yang and Pedersen, 1997; Joachims 1998). However, Mladenic and Grobelnik (1999) have found that feature selection methods could interact with classification methods. For example, information gain has negative effects on naïve Bayes

classifiers, while Odds ratio fits naïve Bayes classifiers best. Forman (2003) found that no feature selection methods can improve the performance of SVM classifiers. Because both SVMs and naïve Bayes classifiers are linear classifiers, each of their feature has a weight (coefficient) in the linear decision functions. Therefore both SVM and naïve Bayes algorithms can be used as feature selection methods as well (Guyon, 2002; Mladenic *et al.*, 2004). The feature weighting function in naïve Bayes algorithm is actually the same as Odds ratio (Mladenic and Grobelnik, 1999). This study uses SVM and naïve Bayes algorithms themselves as feature selection methods.

2.5 Classification evaluation methods

Cross validation and hold-out tests are the usual methods for classification result evaluation. N-fold cross validation splits a data set into N folds and runs classification experiment N times. Each time one fold of data is used as test set and the classifier is trained on the other N-1 folds of data. The classification accuracy is averaged over the results of N runs. Hold-out test divides a data set into a training subset and a test subset. A classifier is trained on the training subset and tested on the test subset. For data sets with a small number of examples, an arbitrary split would result in both small training and test sets, potentially yielding varied results for different ways of splitting. Both of the data sets in this study have no more than two hundred examples, therefore 10-fold cross validation is used to evaluate the classifiers. Paired *t*-test is used to measure the significance of accuracy differences (α=0.05). In the case of comparing multiple means Bonferroni correction is used to adjust the significance level in individual comparison (Bland and Altman, 1995).

3 Experiment design

A series of experiments are designed to test the performance of naïve Bayes and SVM algorithms combined with different feature reduction tools. The following experiments are run for both eroticism classification and sentimentalism classification tasks.

3.1 Experiment 1: document representation model selection

The purpose of this experiment is to choose the best text representation model for each algorithm to use in the following experiments. Without prior knowledge, the initial feature set for the eroticism classification is the full vocabulary excluding the words occurred only once. According to the scholars' domain knowledge, the initial feature set for the sentimentalism classification is the content words - nouns (except proper nouns), verbs, adjectives and adverbs. Rare words (frequency<5) are excluded from the vocabulary. The Brill part-of-speech tagger (Brill, 1995) is used to extract the content words.

3.2 Experiment 2: using stopwords as feature sets

This experiment evaluates the usefulness of stopwords in the two classification tasks. There are two ways to evaluate the contribution of stopwords to classification. The first approach compares the accuracies before and after removing stopwords from the feature set. The second approach directly uses stopwords as independent feature sets for classification. In text classification, usually a large number of features are redundant (Joachims, 1998). If some features are removed and the classification accuracy does not change, it does not necessarily mean that these features are not relevant because similar features might exist in the feature sets, contributing to the classification. Hence the second approach is used to design this experiment. Two definitions of stop words are examined respectively. The common word list generated from the Brown Corpus

and the function word groups generated by the Brill part-of-speech tagger are used as independent feature sets for classification.

3.3 Experiment 3: stemming

This experiment evaluates the effect of stemming on classification performance at both macro and micro levels. At macro level, it examines whether the overall classification accuracies change significantly after stemming. At micro level it compares the contribution of individual features before and after stemming toward classification. For example, the features “woman” and “women” will be merged as one feature “woman” after stemming. If “woman” and ‘women” are relevant to the classes (e.g. the eroticism in Dickinson’s poem) in a similar way, this word stemming and merging event should not negatively affect the classification result. Otherwise, if one word indicates “erotic” and the other one indicates “not-erotic”, the conflation would neutralize two discriminative features and result in performance decrease. The idea of stemming and merging word features is similar to word clustering. All words with the same stem are gathered into one cluster. To group words into clusters, Baker and McCallum (1998) developed the averaged Kullback-Leibler-Divergence (KLD) to measure the similarity between words with regard to their contributions to classification. The smaller the KLD values (minimum=0), the more similar the words are. Similar words are then grouped into the same clusters. In this study KLD is used in a different way. The KLD between original features and their conflated forms are computed and sorted in decreasing order. Good conflations with KLD values close to zero are located at the bottom of the list, and bad conflations with high KLD values are at the top.

3.4 Experiment 4: statistical feature selection

The effectiveness of feature selection is measured from two perspectives. The classification accuracy measures the relevance of the selected features. The feature reduction rate measures the compactness of the selected feature subset. Feature reduction rate describes the proportion of features removed from the original feature set. The reduced feature set has to cover all documents, which means no empty document vectors should be generated after feature reduction (Yang and Pedersen, 1997). For each classifier the features are sorted in decreasing order by their absolute weights in the linear decision function. The top-ranked (heaviest) 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80% and 90% features are selected as the reduced feature sets to build new document vectors. The cross validation accuracies before and after the feature selection are compared to see if there are significant changes. The feature reduction rates are compared across classifiers. This experiment can start with either the stemmed feature sets or the original feature sets. To examine the potential interaction between stemming and statistical feature selection, the above experiments are repeated on both original and stemmed feature sets.

3.5 Experiment 5: learning curve and confidence curve

A learning curve describes a classifier's performance growth with increasing number of training examples. The turning point where the curve becomes flat indicates the minimum number of training examples needed for stable prediction accuracy. In the learning curve experiment, 10% examples will be reserved as test examples, and the rest 90% are used for training. The training set size increases from 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, to 90%. At each size the classification algorithm runs 50 times. Each time the specified percent of data is randomly selected from the 90% training set. The 50 classification accuracies are then averaged at each different training set size. At the beginning, the whole data set is split into ten folds. The above

experiment is repeated on each fold. The averaged classification accuracies will be used to draw the learning curve.

A linear classifier outputs a prediction value for each test example. This value indicates the distance between the test example and the decision hyperplane. The farther the data point is from the decision hyperplane, the more confident is the prediction. In this sense, the distance is a kind of confidence index of the prediction. The confidence curve experiment compares the confidence of each classifier's predictions on the same test data. The data set is randomly split to 60% training set and 40% testing set. Each classifier's predictions are sorted in decreasing order. The confidence curve plots the classifier's prediction accuracies in the top 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90% and 100% predictions. Slowly decreasing confidence curve means the classifier is able to maintain high confidence for most of its predictions.

4 The Dickinson erotic poem classification

The Dickinson data set contains 269 poems, among which 102 poems were labeled as erotic (positive), and 167 not-erotic (negative). The original vocabulary of the Dickinson collection consists of 3984 unique words. 1253 words remain after excluding the words which occur only once.

4.1 The text representation model selection

Table 2 lists the classification accuracies of SVM and naïve Bayes variations. "Svm-ntf" is the best representation for SVM. It is significantly better than "svm-bool", but its differences with "svm-tf" and "svm-tfidf" are not significant (see Table 3). For naïve Bayes variations, "nb-tf" is better than "nb-bool" and the majority baseline, but the differences are not significant. "Svm-ntf" and "nb-tf" will be used in the following experiments.

Table 2: choosing text representations (Dickinson)

text representation model	10-fold CV accuracy (%)
svm-bool	66.23
svm-tf	68.40
svm-ntf	**71.42**
svm-tfidf	67.31
nb-bool	65.46
nb-tf	**68.45**
majority vote	62.08

Table 3: paired *t*-tests for choosing text representations (Dickinson)

pairs	*t*	*p*-value
svm-ntf vs.svm-bool	3.509	0.007*
svm-ntf vs. svm-tf	1.355	0.208
svm-ntf vs. svm-tfidf	1.946	0.083
svm-ntf vs. majority	3.911	0.004*
nb-tf vs. nb-bool	1.441	0.183
nb-tf vs. majority	1.280	0.232
svm-ntf vs. nb-tf	0.513	0.620

4.2 Stopword features

The Brown stopword list consists of 425 most common words in the Brown corpus. 306 of them are found in Dickinson poems, but most of them are not longer "common". Table 4 compares the classification accuracies with different stopword feature sets. The 306 Brown stopword features work as well as the total 911 word features for the eroticism classification. The pronoun group has only 29 words, but the classifiers with pronoun features achieve the level of accuracy close to those with all 911 features. "You" and "I" are the best individual predictors for erotic poems. In summary stopwords are highly discriminative features for Dickinson erotic poem classification.

Table 4: the performance of stopword features (Dickinson)

feature set	size	svm-ntf (%)	nb-tf (%)
all	1253	71.4	68.5
Brown stopwords	306	69.6	69.9
pronoun	29	68.4	66.6
modal	14	52.1	47.6
prep-conj	67	56.2	57.3
determiner	19	59.8	60.3

4.3 Stemming

Table 5 lists the classification accuracies before and after stemming. At the macro level, the feature set is reduced by 13%, but there is no significant accuracy change before and after stemming. At the micro level, table 6 lists a few conflation events with largest and smallest KLD values. Some events are good, such as merging “silently” with “silent”. Some conflations are bad (with large KLD values), such as merging “hearts” with “heart”, “women” with “woman” and “thinking” with “think”. For some nouns, the singular forms are more relevant to erotic poems while the plural forms are more relevant to non-erotic poems. A possible explanation is that singular words like “woman” and “heart” are more self-portraying than their plural forms “women” and “hearts”.

A usual pre-processing step in text classification is to convert all words into lower cases. Dickinson is known for her unconventional capitalization. Many words, especially nouns, were capitalized no matter where they occurred. A Dickinson scholar explained it as an old-fashioned emphasis borrowed from German. This study examines the case merge as a special kind of word conflation. At the macro level no significant classification accuracy change is observed after the case merge. At the micro level there exist both good and bad case merges. For some words, capitalization does not change their relevance to eroticism, for example “Dream” vs. “dream”, “Place” vs. “place”, and “Road” vs. “road”. For other words, the capitalized forms bear different meanings, for example “Joy” vs. “joy”, “Royal” vs. “royal”, “Red” vs. “red”, and “Love” vs.

"love". In these cases, Dickinson used the capitalized forms to describe general concepts in abstract thinking in non-erotic poems, while she used the lower case forms to describe personal life scenarios in erotic poems.

For both case merging and stemming experiments, the overall classification accuracies do not change significantly. But it does not necessarily mean that all of these conflations do not matter. In fact, both good and bad conflations occur simultaneously, although their effects are neutralized overall.

Table 5: the effect of stemming (Dickinson)

stemming	feature set	"svm-ntf" accuracy (%)	"nb-tf" accuracy (%)
before case merging	1253	71.4	68.5
before stemming	1049	70.7	69.9
partial stemming	959	69.9	69.2
full stemming	911	70.7	69.2

Table 6: KLD rankings of stemming/merging events (Dickinson)

words before merge	words after merge	KLD
hearts	heart	1.029
heart	heart	0.006
thinking	think	0.785
thought	think	0.259
think	think	0.096
woman	woman	0.229
women	woman	0.060
silently	silent	0
silent	silent	0

4.4 Statistical feature selection

Table 7 shows the naïve Bayes feature selection results. For the stemmed feature set, the classification accuracy increases from 69.2% to 81.0% after self feature selection. The paired t-test result shows that the accuracy difference is significant ($t = 6.449$, $p < 0.001$). However naïve

Bayes self feature selection can only reduce the feature set up to 40% without generating empty documents. For the not-stemmed feature set, feature selection improves the accuracy even more (from 68.5% to 82.5%). However, there is no significant difference between the feature reduction results with or without stemming. Therefore stemming does not significantly affect the naïve Bayes feature selection in this task ($t = 0.675$, $p = 0.517$).

Table 7: naïve Bayes self feature selection (Dickinson)

	with stemming		without stemming	
percent	features	accuracy (%)	features	accuracy (%)
100%	911	69.2	1253	68.5
90%	820	75.0	1128	72.5
80%	729	76.2	1003	78.8
70%	638	78.4	877	82.5
60%	547	81.0	752	-
50%	456	-	627	-
40%	364	-	501	-
30%	273	-	376	-
20%	182	-	251	-
10%	91	-	125	-

During feature reduction the SVM classification accuracies increase with some fluctuations (table 8). The accuracy changes from 70.7% to 76.2% for stemmed features, and from 71.4% to 77.0% for not-stemmed features. The improvements are significant ($t = 3.143$, $p = 0.012$), although not as much as the improvements for naïve Bayes. However, SVM yields high feature reduction rate. Actually SVM with the top 10% features performs better than SVM with the entire feature set.

Table 8: SVM self feature selection (Dickinson)

	with stemming		without stemming	
percent	features	accuracy (%)	features	accuracy (%)
100%	911	70.7	1253	71.4
90%	820	67.7	1128	66.9
80%	729	71.0	1003	67.3
70%	638	71.4	877	69.1
60%	547	71.8	752	72.1
50%	456	72.5	627	73.3
40%	364	73.2	501	73.6
30%	273	74.7	376	73.2
20%	182	74.0	251	74.0
10%	91	76.2	125	77.0

To compare the two feature ranking and selection methods in more detail, Figure 1 plots the features with their SVM weights on X axis and naïve Bayes weights on Y axis. The two methods generally agree upon which features are "erotic" or not, because most features fall into the first and third quadrants in Figure 1. However there are only 27 shared features in both top 100 feature lists. Apparently the two methods prefer different kinds of features as the top ones.

Figure 1: SVM and naïve Bayes feature ranking agreement (Dickinson)

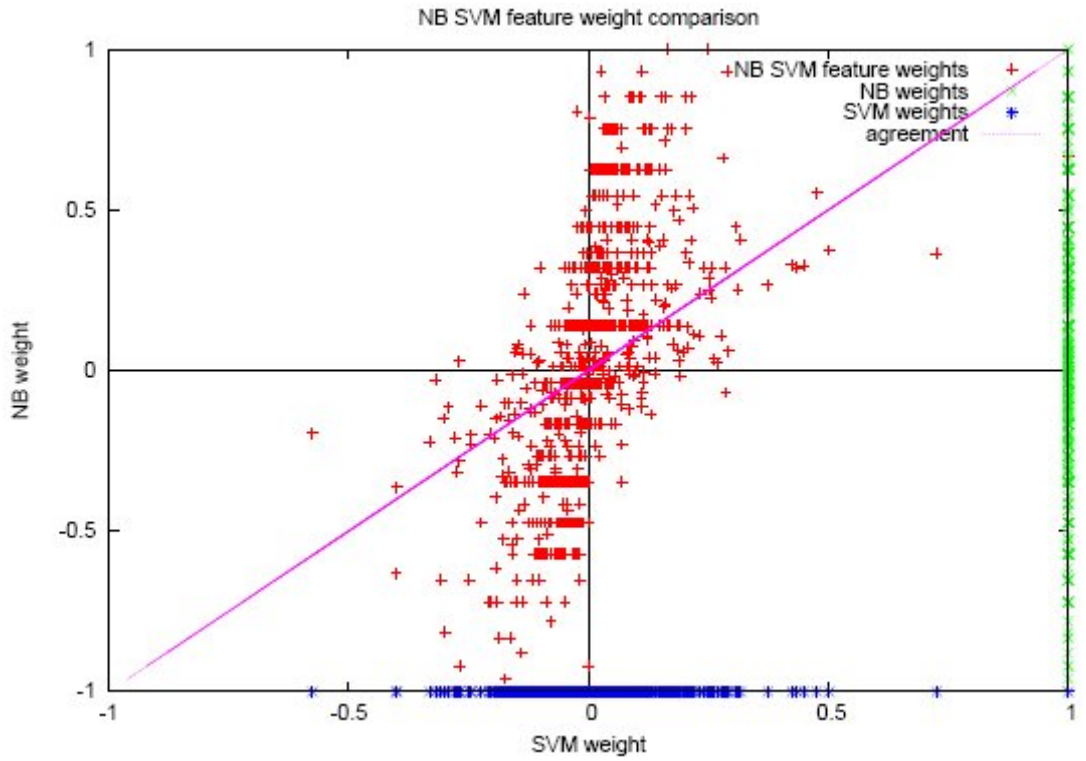
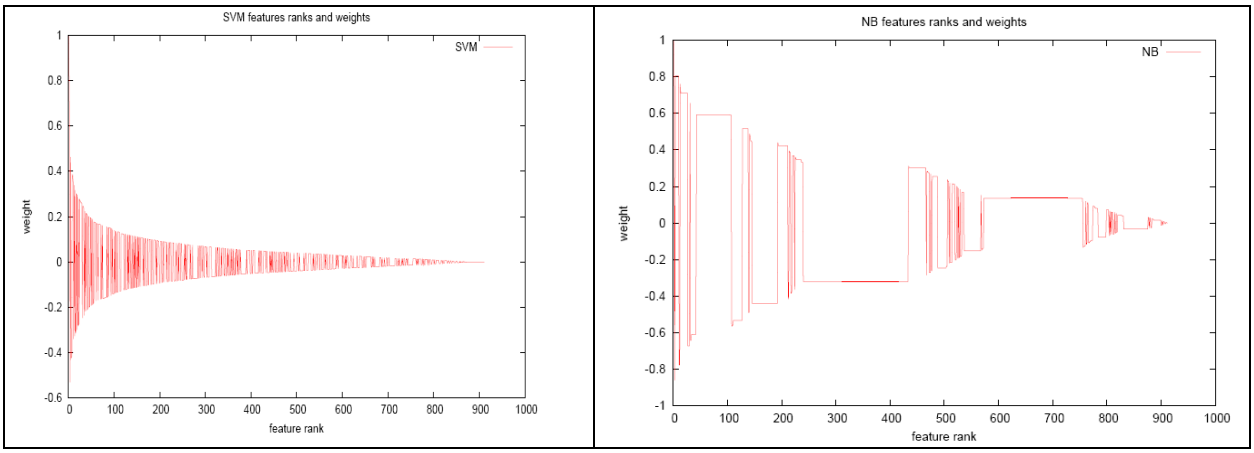


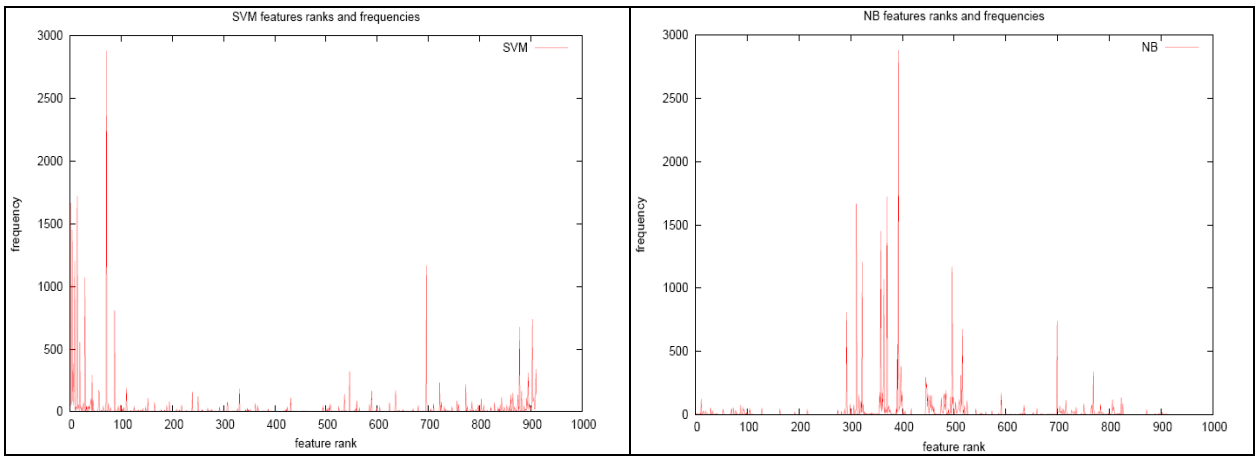
Figure 2 plots the relation between the feature ranks and their weights. The feature weights are normalized as proportional to the top feature weight. The left figure shows that the SVM feature weights decrease quickly and smoothly from top rank to bottom rank. The small number of top features have strong influence on the classification decisions. The remaining features are not important due to the small feature weights. This explains the SVM high reduction rate from one aspect. The right figure shows that there are large numbers of naïve Bayes features with same weights. The feature values decrease slowly. Most features ranked in the middle still have heavy feature weights. Because of the large number of heavy weight features, naïve Bayes can not achieve high feature reduction rate.

Figure 2: SVM and naïve Bayes feature ranks and weights



Both “svm-ntf” and “nb-tf” use word frequencies as feature values, normalized or not. Figure 3 plots the relations between feature ranks and their frequencies for both classifiers. The left figure (SVM) shows that high frequency words accumulate at the top SVM feature ranks. Therefore a small feature subset is enough to cover the whole collection without generating empty documents. In contrast, the right figure (naïve Bayes) shows that low frequency words dominate the top naïve Bayes feature ranks. Most high frequent words rank in the middle, so a larger feature subset is needed to avoid generating empty documents. In consequence naïve Bayes cannot achieve high reduction rate.

Figure 3: SVM and naïve Bayes feature ranks and frequencies



The above relations between feature ranks and frequencies can be explained by the feature ranking functions of the two methods. "Nb-tf" uses the log probability ratio $\log\frac{p(w\mid pos)}{p(w\mid neg)}$ to measure feature weights (Mladenic and Grobelnik, 1999). For example, if words A and B occur in exactly the same documents, and B's occurrences in each document is always twice as A's occurrences, "nb-tf" would assign the same weights for A and B. "Svm-ntf" uses the function $w_j = \sum_{i=1}^{l} \alpha_i y_i x_{ij}$ to measure feature weights. In this function x_{ij} is the normalized frequency for word w_j in the Support Vector i; α_i is the Support Vector's non-negative coefficient and y_i is its class label (1 or -1). Therefore in the above example "svm-ntf" would assign word B with doubled weight of word A. In Dickinson's poems most words are not frequent, but their frequency ratio in the two classes could be high. Naïve Bayes assigns heavy weights to these words while SVM devalues them.

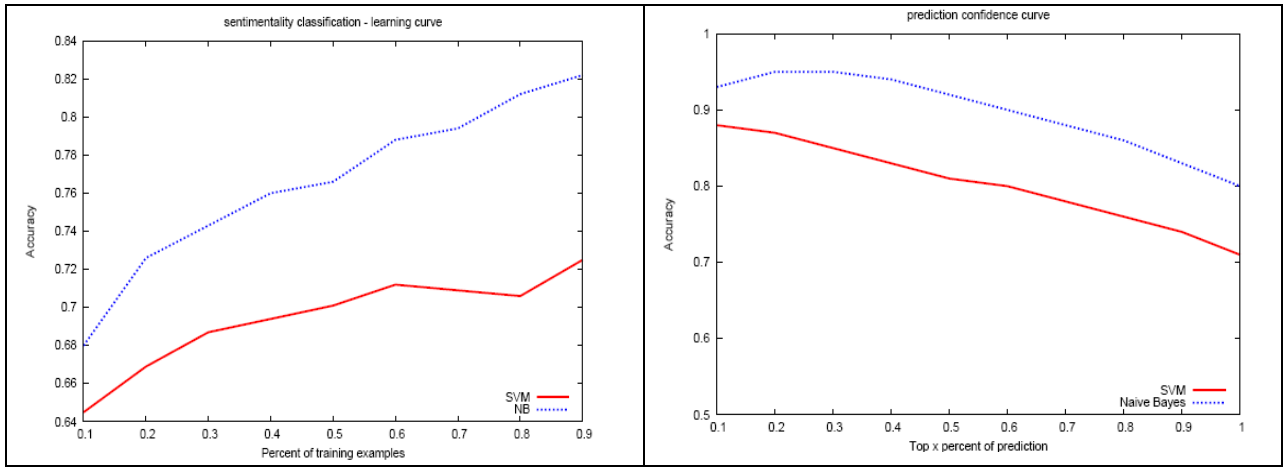
The difference between the two feature selection methods is also related to the feature informativeness. Naïve Bayes selects unique words in each category as top features, which are usually in low frequencies. The scholars are surprised at the first sight of these words (e.g. "write", "mine", and "Vinnie"), but they managed to make sense of them later. A possible explanation is that the occurrences of these words are very limited, therefore it is not hard for the scholars to associate the words with their context and infer their relevance to eroticism. In contrast, SVM chooses the high frequency words as top features, such as the pronouns "you", "I", "my", "me", "your" and "her". It is within the scholars' prior knowledge that pronouns are necessary to construct personal conversations. Although these features do not surprise the scholars, they exhibit the common characteristics of Dickinson's erotic poems. This result is

consistent with the stopword experiment result in that pronouns are highly discriminant features for eroticism classification.

4.5 Learning curve and confidence curve

Both classifiers' learning curves and confidence curves are plotted in Figure 4. The left figure shows that naïve Bayes learns faster than SVM in this task. However both learning curves do not level off, which indicates that the classifiers need more training data to reach stable performance. For both algorithms the classification accuracies decrease at similar speed with the decrease of the confidence (see the right figure in Figure 4).

Figure 4: learning curves and confidence curves (Dickinson)



5 The sentiment classification of early American novel chapters

For sentimentalism analysis, the literary scholars at University of Virginia are most interested in content words. But they suggested that proper nouns be excluded from the feature set because most of them are character names. A sentimentalism classifier aims to learn the sentimental language rather than the character designs in particular novels. The sentimentalism collection

consists of 184 chapters, among which 95 chapters have high level of sentimentality and 89 chapters have low sentimental level. The original vocabulary contains 19585 word tokens. Because the average chapter length is much longer than that of the Dickinson poems, the minimum word frequency is arbitrarily set to 5. Again, the Brill tagger is used to extract content words - nouns (without proper nouns), verbs, adjectives and adverbs. Eventually the feature set consists of 5704 words.

5.1 Text representation model selection

Boolean feature representations are the best for sentimentalism classification (table 9). Both "svm-bool" and "nb-bool" are significantly better than the majority baseline, but their difference with other SVM and naïve Bayes variations are not significant (table 10). "Svm-bool" and "nb-bool" are then used in the following experiments.

Table 9: text representation model selection (Sentimentalism)

text representation model	10-fold CV accuracy (%)
svm-bool	**66.4**
svm-tf	62.0
svm-ntf	63.4
svm-tfidf	60.5
nb-bool	**64.9**
nb-tf	64.1
majority vote	51.6

Table 10: paired *t*-tests for text representation model selection (Sentimentalism)

pairs	*t*	*p*-value
svm-bool vs.svm-tf	1.121	0.291
svm-bool vs. svm-ntf	0.675	0.517
svm-bool vs. svm-tfidf	2.500	0.034
svm-bool vs. majority	3.630	0.005*
nb-bool vs. nb-tf	0.352	0.733
nb-bool vs. majority	3.318	0.009*
svm-bool vs. nb-bool	0.647	0.534

5.2 Stopword features

Table 11 lists the classification accuracies with different stopword groups as feature sets. Neither the Brown stopwords nor the function word groups achieved accuracies significantly higher than the trivial majority baseline for both algorithms. This result confirms the scholars' heuristic that content words are more relevant in this case.

Table 11: performance of stopword features (Sentimentalism)

feature set	size	"svm-bool" accuracy (%)	"nb-bool" accuracy (%)
all	5704	66.4	64.9
Brown stopwords	404	56.0	57.2
pronoun	27	54.5	58.1
modal	15	54.9	57.1
prep-conj	88	52.1	55.4
determiner	22	54.5	55.7

5.3 Stemming

For sentimentalism classification the accuracies of both classifiers do not change significantly after stemming. Stemming reduces the feature set size by 36% (table 12). Some conflations are good, such as merging "difficulties" with "difficulty" and "wheels" with "wheel". Other conflations are bad, such as merging "wildness" with "wild" and "pitying" with "pity" (table 13). "Wildness" is exclusively used in highly sentimental chapters while "wild" occurs in both high sentimental and low sentimental chapters with similar frequencies. See below for a few examples of using the word "wildness" in sentimental chapter.

"There was a piercing wildness in the cry…" (Uncle Tom's Cabin, chapter 27)

"Her soft brown eyes had a flash of despairing wildness in them…" (The Minister's Wooing, chapter 23)

Table 12: the effect of stemming (Sentimentalism)

stemming	feature set	"svm-bool" accuracy (%)	nb-bool accuracy (%)
before	5704	66.4	64.9
after	3669	66.9	64.3

Table 13: KLD rankings of stemming/merging events (Sentimentalism)

words before merge	words after merge	KLD
wildness	wild	**0.583**
wild	wild	0.003
pitying	piti	**0.515**
pitiful	piti	0.041
pitied	piti	0.005
pity	piti	0.001
difficulties	difficulti	0
difficulty	difficulti	0
wheels	wheel	0
wheel	wheel	0

5.4 Statistical feature selection

For stemmed features, the naïve Bayes classification accuracy increases from 70.2% to 88.0% after self feature selection (table 14). The performance difference is significant ($t = 7.796$, $p < 0.001$). The feature reduction rate is as high as 80%. Feature reduction without stemming produces even more accuracy improvement (from 65.4% to 92.4%) and higher reduction rate (90%). However stemming does not significantly affect the naïve Bayes feature reduction results.

Table 14: naïve Bayes self feature selection (Sentimentalism)

	with stemming		without stemming	
percent	features	accuracy (%)	features	accuracy (%)
100%	3669	70.2	5704	65.4
90%	3302	68.8	5134	67.4
80%	2935	71.5	4563	73.6
70%	2568	76.5	3993	77.7
60%	2201	80.6	3422	80.4
50%	1835	81.7	2852	83.8
40%	1468	84.1	2282	86.3
30%	1101	88.7	1711	89.2
20%	734	88.0	1141	91.8
10%	367	-	570	92.4

For stemmed features, the SVM classification accuracy fluctuates with the increase of feature reduction rate. There is no significant accuracy improvement after feature reduction. However, without stemming the SVM classification accuracy steadily improves with the increase of feature reduction rate. With top 10% not-stemmed features the SVM classifier achieves 94.1% accuracy.

Why does stemming affect SVM feature selection in sentimentalism classification but not in eroticism classification? Recall that stemming reduces features by 13% for SVM eroticism classification, but the reduction rate is 36% for SVM sentimentalism classification. The stemming process might have conflated and neutralized a large number of discriminative features, and therefore resulted in the loss of candidate discriminative features for future statistical feature selection.

Table 14: SVM self feature selection (Sentimentalism)

	with stemming		without stemming	
percent	features	accuracy (%)	features	accuracy (%)
100%	3669	69.5	5704	67.0
90%	3302	66.3	5134	67.1
80%	2935	65.7	4563	72.8
70%	2568	66.3	3993	76.7
60%	2201	65.2	3422	82.4
50%	1835	62.0	2852	85.9
40%	1468	62.0	2282	88.8
30%	1101	61.4	1711	89.7
20%	734	63.6	1141	91.2
10%	367	66.3	570	94.1

Both naïve Bayes and SVM algorithms reach comparable classification accuracies with their own top 10% (570) features. However, there are only 1/3 shared features in the two top 10% feature lists. Figure 5 plots the relation between the feature weights measured by the two feature selection methods. In the figure the points disperse away from the diagonal toward both ends of the axes. In other words, the two weighting measures agree basically upon the light-weighted features, but they disagree upon the features with heaviest weights.

Figure 5: naïve Bayes and SVM feature ranking agreement

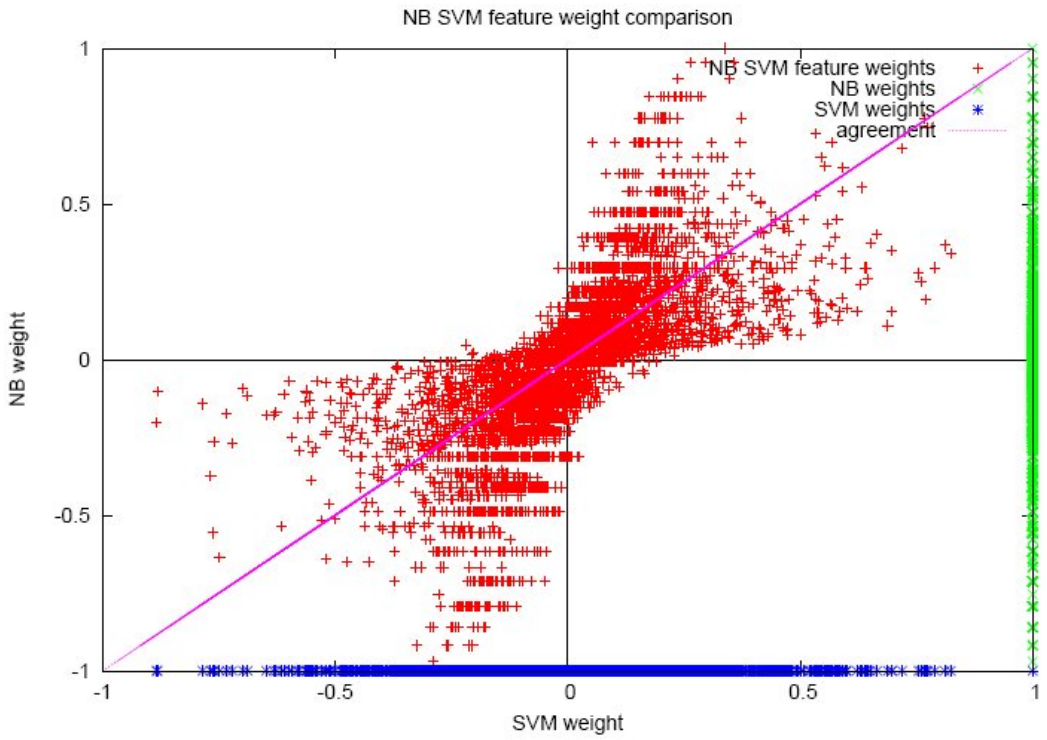


Figure 6 plots the feature ranks and their weights for both classifiers. Similar to Figure 2 in the Dickinson erotic poem classification, the left figure in Figure 6 shows that the SVM feature weights decrease quickly and smoothly from top ranks to bottom ranks. This time the relations between feature ranks and weights for naïve Bayes (the right figure) and SVM (the left figure) are similar, except that SVM feature weights decrease faster. This is consistent with the results that the two methods have similar feature reduction rate for sentimentalism classification.

Figure 7 plots feature ranks and their frequencies for both classifiers. Similar to Figure 3 in the Dickinson erotic poem classification, the top naïve Bayes features (in the right figure) are all low frequency words, while the frequencies of the top SVM features (in the left figure) are more distributed across the range. This time both algorithms use Boolean feature values, hence the frequencies as shown in Figure 7 are the words' document frequencies[8].

[8] Document frequency is the number of documents in which the word occurs.

Figure 6: SVM and naïve Bayes feature ranks and weights

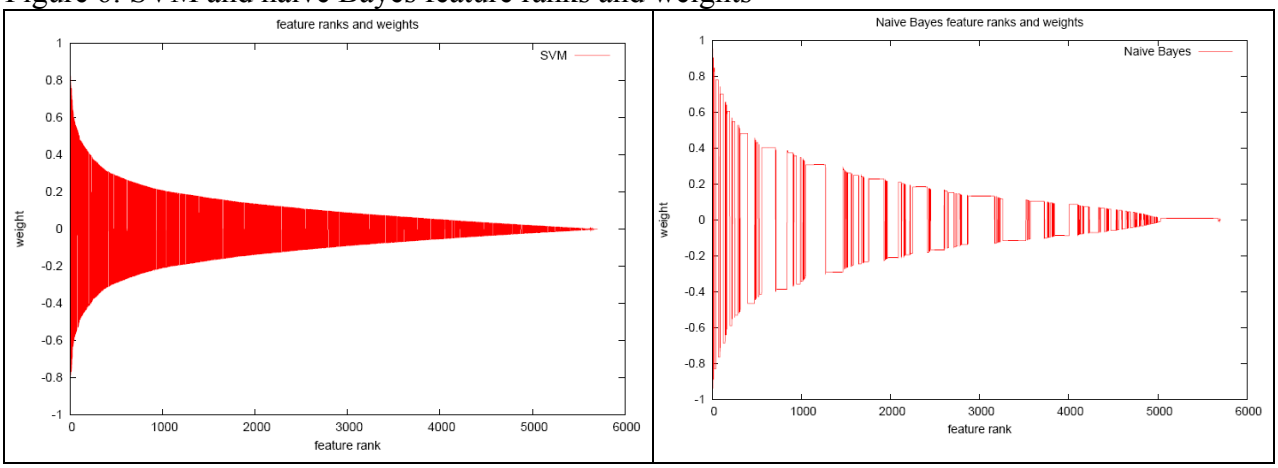
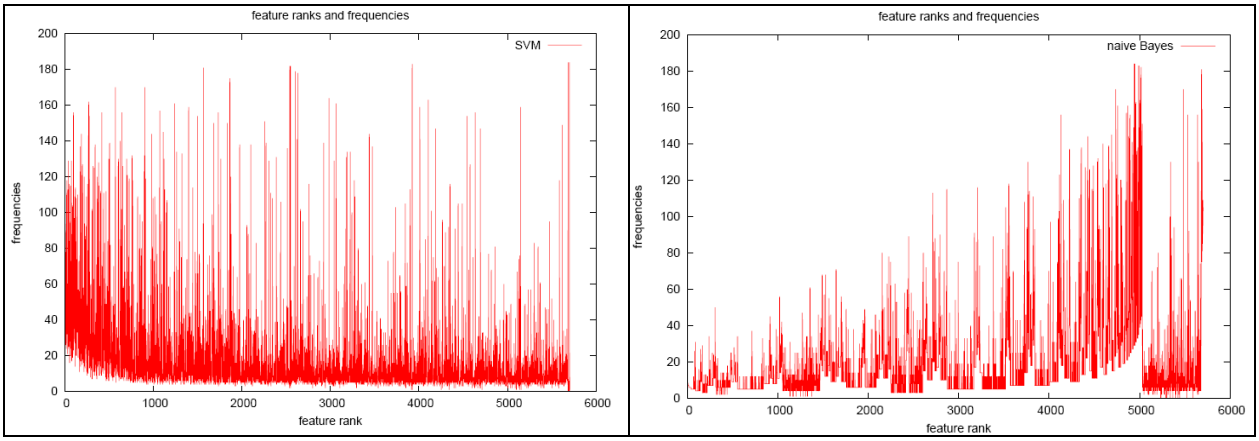


Figure 7: SVM and naïve Bayes feature ranks and frequencies

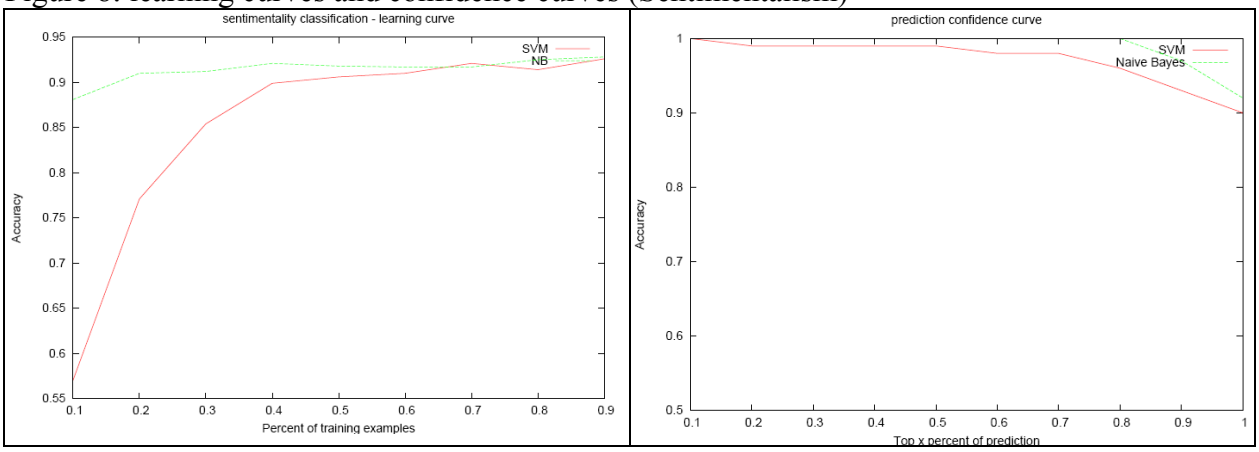


Both SVM and naïve classifiers include some sentimental words in their top feature lists, such as “die”, “sorrow”, “beloved” and “agony”. However, many features in both lists do not seem “sentimental” to the literary scholars, such as the words “to-morrow”, “paternal” and “payment”. The novel chapters are generally longer than the Dickinson poems. It is not surprising to find low sentimental text snippets mixed with highly sentimental ones. In consequence, some words which are not sentimental are also measured as sentimental because of their sentimental context.

5.5 Learning curves and confidence curves

Figure 8 shows the learning curves and confidence curves of both classifiers. The left figure shows that the SVM learning curve starts with low accuracy but improves fast with the increase of training example numbers. The learning curve levels off when the number of training examples exceeds 40%. The naïve Bayes classifier starts with 88% accuracy with only 10% training examples, leaving less room for improvement with the increase of training examples. The right figure in Figure 8 shows that the confidence level of naïve Bayes predictions decreases more slowly than that of SVM. Overall naïve Bayes has better learning curve and confidence curve in sentimentalism classification.

Figure 8: learning curves and confidence curves (Sentimentalism)



6 Conclusion

The evaluation results in this study demonstrate that SVMs are not all winners in literary text classification tasks. Both SVM and naïve Bayes classifiers achieved high accuracies in sentimental chapter classification, but the naïve Bayes classifier outperformed SVM in erotic poem classification. Self feature selection helped both algorithms improve their performance in

both tasks. However, the two algorithms selected relevant features in different frequency ranges, and therefore captured different characteristics of the target classes. The naïve Bayes classifiers prefer words unique to the classes, which are often not frequent. In contrast, SVMs prefer high frequent and discriminant words, which are scarce in some genres such as poems. For the purpose of feature relevance analysis the two methods should be used as complementary to each other rather than one over the other.

High classification accuracy is not necessarily associated with good generalizability. Despite the high accuracy in erotic poem classification, the naïve Bayes classifier is not a good example-based eroticism retrieval tool. Its learning curve does not level off with the increase of training examples, which indicates limited generalizability. In other words, this classifier is only good for summarizing the characteristics of the training data. Both algorithms yield high potential for example-based sentimentalism retrieval because of their fast increasing learning curves and strong confidences in predictions.

The evaluation results in this study also suggest that arbitrary feature reduction steps such as stemming and stopword removal should be taken very carefully. Stopwords were highly discriminative features for erotic poem classification. In sentimental chapter classification stemming undermined subsequent feature selection by aggressively conflating and neutralizing discriminative features.

Overall, while the use of text classification methods is very promising in literary text analysis applications, empirical experience on classification methods obtained from other domains should be carefully examined before applying to the new domain.

References

Argamon, S. and Olsen, M. (2006). Toward meaningful computing. *Communications of ACM*, 49(4), 33-35

Argamon, S., Saric, M., and Stein, S. (2003). Learning algorithms and features for multiple authorship discrimination. *Proceedings of IJCAI'03 Workshop on Computational Approaches to Style Anslysis and Synthesis*

Baeza-Yates, R. and Ribeiro-Neto, B. (1999). *Modern Information Retrieval*. Addison Wesley

Baker, L. D. and McCallum, A. K. (1998). Distributional clustering of words for text classification. *Proceedings of the 21th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'98)*, 96–103

Biber, D. (1988). *Variations across Speech and Writing*. Cambridge University Press

Biber, D. (1995). *Dimensions of Register Variation*. Cambridge University Press

Bland, J. M. and Altman, D. G. (1995). Multiple significance tests: the bonferroni method. *British Medical Journal*, 310, pp. 170

Brill, E. (1995). Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4), 543–566

Cohen, W. (1995). Learning to classify English text with ILP methods. In L. D. Raedt, (ed), *Advances in Inductive Logic Programming*. Amsterdam, NetherLand: IOS Press

Craig, H. (1999). Authorial attribution and computational stylistics: if you can tell authors apart, have you learned anything about them? *Literary and Linguistic Computing*, 14(1), 103–113

Dumais, S., Platt, J., Heckerman, D., and M. Sahami. (1998). Inductive learning algorithms and representations for text categorization. *Proceedings of the 7th International Conference on Information and Knowledge Management (ICKM'98)*, 148-155

Domingos, P. and Pazzani, M. (1997). On the optimality of the simple Bayesian classifier under zero-one loss. *Machine Learning*, 29:103–130

Forman, G. (2003). An extensive empirical study of feature selection metrics for text categorization. *Journal of Machine Learning Research*, 3:1289–1305

Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using Support Vector Machines. *Machine Learning*, 46(1-3), 389–422

Heffernan, J. (2004). Museum of words: The poetics of ekphrasis from Homer to Ashbery. University Of Chicago Press

Holmes, D. (1994). Authorship attribution. *Computers and the Humanities*, 28(2), 87-106

Holmes, D. (1998). The evolution of stylometry in humanities scholarship. *Literary and Linguistic Computing*, 13(3), 111-117

Horton, R., Olsen, M., Roe, G., and Voyer, R. (2007). Mining eighteenth century ontologies: machine learning and knowledge classification in the Encyclopédie. Digital Humanities 2007, Champaign, Illinois

Horton, T., Taylor, C., Yu, B., and Xiang, X. (2006). "Quite right, dear and interesting": seeking the sentimental in nineteenth century American fiction. Digital Humanities 2006, Paris, France

Joachims, T. (1998). Text categorization with Support Vector Machines: Learning with many relevant features. *Lecture Notes in Computer Science (ECML'98)*, Issue 1398, 137-142

Juola, P. and Baayen, H. (2005). A controlled-corpus experiment in authorship identification by cross-entropy. *Literary and Linguistic Computing*, 20(Suppl 1), 59-67

Koppel, M., Argamon, S., and Shimoni A. R. (2002). Automatically categorizing written texts by author gender. *Literary and Linguistic Computing*, 17, 401-412

Lewis, D. D. (1992). An evaluation of phrasal and clustered representations on a text categorization task. *Proceedings of the 15th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'92)*, 37–50

Lewis, D. D. (1998). Naïve Bayes at forty: The independence assumption in information retrieval. *Proceedings of the 10th European Conference on Machine Learning (ECML'98)*

McCallum, A. and Nigam, K. (1998). A comparison of event models for naïve Bayes text classification. In *AAAI 98 Workshop on Learning for Text Categorization*

Mitchell, T. M. (1997). Machine Learning. McGraw-Hill.

Mladenic, D., Brank, J., Grobelnik, M, and Milic-Frayling, N. (2004). Feature selection using linear classifier weights: Interaction with classification models. *Proceedings of the 27nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'04)*, 234–241

Mladenic, D. and Grobelnik, M. (1999). Feature selection for unbalanced class distribution and naïve Bayes. *Proceedings of the Sixteenth International Conference on Machine Learning (ICML'99)*, 258–267

Mosteller, F. and Wallace, D. (1964). Inference and disputed authorship: the federalist papers. Massachusetts: Addison-Wesley

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumps up?: Sentiment classification using machine learning techniques. *Proceedings of the ACL-02 Conference on Empirical Methods in Natural Language Processing (EMNLP'02)*, 79-86

Plaisant, C., Rose, J., Yu, B., Auvil, L., Kirschenbaum, M.G., Smith, M. N., Clement, T., and Lord, G. (2006) Exploring erotics in Emily Dickinson's correspondence with text mining and visual interfaces. *Proceedings of the 6th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL'06)*, 141-150

Porter, M. (1980). An algorithm for suffix stripping. *Program*, 14(3), 130–137

Ramsay, S. (2004). In praise of pattern. In "The Face of Text" - 3rd Conference of the Canadian Symposium on Text Analysis (CaSTA)

Ramsay, S. (2008). Algorithmic criticism. In R. Siemens and S. Schreibman (Eds.), *A Companion to Digital Literary Studies*. Oxford, UK: Wiley-Blackwell, pp. 477-491

Riloff, E. (1995). Little words can make a big difference for text classification. *Proceedings of the 18th annual international ACM SIGIR conference on research and development in information retrieval (SIGIR'95)*, 130–136

Scott, S. and Matwin, S. (1999). Feature engineering for text classification. *Proceedings of the 16th International Conference on Machine Learning*, 379-388

Sebastiani, F. (2002) Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1), 1–47

Unsworth, J. (2000). Scholarly primitives: what methods do humanities researchers have in common, and how might our tools reflect this? *Symposium on Humanities Computing: formal methods, experimental practice*, King's College, London

Vapnik, V. N. (1982). *Estimating of Dependencies Based on Empirical Data.* New York: Springer-Verlag.

Vapnik, V. N. (1999). *The Nature of Statistical Learning Theory*. 2nd Edition. Springer

Yang, Y. and Liu, X. (1999). A re-evaluation of text categorization methods. *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'99)*, 42–49

Yang, Y. and Pedersen, J. (1997). A comparative study on feature selection in text categorization. *Proceedings of the 14th International Conference on Machine Learning (ICML'97)*, 412-420

Yu, B. and Unsworth, J. (2006). Toward discovering potential data mining applications in literary criticism. Digital Humanities 2006, Paris, France