MACHINE LEARNING METHOD FOR AUTHORSHIP ATTRIBUTION

By

Xianfeng Hu

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

Applied Mathematics—Doctor of Philosophy

2015

ABSTRACT

MACHINE LEARNING METHOD FOR AUTHORSHIP ATTRIBUTION

By

Xianfeng Hu

Broadly speaking, the authorship identification or authorship attribution problem is to determine the authorship of a given sample such as text, painting and so on. Our main work is to develop an effective and mathe-sound approach for the analysis of authorship of doubted books.

Inspired by various authorship attribution problems in the history of literature and the application of machine learning in the study of literary stylometry, we develop a rigorous new method for the mathematical analysis of authorship by testing for a so-called chrono-divide in writing styles. Our method incorporates some of the latest advances in the study of authorship attribution, particularly techniques from support vector machines. By introducing the notion of relative frequency of word and phrases as feature ranking metrics our method proves to be highly effective and robust.

Applying our method to the classical Chinese novel *Dream of the Red Chamber* has led to convincing if not irrefutable evidence that the first 80 chapters and the last 40 chapters of the book were written by two different authors.

Also applying our method to the English novel *Micro*, we are able to confirm the existence of the chrono-divide and identify its location so that we can differentiate the contribution of Michael Crichton and Richard Preston, the authors of the novel.

We have also tested our method to the other three Great Classical Novels in Chinese. As expected no chrono-divides have been found in these novels. This provides further evidence of the robustness of our method.

We also proposed a new approach to authorship identification to solve the open class problem where the candidate group is nonexistent or very large, which is reliably scaled from a new method we have developed for the close class problem in which the candidates author pool is small. This is attained by using support vector machines and by analyzing the relative frequencies of common words in the function words dictionary and most frequently used words. This method scales very nicely to the open class problem through a novel *author randomization technique*, where an author in question is compared repeatedly to randomly selected authors. The author randomization technique proves to be highly robust and effective. Using our approaches we have found answers to three well known authorship controversies: (1) Did Robert Galbraith write *Cuckoo's Calling*? (2) Did Harper Lee write *To Kill a Mockingbird* or did her friend Truman Capote write it? (3) Did Bill Ayers write Obama's autobiography *Dreams From My Father*?

ACKNOWLEDGMENTS

First and foremost I want to thank my advisor Yang Wang, the smartest person I know. It has been an honor to be his Ph.D. student. I want to thank him for helping me to shape and guide the direction of the research field in particular. His support and insightful discussions about research made my pursuit of Ph.D. possible. I appreciate all his contribution of time, ideas and funding to make my Ph.D. productive and stimulating over the past five years. The enthusiasm he has for his research was contagious and motivational for me, even during the tough times in the Ph.D. pursuit. Beyond his scientific guidance, his advice on navigating an academic career has been invaluable.

I will forever be thankful to my advisor Min Wu in South China University of Technology. She was and remains my excellent example as a successful woman mathematician, mentor and teacher. She is the reason why I decided go to pursue a career in research. I am grateful for her support and encouragement to pursue Ph.D.

I thank Mark Iwen for sharing his GFFT code. It is he who leaded me to use c++ and showed tricks of debug. I really appreciate his patient and time during discussion. Thank him very much for helping me to prepare job application materials and suggestions during application. I also want to thank Qiang Wu for helping to start my first project and matlab. I thank him for his inspiring discussion and wonderful suggestions in life. I also want to thank Aditya Viswanathan for his wonderful suggestions of toolbox during discussion. I would like to thank Zhengfang Zhou, Andrew Christlieb and Yingda Cheng for serving on my defense committee.

I would like to thank my fellow math students for their friendship over the past five years and for sharing the wonderful life with me, Zixuan Wang and Yuqi Hong in particular. The

nights spent playing tennis, sharing dinner, or laughing over desert made the experience of graduate school more rewarding. I would also thank my friends Liping Chen and Liangmin Zhou for helping me during study and teach. My parents Junlin and Genshui, my brother Mubiao and sisters Nian, Xue and Jie, have been incredibly supportive throughout my student life. Thank them for giving me a warm family and happy childhood and I dedicate this dissertation to them.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Authorship Attribution

Did Francis Bacon or Christopher Marlowe write some of the plays that were attributed to William Shakespeare? Did Cao Xueqin write the last forty chapters of *Dreams of the Red Chamber*, one of the greatest masterpieces in the history of Chinese literature? Who wrote the *Federalist* papers? Did Bill Ayers write Obama's autobiography *Dreams From My Father*? There have been no shortages of authorship controversies throughout the history, and these are just a few of them. These questions have often generated fierce debates among historians and literary scholars. Historically connoisseurship, disciplinary knowledge and background history have been central in such debates. In recent years, however, the "unemotional approach" using quantitative analysis based on mathematics, statistics and computation has gained prominence in the study of authorship attribution.

The authorship attribution is to determine the authorship of works like books, visual arts and so on whose authors are unknown. There are two classes of problems in authorship attribution [21] in general. The most common one is the *close class*, where it is known that the sample of text is by one of the authors from a given set, usually a very small set with two to three authors. For example, *Federalist* papers mentioned earlier is a typical closed class problem, where the authors are confined to Alexander Hamilton, James Madison and John Jay. Far more challenging is the *open class* problem, where the sample text may come from a far larger set of authors or even no suspect author. An example of an open class problem is to determine the authorship of books written anonymously or under pseudonyms, e.g. *The Primary Color* or *Imperial Hubris: Why the West is Losing the War on Terror.* The study of authorship attribution has seen rapid growth in recent years. Still, there remains many challenges even for the closed class problem. Accuracy is one of the most prominent concerns. For the open class problem the challenge is far greater, and so far there have been few attempts and none of the known techniques can be reliably scaled to handle the open class problem.

We study both open class and closed class problems in authorship attribution. The first part of this thesis focus on an important authorship question in closed class problem, which we call chrono-divide, namely to determine whether a body of text is written by a single author. Such a study is valuable in a number of ways. For example, it is widely speculated among the Shakespearean scholars that some of the Shakespeare plays were in fact collaboratively written by Shakespeare and others, e.g. Middleton and Fletcher. It is of scholarly interest to not only confirm it but also to find out exactly which parts of the plays were written by playwrights other than Shakespeare. There were also historical controversies involving suspected fraud, as in the case of *Dream of the Red Chamber*, where a particular book or sequel attributed to certain well known author might in fact be perpetrated by someone else. A related practice was for a well known and prolific author to write only the first few chapters of a book and then pass it on to a ghost writer. There are clear benefits both ethically and scholarly to detect frauds and ghost writings. Modern days see an explosion of coauthored books and articles. It would be interesting to detect stylistic inconsistencies among parts of such books. As we shall see, mathematics can play a central role in the study of authorship, leading to the rapid growth of the field *stylometry.*

We will introduce our method on the detection of chrono-divide in writing styles with a given body of text in this part and use two case studies to show ways it can be done, and to illustrate the effectiveness and robustness of our methods by comparing the study of cases just by one author. The first case study is the classical Chinese novel *Dream of the Red Chamber*. The controversy surrounding the book was well known and intensely debated in the Chinese literary circle for over 250 years. The second case study is the book *Micro* written by Michael Crichton and Richard Preston. The one author cases will be China's other three classical novels, *Three Kingdoms*, *Journey to the West* and *The Water Margin*. The research for the first case study has been done in our work [39], and much of what we present here is reproduced from [39]. All materials in the second case study are new.

The second part of this thesis will be the study of open problem in authorship attribution, which is to detect the author of a body of text whose unknown true author is among a large set of candidate authors. In many cases there isn't even a suspect so the set of candidate authors is the entire database. To solve an open class identification problem, we first propose a new method for solving the close class problem where the number of candidate author of a text is small (e.g. ≤ 8). We then reliably scale it to solve the open class problem through the author randomization technique. As long as the true author is in our database we can reliably identify the author. Furthermore, if the true author is not in the database we can also reliably rule out any author in the database.

We will describe our techniques and the algorithm of our approach, and demonstrate the effectiveness through three case studies: (1) *Cuckoo's Calling* by Robert Galbraith, which turned out to be the pseudonym for J. K. Rowling; (2) *To Kill a Mockingbird* by Harper Lee in 1960, which some speculated to be written by her long time friend Truman Capote; (3) *Dreams From My Father* by President Obama, whose authorship is in doubt ever since Bill

Ayers "confessed" that he was the true author.

1.2 Related Works

The problem of style quantification and authorship attribution in the literature goes at least as far back as 1854 by the English mathematician Augustus De Morgan [11], who in a letter to a clergyman on the subject of Gospel authorship, suggested that the lengths of words might be used to differentiate authors. In 1897 the term *stylometry* was first coined by the historian of philosophy, Wincenty Lutaslowski, as a catch-all for a collection of statistical techniques applied to questions of authorship and evolution of style in the literary arts (see e.g. [29]). Today, literary stylometry is a well developed and highly interdisciplinary research area that draws extensively from a number of disciplines such as mathematics and statistics, literature and linguistics, computer science, information theory and others. It is a central area of research in statistical learning (see e.g. [18]).

The main assumption of authorship attribution is that every person has his or her own style of speaking, writing and painting. These styles can be quantified using some so called "authorial fingerprints" [21] or features. Over the history of authorship attribution study many different kinds of features have been proposed for authorship identification. The goal has always been to find certain feature or features that will differentiate one author from another. Yule [41] suggested the length of sentence as a feature of writing style and applied it to two cases of disputed authorships. A popular classic technique for stylometric analysis of authorship involves comparing frequencies of the so-called *function words*, a class of words that in general have little content meaning, but instead serve to express grammatical relationships with other words within a sentence. The different percentages of nouns, verbs, adjectives, adverbs and other parts of speech can also be useful characters of writing style, see [1, 5, 6]. Ellegard [14] determined the authorship of the Junius Letters using function words frequencies and synonym pairs (e.g. ratio of "big" vs "large"). Mosteller and Wallace [28] focused on the distribution of 30 function words from the various Federalist papers and analyzed the authorship of them. Popular earlier features also include average length of words, cumulative sums (Qsums) [3, 4, 15, 27], n-grams. In [22] the authors presented a method for computer-assisted authorship attribution based on character-level n-gram author profiles, see [21] and [33] for a comprehensive review. While some of these features are effective, many have shown to have less discriminating power. We shall not discuss the advantages of different features here. Instead we refer all interested readers to the excellent survey articles by Juola [21] and Stamatatos [33] for a comprehensive summary of the pros and cons of different features in stylomtery analysis. It has been noted, however, that frequencies of content independent function words are among the most effective features [7].

The field of literary stylometry has seen impressive advances in recent years, with more and more new and sophisticated mathematical techniques as well as softwares being developed. Machine learning has proven to be a powerful tool in classification in last decade and researchers started to use it in authorship attribution. Koppel [26] used the exponential Gradient(EG) algorithm of Kivinen and Warmuth to find a linear separator between male-authored and female-authored documents. Burrows [7] [8] applied principal components analysis (PCA) on word frequencies to authorship attribution. Kjell [24, 23, 25] analyzed authorship using neural network classifiers, Bayesian classifiers and nearest neighbor classifiers. [12] applied support vector machine to identify authors of German newspaper articles. Jockers [20] even compared the performance of five methods (Delta, k-nearest neighbors, the support vector machine, nearest shrunken centroids, regularized discriminant analysis) in the classic authorship attribution problem involving the Federalist Papers.

1.3 Organization of the Thesis

The rest of thesis is organized as follows: we will first present an overview of machine learning. We will introduce classic Support Vector Machine in particular and its generalization to multi-class classification in chapter 2. Chapter 3 will introduce our study on the stylometry analysis, which focus on detecting chrono-divide in writing styles. We will describe our algorithm in detail and analyze experimental results. In chapter 4, we will show our method for authorship attribution problems where the true author is known in a small group and extend this method to open class problems where the candidate authors of a text is in large set or even no such set. We will report the accuracy of our method for close problems and show the robustness of our method for open class problems through extensive experiment and the testing results of three cases in last part of this chapter.

Chapter 2

Introduction to Machine Learning Techniques

2.1 Introduction

The main objective of machine learning, a subfield of computer science, is to learn the patterns of the given data which is represented by so-called features or descriptors. The underlying principle of machine learning is that the data from real life is not generated randomly, there always exist patterns in it, although we don't know exactly what they are. Machine learning can explore the patterns of the given training data and make prediction for the new data set. Machine learning is widely used in all kinds of area like face recognition, page ranking and speech recognition.There are three types of problems in machine learning in general:

- Supervised machine learning where the output of the training sample is given, and the goal is to predict the outputs of new samples;
- Unsupervised machine learning where no label is given for the training sample, and the goal is to find the structure of the training data;
- Reinforcement where the goal is given and the training data is dynamic.

Generally speaking, machine learning is the process of optimization. Until now, many optimization algorithms have been proposed for different type of problem in machine learning, such as Naive Bayes, Nearest Neighbors, Support Vector Machine(SVM) and so on. In this thesis, I will focus on SVM , which is proposed by Vapnik [35] in 1970s. It is the most popular machine learning algorithms since then. It is a supervised learning method which is used in both feature selection and pattern recognitions. SVM has been highly developed in both theory and algorithm and famous for its capability of dealing large data set.

2.2 Linear Support Vector Machine

2.2.1 Separable case

In the binary classification setting, given the training data set

$$\{(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\}, x_i \in R^n, y_i \in \{1, -1\}, i = 1, \ldots, l$$

we denote

$$\begin{aligned} I &= \{x_i | y_i = 1\} \\ II &= \{x_i | y_i = -1\} \end{aligned}$$

We say that I and II are linear separable if there is a unit vector w and constant b such that

$$\begin{aligned} & x_i^\top w + b > 1 && \forall\ x_i \in I \\ & x_i^\top w + b < -1 && \forall\ x_i \in II, \end{aligned} \tag{2.1}$$

i.e the data can be separated by the hyperplane

$$x^{\top}w + b = 0$$

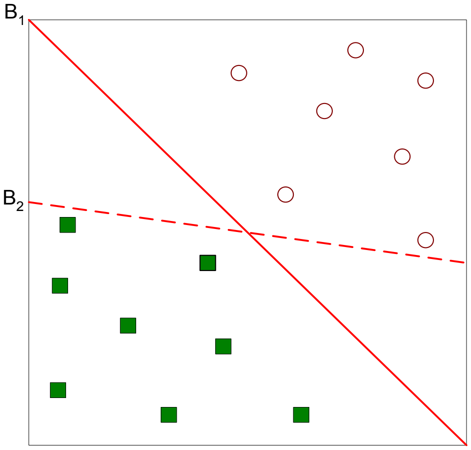


Figure 2.1: Hyperplanes

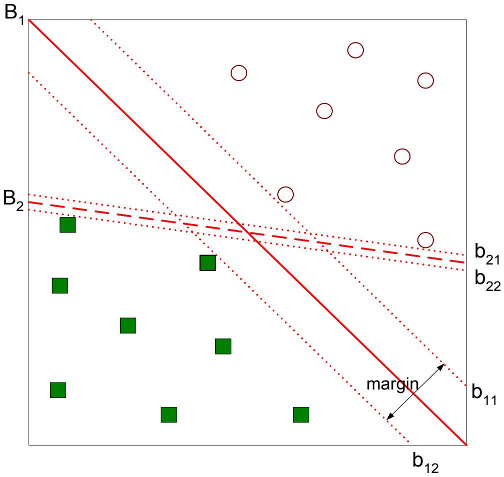


Figure 2.2: Optimal hyperplane

As we can see in figure 2.1, there will be infinite many hyperplanes which can separate the two groups. Which one should we choose? To best separate the two groups, we need to maximize the margin ρ (equation 2.2) as in figure 2.2, which is the distance between the hyperplane b_{11} and b_{12}. The hyperplane which separates the two groups of the training data set and has the maximal margin is called the *Optimal Hyperplane* and it is unique (see [36]).

$$\rho = \frac{2}{\|w\|_2} \tag{2.2}$$

Hence, the optimal hyperplane can be constructed as the following Primal optimization problem:

$$\begin{array}{lccl} \min\limits_{w \in H} & \frac{1}{2}\|w\|_2 & & \\ \text{s. t.} \quad y_i(x_i^\top w + b) & \geq & 1 & \forall i \in \{1, \ldots, l\} \end{array} \tag{2.3}$$

To solve this quadratic optimization problem, we need to find the saddle point of the Lagrange function:

$$L(w, b, \lambda) = \frac{1}{2} w^\top w - \sum_{i=1}^{l} \lambda_i [y_i(x_i^\top w + b) - 1], \tag{2.4}$$

where $\lambda_i \geq 0$ are the Lagrange multipliers.

According to the Fermat theorem, the minimumizer of this function has to satisfy the following condition:

$$\begin{array}{l} \frac{\partial L(w,b,\lambda)}{\partial w} = w - \sum\limits_{i}^{l} \lambda_i y_i x_i = 0 \\ \frac{\partial L(w,b,\lambda)}{\partial b} = \sum\limits_{i}^{l} \lambda_i y_i = 0. \end{array} \tag{2.5}$$

From equation 2.5, we get:

$$\begin{array}{l} w = \sum\limits_{i=1}^{l} \lambda_i y_i x_i \\ \sum\limits_{i=1}^{l} \lambda_i y_i = 0 \end{array} \tag{2.6}$$

Plug equation 2.6 into equation 2.4, we have changed the primal problem to its *dual* problem:

$$\begin{aligned} \max_{\lambda} \quad W(\lambda) &= \sum_i \lambda_i - \frac{1}{2}\sum_{i,j} \lambda_i \lambda_j y_i y_j x_i^\top x_j \\ \text{s. t.} \quad \sum_{i=1}^{l} \lambda_i y_i &= 0 \\ \lambda_i &\geq 0 \quad \forall i \in \{1, \ldots, l\} \end{aligned} \tag{2.7}$$

Thus, by solving equation 2.7, we can obtain λ_i. Meanwhile we get w by equation 2.6 and $b = \frac{1}{2}(\min_{x_i \in I} x_i^\top w + \max_{x_i \in II} x_i^\top w)$. The instances with $\lambda_i > 0$ are called *Support Vectors*, which are closest to the hyperplanes b_{11} and b_{12}. Finally, we obtain the decision function:

$$f(x) = w^\top x + b$$

and the sign of $f(x)$ is the predicted label of instance x.

2.2.2 Non-separable case

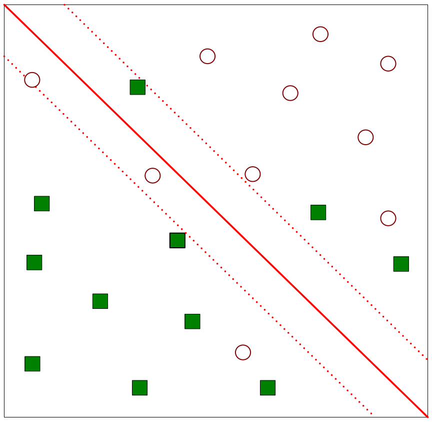


Figure 2.3: Linearly non-separable data set

In the previous part, we have showed how to find the optimal hyperplane for linear separable data, but in real life, most of data are either not linearly separable (e.g figure 2.3), or only separable with very small margins because of the presence of outliers. We will generalize the concept of optimal hyperplane for the non-separable case by relaxing the constraints 2.1 in this part. By introducing nonnegative *slack* variables $\xi_1, \ldots, \xi_l$ into constraints 2.1, we can construct the hyperplane with the smallest errors. Thus, the optimization problem becomes:

$$\begin{array}{rrcl} \min\limits_{w \in H, \xi \in R^l} & \frac{1}{2}\|w\|_2 + C \sum\limits_{i=1}^{l} \xi_i & & \\ \text{s. t.} & y_i(x_i^\top w + b) & \geq & 1 - \xi_i \\ & \xi_i & \geq & 0 \quad \forall i \in \{1, \ldots, l\} \end{array} \tag{2.8}$$

where $C > 0$ is the parameter that specifies the penalty of misclassification. When we set C to be infinitely large, the above problem is the same as the linear case. Similar to the linear separable case, we need to find the saddle point of the following Lagrangian in order to solve the above optimization problem:

$$\begin{array}{rrl} \max\limits_{\lambda} & W(\lambda) = & \sum_i \lambda_i - \frac{1}{2} \sum\limits_{i,j} \lambda_i \lambda_j y_i y_j x_i^\top x_j \\ \text{s.t.} & \sum\limits_{i=1}^{l} \lambda_i y_i = & 0 \\ & 0 \leq \lambda_i \leq C & \forall i \in \{1, \ldots, l\} \end{array} \tag{2.9}$$

This formula is similar to the linear case except for the upper bound of λ_i and it can be solved in the same way.

2.3 Nonlinear Support Vector machine

Until now, we have only discussed how to use SVMs to deal with linearly separable data sets. However, for the data in figure 2.4, which is not linear separable, we can extend the linear SVM to more general forms using “kernel” functions and separate the data with linear SVM in a new space. The basic idea of nonlinear SVM is to map the input vectors $\{x_i\}$ into the

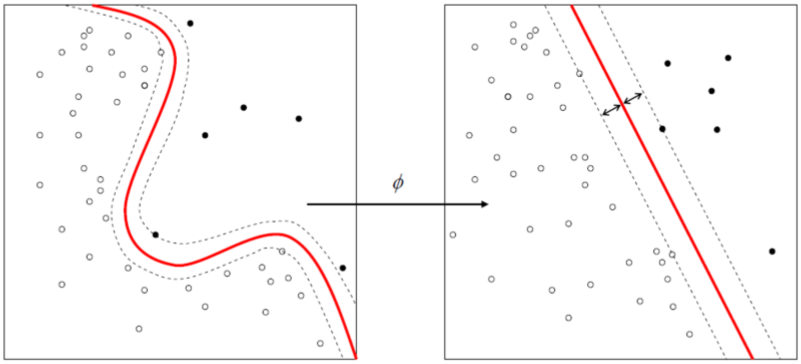


Figure 2.4: Linear SVM in new space

high dimensional space through nonlinear mapping ϕ. In the new space, we can construct the optimal hyperplane as in previous section, but nonlinear in the input space.

Instead of computing inner product $\langle x_i, x_j\rangle$, we now calculate $K(x_i, x_j) = \langle\phi(x_i), \phi(x_j)\rangle$, which is called a kernel function. Thus the optimization problem is:

$$\begin{array}{lrcl} \max\limits_{\lambda} & W(\lambda) & = & \sum\limits_{i} \lambda_i - \frac{1}{2}\sum_{i,j}\lambda_i\lambda_j y_i y_j K(x_i, x_j) \\ \text{s. t.} & \sum\limits_{i=1}^{l} \lambda_i y_i & = & 0 \\ & 0 \le \lambda_i \le C & & \forall i \in \{1, \dots, l\} \end{array} \tag{2.10}$$

Similarly, the decision function will be:

$$f(x) = w^{\top}\phi(x) + b = \sum_{i=1}^{l} \lambda_i y_i K(x, x_i) + b \tag{2.11}$$

and the sign of $f(x)$ is the the label of x.

Remark 1 *A kernel function K needs to satisfy Mercer's theorem, which requires that the matrix $K = (K(x_i, x_j))_{i,j=1}^{l}$ be symmetric and positive semi-definite. The following three types are typical:*

1. *Linear Kernel: $K(x, y) = \langle x, y\rangle$;*

2. Polynomial Kernel: $K(x,y) = (\langle x,y\rangle)^d$;

3. RBF Kernel: $K(x,y) = \exp(-\frac{\|x-y\|^2}{2\sigma^2})$.

Different kernels are used to deal with different types of classification problems. So for a given training data set without any knowledge of its geometric structure, it is better to try different models by choosing different kernels and their parameters carefully. We usually divide the training data into two parts, one of which is used to train the model and the other one (called validation data set) to test the model. We choose the stable model which yields the highest validation accuracy. For the data we use in our study of authorship identification, linear SVM proves to be the simplest as well as the best kernel compared to the other two types of kernels.

2.4 Multiclass Classification

In the previous section, we have discussed the classical SVM which solves binary classification problems . But in most scenarios, there are multiple categories for the training set, i.e.

$$\{(x_1, y_1), (x_2, y_2), \ldots, (x_l, y_l)\}, x_i \in R^n, y_i \in \{1, 2, \ldots, N\}, i = 1, \ldots, l$$

how to extend the binary SVM to the multi-class classification problem?

The basic and most straightforward idea is to decompose the problem into a series of binary classification problems, and then combine the results to obtain the multiclass classification. There are many techniques that have been developed so far, such as one-vs-rest, one-vs-one, decision directed acyclic graph(DAG) in [34], error correcting code in [13], single machine in [38] and so on. Crammer [10] and Vapnik [36] even consider all the classes to-

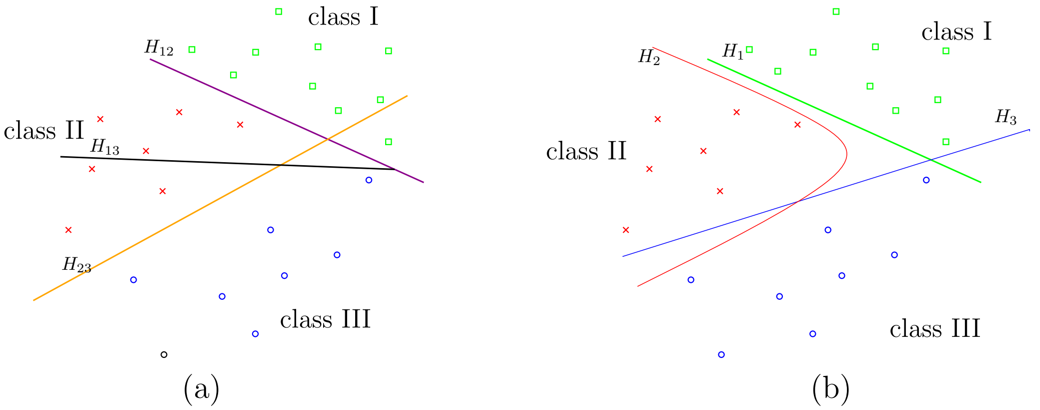


Figure 2.5: 3-class classification: (a) one-vs-one, H_{ij} separates class i and class j; (b) one-vs-rest, H_i separates class i from other classes.

gether in one optimization formulation, but it is computationally expensive and not widely used. In the mean time, the one-vs-rest and one-vs-one are so simple and effective that they have always been the most popular techniques for the multiclass classification problems. Here we will discuss one-vs-rest and one-vs-one in detail.

One-vs-Rest classification is to build N different binary classifiers $\{f_i\}_{i=1}^N$, each of which distinguishes one class from the remaining $N-1$ classes together viewed as a single class. To construct the i-th binary classifier, we let the samples from the i-th class be the positive samples, and other samples from the other $N-1$ classes be the negative samples. For test sample x, we attribute it to the class j_0 where $j_0 = \text{argmax } f_i(x)$. The advantage of this method is that we just need to solve N optimization problems to solve N class classification problem. But there are three main disadvantages using this method:

- there is no bound on the generalization error;
- the scale of the decision values may be quite different in different classifiers;
- the data is usually extremely imbalanced, where the negative sample dominates each classifier.

One-vs-One classification is to build one classifier for each pair of classes, which yields $N(N-1)/2$ classifiers $\{f_{ij}\}_{i,j=1}^{N}$ (note $f_{ij} = -f_{ji}$) in total. Each classifier f_{ij} is to distinguish classes i and j, and it can be viewed as the score of a "match" between class i and class j. A sample x is attributed to class i_0 through some voting scheme, for example we may choose $i_0 = \text{argmax } \sum_j f_{ij}(x)$, or i_0 be the class that has won the most number of "matches", $i_0 = \text{argmax } \sum_j \text{sign}(f_{ij}(x))$. Although this method has the advantage of more balanced training sample sizes for each classifier, it also has some disadvantages:

- There is no bounds on the generalization error;
- computationally it is expensive because it requires $N(N-1)/2$ classifiers;
- it can easily cause overfitting.

Both strategies have their advantages and disadvantages, so it is hard to conclude which one is always better. The performance of one-vs-rest and one-vs-one has been observed to be comparable in general [19], so we choose the one-vs-rest strategy considering the computational cost. Because the objective is the overall accuracy, we choose one single parameter set to maximize overall accuracy rather than selecting a different parameter set for each classifier. The latter is prone to over-fitting, in which case the training accuracy is high but testing accuracy is usually low.

2.5 Cross Validation

Cross validation is a strategy to assess the accuracy of a model when the test set is not available. The idea of one round of cross validation is to separate the training data into two parts, one of which is used to train the model, the remaining part (called validation data)

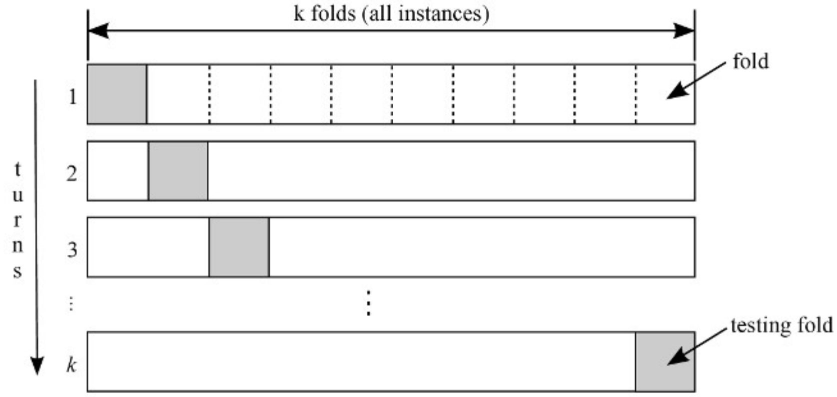


Figure 2.6: The k-fold cross validation

is used to predict the performance of the model. In k-fold cross validation (see 2.6), the training set is partitioned into k parts, in which $k-1$ parts are used to train the model, and the remaining part is used to test how accurate the model is. Each part works as test set once and only once, so there are k models constructed in k-fold cross validation.

The advantage of k-fold cross validation is that each sample test has one and only one chance to test the model. When the training data set is small or the set of parameter is very large, one model can easily lead to over-fitting. Cross validation can efficiently avoid this problem and help to select a better and stable model. However, cross validation usually requires additional computational cost. In this thesis, we use a 5-fold cross validation in the training process and apply the plurality voting scheme to label the test samples.

Chapter 3

Stylometry Anaysis: Detecting Chrono-Divide in Writing Styles

3.1 Chrono-divide

The main idea behind statistically or computationally-supported authorship attribution is that by measuring some textual features we can distinguish between texts written by different authors. Nearly a thousand different measures including sentence length, word length, word frequencies, character frequencies, and vocabulary richness functions had been proposed thus far [32] over the years. Some of these measures, such as frequencies of function words, have proven effective while others, such as length of words, have proven less effective [21]. The field of literary stylometry has seen impressive advances over the years, and has become an increasingly important research field in the digital age with the explosion of texts online.

This part focuses on a particular class of authorship controversies, in which there is a suspected change of authorship at some point of a book. In other words, one suspects that the first X chapters of a book were written by one author while the remaining Y chapters were written by another. Clearly, the authorship controversy for *Dream of the Red Chamber* falls into this category. Since no two authors have exactly the same writing style, no matter how similar they might be, a book written in such a fashion would have a stylistic discontinuity going from Chapter X to Chapter $X+1$. If we can quantify the styles of the two authors by a stylometric function $S(n)$ (a classifier) where n denotes chapters, or chronologically ordered samples, of the book in question, this stylistic discontinuity will appear as a dividing point in the stylometric function $S(n)$ going from $n = X$ to $n = X + 1$. Because the samples are ordered by time, we shall call this divide in the stylometric function $S(n)$ a *chrono-divide in style*, or simply a chrono-divide.

The underlying principle of our study is that if a book is in fact written by two authors A and B, then there should exist a group of features that characterize the difference of their respective styles. These features will lead to a stylometric function that separate the book into two different classes. In the rest of this part we shall use the more conventional term *classifier* for such a stylometric function. The foundational principle for literary stylometry is built around finding such classifiers. Suppose that a chrono-divide in style exists. Then an effective classifier will show a break point somewhere in the middle of the book, before and after which the classifier gives positive values and negative values, respectively. Thus in analyzing a book suspected to be written by two authors with a chrono-divide, one can look for a classifier that gives rise to such a break point. The existence of such a classifier will provide strong support for the two-author hypothesis. Conversely, if such a classifier cannot be found then we can confidently reject the two-author with a chrono-divide hypothesis.

We use function characters and words to build and select a group of stylometric features having the highest discriminative power, and from which we construct our classifier. We shall detail our method in the following subsections.

3.2 Methodology

3.2.1 Initial stylometric feature extraction

Suppose the book in question is suspected to be written by two authors. For simplicity we shall call the part written by author A *Part A* and the part written by author B *Part B*. In many cases, such as with *Dream of the Red Chamber*, both Part A and part B are known. In some cases, they are not precisely known like *Micro*. However, for books suspected to have a chrono-divide from authorship change, there is usually a good estimate for where the divide is. Typically the first few chapters can be confidently attributed to A and the last few chapters to B.

We begin by choosing a feature set consisting of the kinds of features that might be used consistently by a single author over a variety of writings. Typically, these features include the frequencies of words (or characters for books in Chinese), phrases, mean and variation of sentence length, and frequencies of direct speeches and exclamations, and others. In our analysis, to get a better understanding of an author's writing style, we first find the most frequently used characters and words in the book, e.g. we would find the 500 most frequently used characters in the whole book, from which we pick out only, say, n function characters. We choose m words (combinations of characters) among the 300 most frequently used words in the same way. An important point is that by selecting only function characters and words we obtain a selection of characters and words that are *content independent*. This leads to an initial set of features consisting of the frequencies of the n characters and the m words, plus the mean and variance of sentence length as well as the frequencies of direct speeches and exclamations. These features will be computed over given sample texts of the book (e.g. chapters). We normalize each sample text in the following way: set the median of the mean and variation of sentence length and the frequencies of direct speeches, exclamations, n characters and m words in each work of A and B to be 1. For each sample, we now get $n + m + 4$ features.

3.2.2 Data preparation

Having constructed the appropriate feature vectors, we build a distinguishing model through a machine learning algorithm. To do so requires careful data preparation. Since we usually have in hand only limited samples while the number of features will be very large, building a model directly on the entire book will easily lead to over-fitting. To overcome the over-fitting problem, we use the standard technique of separating the whole data into samples consisting of training data and test data. Our model will be established based only on the training data while its performance is tested over the independent test data. If we know Part A and Part B already then a subset of each can be designated as training data. For books suspected to have a chrono-divide in style, the training data will consist of the first few chapters and the last few chapters. The rest of the book will be used as test data.

In order to obtain more training sets and testing sets we shall chunk the book in question into smaller pieces of sample texts of relatively uniform size and style. In all the books we have studied, we have kept the sample texts to be at least 1000 characters long. In the case of *Dream of the Red Chamber* each sample text is a chapter.

3.2.3 Feature subset selection

When we build authorship analysis model using the training data only, we do not use all the features ($n + m + 4$ features). Instead we start out with all of them, but eventually select a subset of features that achieves the highest discriminative powers. Feature subset selection has been well understood for high dimensional data analysis in the machine learning context. First, the number of discriminative features may be small because the number of features an author uses in a consistently different way from others is usually not very big. Moreover, the classifier can perform very poorly if too many irrelevant features are included into the model. In this paper we will use Support Vector Machines Recursive Feature Elimination (SVM-RFE) introduced in [17] to realize feature selection.

SVM-RFE is a feature ranking method. Given a set of samples we can use linear SVM to build a linear classifier. It ranks the importance of the features according to their weights. As mentioned above, because of large feature size and small sample size, the classifier might not be robust. In addition, the high correlation between the features may result in small weights for relevant features. Thus the ranking by SVM classifier directly may be inaccurate. In order to refine the ranking, the least important feature is removed and the linear SVM classifier is retrained. This new classifier provides a refined ranking for the remaining features. The process is then repeated until the ranking of all features are refined. This is the SVM-RFE method introduced in [17]. The idea underlying SVM-RFE is that in each repeat, although the overall ranking may be poor, the least important feature is very unlikely a relevant one. By iteratively eliminating the least important features the new classifiers will become more and more reliable and hence will provide better and better ranking. In the application of gene expression data analysis SVM-RFE has been proven to be substantially superior to the SVM direct ranking without RFE.

However in general SVM-RFE is not stable under the perturbation of samples. A small change in samples may result in very different feature ranking. There are two possible reasons. One is that the highly correlated variables are too sensitive and may be ranked in different orders by different classifiers. Another is that, due to the randomness, some subset of samples might be singular in the sense that they are less representative for the whole data structure. As a result the SVM classifiers are over-fitting and the feature ranking by SVM-RFE is therefore unreliable. The first situation is less harmful for classification performance while the second is vital. To overcome this phenomenon and guarantee the stability of the ranking, we use a pseudo-aggregation technique. We randomly choose a subset of training samples to run SVM-RFE to select the top important features. This process is repeated tens or hundreds times and only those features that appear important very frequently are deemed as truly important ones. This removes the randomness and results in a much more reliable ranking.

With this ranking of features, we can conclude which statistics are useful for quantifying the writing style. We use cross validation to select the number of features included in the final classification model. This group of features is a stable and most discriminative subset of features. A final classifier is built to classify the test data.

3.2.4 Data analysis

The classifier we have built is used to analyze the authorship question. We examine the discriminative power of the classifier on the training data. If it cannot even reliably classify the training data we can convincingly reject the two-author hypothesis. Even if it can the telling story will be whether it can classify, or detect a chrono-divide, from the test data. If it fails then again we should reject the two-author hypothesis. On the other hand, if the classifier classifies the training data, and it can also classify the test data accurately or detect a clear chrono-divide, we can then convincingly conclude that the book does contain two different writing styles and can therefore be confidently attributed to two different authors.

Moreover, the feature subset and the classifier describe the difference of the two authors' writing styles.

3.2.5 The algorithm

In the following we summarize the process of our algorithm:

1. Initialize the data (the book), which contains parts A and B suspected to be written by two different authors.

2. Split part A and part B into many sections and extract the features for each section as described in section 3.2.1. This forms the whole data set D, containing D_A and D_B.

3. Choose a portion (e.g. 20%-30%) of D_A and D_B respectively to form the test data set and leave the remaining as the training data set. The test data will not be used until the final model is built.

4. Randomly choose a subset from the training data as modeling data and the rest (again 20%-30%) as the validation data. Run SVM-RFE on the modeling data and use the validation data to determine all the parameters used. This provides a ranking of all the $n + m + 4$ features extracted in step 2.

5. For d range from 1 to $n + m + 4$, build a classifier using only the top d features and evaluate their performance on the validation data. The best model is the one with minimal validation error and minimal number of top features. The feature subset of this best model is recorded.

6. Repeat T times step 4 and step 5 to obtain T best models and T subsets of corresponding important features. We recommend T to be larger than 50. Rank all the

features in these subsets according to their appearance frequency. Denote N as the total number of features included.

7. For $d = 1, ..., N$, using cross validation to select the number of features that should be included in the final classifier. Denote it by d_*. Note we require both the cross validation error and the number of features to be as small.

8. Retrain the model using the whole training set based on this top d_* important features.

9. Using the classifier to classify the test data. Draw the conclusion according to the performance.

Since our ranking process involves aggregation of large number of models that are trained using SVM-RFE based on different subsets of the same data source, we refer to our approach as pseudo-aggregation SVM-RFE method.

3.3 Case Study: Analysis of *Dream of the Red Chamber*

3.3.1 Background

Dream of the Red Chamber by Cao Xueqin is one of China's Four Great Classical Novels. For more than one and a half centuries it has been widely acknowledged as the greatest literary masterpiece ever written in the history of Chinese literature. The novel is remarkable for its vividly detailed descriptions of life of China in the 18th century during the Qing Dynasty and the psychological affairs of its large cast of characters. There is a vast literature in *Redology*, a term devoted exclusively to the study of *Dream of the Red Chamber*, that touches upon virtually all aspects of the book one can imagine, from the analysis of even minor characters in the book to in-depth literary study of the book. Much of the scope of Redology is outside the focus of this paper.

The original manuscript of *Dream of the Red Chamber* began to circulate in the year 1759. The problems concerning the text and authorship of the novel are extremely complex and have remained very controversial even today, and they remain an important part of Redology studies. Cao, who died in 1763-4, did not live to publish his novel. Only hand-copied manuscripts – some 80 chapters – had been circulating. It was not until 1791 that the first printed version was published, which was put together by Cheng Weiyuan and Gao E and was known as the *Cheng-Gao version*. The Cheng-Gao version has 120 chapters, 40 more than the various hand-copied versions that were circulating at that time. Cheng and Gao claimed that this "complete version" was based on previously unknown working papers of Cao, which they obtained through different channels. It was these last 40 chapters that were the subject of intense debate and scrutiny. Most modern scholars believe that these 40 chapters were not written by Cao. Many view those late additions as the work of Gao E. Some critics, such as the renowned scholar Hu Shi, called them forgeries perpetrated by Gao, while others believe that Gao was duped into taking someone else's forgery as an original work. There is, however, a minority of critics who view the last 40 chapters as genuine.

The analysis of the authenticity of the last 40 chapters has largely been based on the examination of plots and prose style by Redology scholars and connoisseurs. For example, many scholars consider the plotting and prose of the last 40 chapters to be inferior to the first 80 chapters. Others have argued that the fates of many characters in the end were inconsistent with what earlier chapters have been foreshadowing. A natural question is whether a mathematical stylometry analysis of the book can shed some light on this authenticity debate.

Although there is a vast Redology literature going back over 100 years, the number of studies of the book based on mathematical and statistical techniques is surprisingly small, particularly in view of the fact that such techniques have been used widely in the West for settling authorship questions. Among the notable efforts, Cao [30] meticulously broke down a number of function characters and words into classes according to their functions. By analyzing their frequencies in the first 40 chapters, the middle 40 chapters and the last 40 chapters, Cao concluded that the first 80 chapters and the last 40 chapters were written by different authors. Zhang & Liu [37] examined the occurrence of 240 characters in the book that are outside the GB2312 encoding system, and found that 210 of them have appeared exclusively in the first 80 chapters while only 20 of them have appeared exclusively in the last 40 chapters. This led to the same conclusion by the other authors. Yu [31] studied the authorship by combining both historical knowledge and statistical tools. In the study Yue tested two hypotheses, that the last 40 chapters were not written by the same author or they were written by the same author. His study focused on the frequencies of 5 particular function characters, the proportion of texts to poems in each chapter, and a few other stylometric peculiarities such as the number of sentences ended with the character "Ma". Using various statistical techniques the comparisons led the paper to draw the conclusion that it is unlikely that the first 80 chapters and the last 40 chapters were written by the same author. At the same time, using historic knowledge about the book and the original author Cao Xueqin, the paper also speculated that it was not likely that the last 40 chapters were created entirely by a single different author such as Gao E. In the opposite direction, the studies of Chan [9] and Li & Li [16] concluded that the entire book was likely written by a single author. The study [16] focused on the usage of functional characters while [9] examined the usage of some eighty thousand characters. Both studies tabulated the frequencies of the selected characters, which led to a frequency vector for each of the first 40 chapters, the middle 40 chapters and the last 40 chapters. The correlations of these frequency vectors were computed. In [16] the correlations were found to be large enough for the authors to conclude that the entire 120 chapters of the book were written by the same author. In [9] a fourth frequency vector using parts of the book *The Gallant Ones* was added for comparison. The author found significantly higher correlations among the first three frequency vectors from chapters of *Dream of the Red Chamber* than the correlations between the fourth frequency vector and the first three. This fact formed the basis of the conclusion by the author that all 120 chapters were written by a single author. A different conclusion was reached by Li [40]. By analyzing the frequencies of 47 functional characters and applying several statistical tests the author conjectured that the last 40 chapters were put together by Gao E using unedited and unfinished manuscripts by Cao Xueqin and his family members.

Although some of these aforementioned studies are impressive in their scopes, missing conspicuously from the Redology literature are studies based on the latest advances in literary stylometry, particularly some of the new and powerful methods from machine learning theory. While comparing the frequencies of function characters and words is clearly a viable way to analyze the authorship question, care needs to be taken to account for random fluctuations of these frequencies, especially when some of the function characters and words used for comparison have limited occurrences overall in the book and sometimes not at all in some chapters. None of the aforementioned studies employed cross validation to address random fluctuations. We have substantial reservations about drawing conclusions from correlations alone as in the studies of Chan [9] and Li & Li [16], because the differentiating power of any single variable such as correlation is rather limited. It would be interesting to see a more comprehensive study of correlations on a large corpus of texts in Chinese to determine its effectiveness as a metric for authorship attribution, something the authors failed to do in both studies. The use of the book *The Gallant Ones* in [9] for benchmark comparison is curious to us in particular, especially considering that the author did not limit to just function characters. The two books are of two different genres and are different in their respective background settings. Considering these differences *and* the fact that *The Gallant Ones* is known *not* to be written by Cao Xueqin, it would be a shock if the correlation between the last 40 chapters of *Dream of the Red Chamber* and the first 80 chapters is *not* higher than the correlation between the last 40 chapters and *The Gallant Ones*. It is possible that the correlation computed in [9] tells more about the genre than the authorship of the books. Again, without extensive evidence that using the same technique the correlation between two bodies of texts written by different authors is generally low even when the plots are closely related, the argument made in [9] is unconvincing at best.

Having established a rigorous protocol for finding chrono-divides, we are now in position to apply this protocol to investigate the authorship controversy of the Cheng-Gao version of *Dream of the Red Chamber*. In particular we investigate the existence of a chrono-divide at Chapter 80.

3.3.2 Separability of the chapters by Cao and Gao

The book is first divided into samples. To balance the number of samples, we generate one sample for each of the first 80 chapters while using the conventional practice of duplicating each of the last 40 chapters into two chapters to obtain 80 samples. From those samples we extract the features by calculating the statistics proposed in subsection 3.2.1. These features are then normalized for fair comparison. In total we have 196 variables. They are the 144 characters and 48 words, the normalized mean and variation of sentence length, and the frequencies of direct speeches and exclamations.

To investigate the authorship controversy we perform three separate tests. First we build a classifier for the whole book and look for the existence of a chrono-divide at Chapter 80. For added robustness and reliability we also perform the same tests only on the first 80 chapters and the last 40 chapters.

In the first experiment we apply our method to the whole Chen-Gao version of *Dream of the Red Chamber*. Samples from the first 60 chapters are designated as training samples for one class while samples from the last 30 chapters are designated as training samples for another class. The remaining samples, from Chapter 61 to 90, are held out as testing samples. The training samples are further randomly split into modeling data of 80 samples and validation data of 40 samples. The SVM-RFE is repeated 100 times and d_* is chosen using 50 cross validation runs. We have the following observations.

Instability of SVM-RFE. The randomness of the modeling set has resulted in very substantial fluctuations in the number of features selected as well as feature rankings. The resulted classifier may also perform quite differently. Table 3.1 lists the features selected using two different modeling data sets. One selects 11 features and the other selects only 4, with only one feature in common. The classifiers also perform differently. The experiments clearly establish the instability of SVM-REF.

Given such instability one cannot reliably draw any conclusions from any single run. For example, if a modeling data set separates the training data well it might be due to over-fitting. Conversely if it separates poorly it might be due to under-fitting. This problem is overcome with our Pseudo Aggregate SVM-RFE method.

Modeling set	Features Selected	Validation Error
1	qu, de, jiu, hui, zhi, dao, shi, ne, bie, zuo	5/40
2	hui, fang, mei, haoxie	1/40

Table 3.1: The features and validation errors of the classifiers obtained from two randomly selected modeling subsets.

Stability of Pseudo Aggregate SVM-RFE. Our pseudo aggregate SVM-RFE approach repeats SVM-RFE 100 times using randomized data sets. The data set from each repeat is used to select a set of features, from which a classifier is being built. For simplicity we shall refer to the data set, features and the resulting classifier together from a repeat as a *model*. To counter random fluctuations we consider important features to be those that appear frequently among the 100 classifiers. This reduced the instability caused by randomness. In fact, our belief is as follows: if the two classes are well separated, there should exist a set of features that help to build a good classifier. Most modeling subsets should be able to select these features out and only a limited number of modeling sets might be singular and miss them. Conversely, if the two classes cannot be well separated, no consistently discriminative features exist. Different modeling set may lead to totally different feature subset. As a result, no feature appears with high frequency in all 100 models. This philosophy, however, is only partially true. When the two classes cannot be separated, the modeling process sometimes can overfit the data by selecting a lot of variables which results in high absolute frequencies for some less important or irrelevant features. Such a phenomenon is usually accompanied by large number of variables and low validation accuracy. To improve the process we propose a more appropriate metric, which we call *relative frequency*. In relative frequency we weight the frequency by two criteria. In the first criteria a variable appearing in short models is weighted more than the variables appearing in long models. This leads to a weight of $h(n_j)$ for a variable in the jth model, with n_j being the number of variables in the jth model. In the second criteria a variable in a model with high predictive accuracy is weighted more than a variable with poor predictive accuracy. This provides another weight $g(A_j)$ for a variable in the jth model, where A_j denotes the accuracy of the jth model computed from the validation process. Mathematically the relative frequency for a variable x_i in a test run of M repeats is defined as

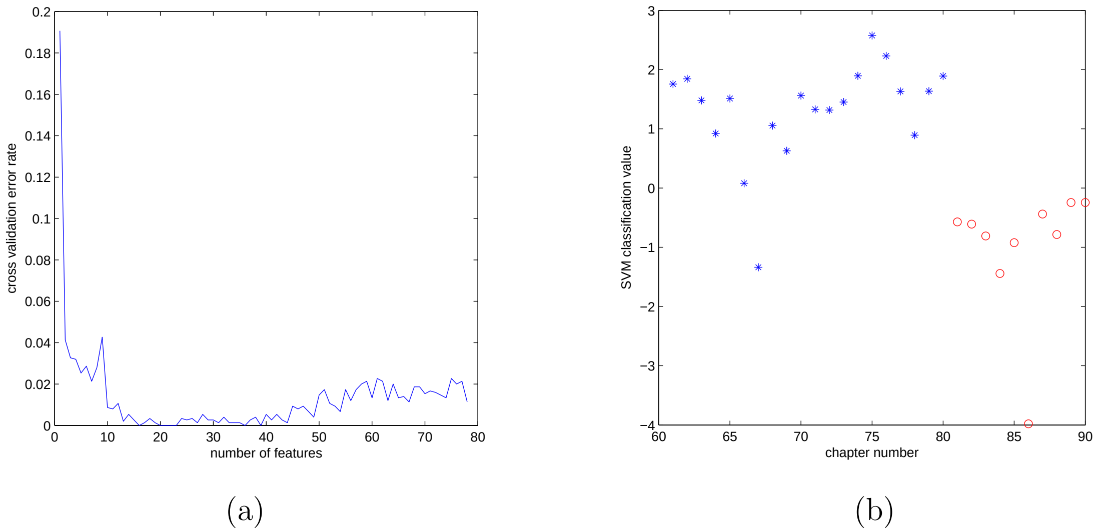
$$rf(x_i) = \frac{1}{M}\sum_{j=1}^{M} g(A_j)h(n_j)\mathbf{1}(x_i \text{ appears in model } j). \tag{3.1}$$

In our study we always set $M = 100$. Also, we set $g(A_j) = \exp(\frac{A_j - 1}{[2A_j - 1]_+})$ where $[t]_+ = \max\{0, t\}$ and $h(n_j) = [1 - cn_j]_+$ for some constant c. For $g(A_j)$ the idea is that if the weight should decay fast if the accuracy is close to 50% or less because it indicates that the classifier is simply not effective. For $h(n_j)$ we put in a penalty for the number of variables used in a model. In our experiments we have chosen $c = 1/30$, which seems to work well.

Our experiments show that features yielded from relative frequency rankings are very stable and consistent. We have performed runs of 100 repeats using different random seeds in MATLAB, and the results are always similar. An additional benefit of using relative frequency instead of absolute frequency is that the existence of an effective classifier is typically accompanied by high relative frequencies for the top features, while low relative frequencies for the top features usually imply poor separability. Hence we can use relative frequency as a simple guide on the separability of the samples. We will show some examples in the next section.

Results and Conclusion. In experiment 1 we have performed a run of 100 repeats on the entire Cheng-Gao version of *Dream of the Red Chamber*. Altogether 70 features have appeared in at least one model. However, of those only a small number of them have appeared with high enough frequency to be viewed as being important. We apply cross validation to select the number of features, and the mean cross validation error rate against different number of features is plotted in Figure 3.1 (a). The figure tells us that 10 to 50 features are enough to tell the style difference between the two parts. Using less characters and words is insufficient, while using more degrades the performance also by bringing in too much noise. The small cross validation error rate is encouraging, and it is already hinting a strong possibility that the two training sample sets have significant stylistic differences to support the two-author hypothesis.

To settle the two-author hypothesis more definitively we apply our classifier on the test data, which until now has never been used during the feature selection and classifier modeling process. In particular we investigate the existence of a chrono-divide in the values obtained through classifier. Figure 3.1 (b), which plots these values, clearly shows a chrono-divide at Chapter 80: For Chapter 81-90 the classifier yields all negative values while for Chapters 61-80 the classifier yields all positive values with the exception of Chapter 67. Allowing some statistical abberations to occur, our results provide an extremely convincing if not irrefutable evidence that there exist clear stylometric differences between the writings of the first 80 chapters and the last 40 chapters. This difference strongly supports the two-author hypothesis for *Dream of the Red Chamber*. We also note that our investigation did not need to assume that the knowledge that the stylistic change should be at Chapter 80. The fact that the chrono-divide we have detected is indeed at Chapter 80 lends even stronger support to the two-author hypothesis.



(a)

(b)

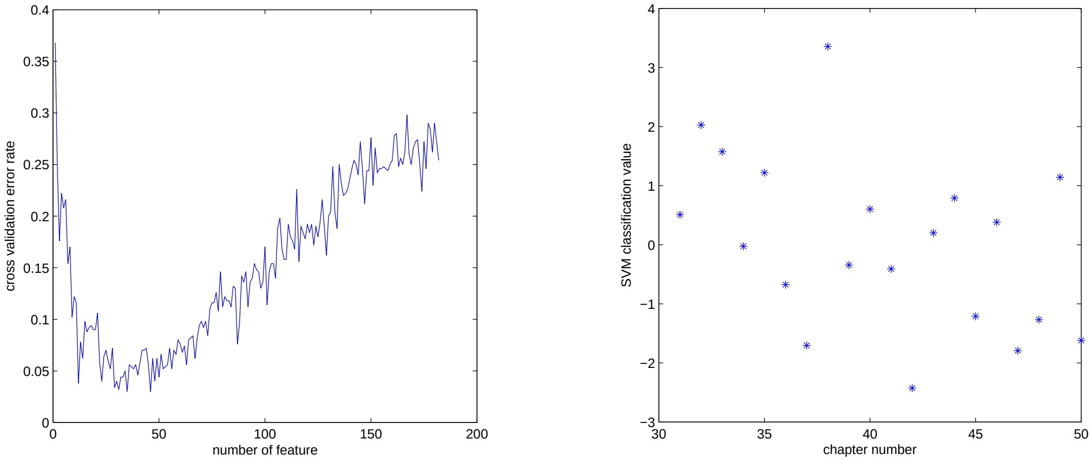
Figure 3.1: Experiment 1: (a) Mean cross validation error rate; (b) Values of SVM classifier on chapters 60-90.

Interestingly, the fact that Chapter 67 appeared as an “outlier” in our classification serves as further evidence to the validity of our analysis. It was only after the tests we realized that the authorship of Chapter 67 itself is one of the controversies in Redology. Unlike the main controversy about the authorship of the first 80 chapters and the last 40 chapters, experts are less unified in their positions here. Again, our results strongly suggests that Chapter 67 is stylistically different from the rest of the first 80 chapters, and it may not be written by Cao. Our finding is consistent with the conclusion of [2].

3.3.3 Non-separability of the first 80 chapters

To further validate our method we apply the same tests to the first 80 chapters of *Dream of the Red Chamber* to see whether we can get a chrono-divide (Experiment 2). We use the first 30 and last 30 chapters as the training data and leave chapters 31-50 as the test data.

Figure 3.2 shows the mean cross validation error and the values of SVM classifier on the test data chapters 31-50. The experiment shows much more features have been selected in the 100 repeats, implying the difficulty of find a consistent subset of discriminative features. The large errors on the training data also indicate the difficulty for separation. When the classifier is applied to the test data, there is clearly no chrono-divide. This suggests that our method yields a conclusion that is completely consistent with what is known.



(a)

(b)

Figure 3.2: Experiment 2: (a) Mean cross validation error rate; (b) Values of SVM classifier on chapters 31-50. Note there is no chrono-divide.

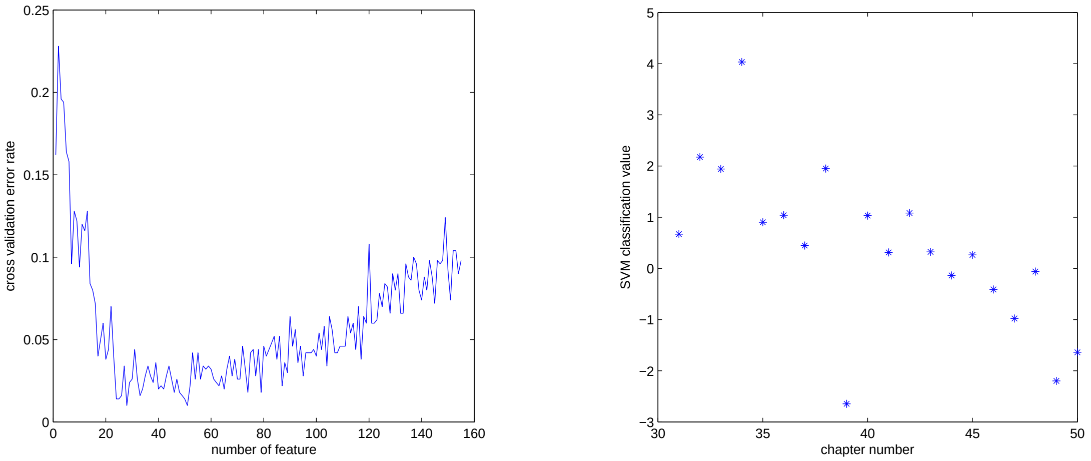
3.3.4 Analysis of chapters 81-120: style change over time

We next apply our method to the last 40 chapters (Experiment 3). Our first experiment has already confirmed that they are unlikely to be written by Cao. However, there are still debates on whether these were written entirely by one author (most likely Gao himself), or by more than one author. Our mathematical analysis may offer some insight here.

We split the 40 chapters into two subsets as before. The training data include Chapters 81-95 as one class and Chapters 106-120 as another. The test data are the middle 10 chapters. Because of the relatively small number of samples we have subdivided each chapter into 2 sections to increase the sample size. As a result we now have 60 samples in the training data and 20 in test data, with 2 samples corresponding to one chapter. The mean cross validation error of the final classifier and its classification values on the test samples are shown in Figures 3.3 (a) and (b) respectively.

In this experiment we observe that the performance in terms of both the classifier and feature ranking is noticeably worse than that in Experiment 1 but substantially better than that in Experiment 2. Furthermore, unlike the results from the first two experiments, the values from the classifier show an interesting trend. Compared with Figure 3.2 (b) where the values appeared to lack any order, the values here exhibit a clear gradual downward shift. On the other hand, compared to Figure 3.1 (b) the values plotted in Figure 3.3 (b) do not show a clear sharp chrono-divide, even though the values change gradually from being positive to being negative. What it tells us is that the writing style of the last 80 chapters had undergone a graduate change, but this change is unlikely to be due to change of authorship.

Our results here could be subject to several interpretations. One plausible interpretation is that Gao might indeed obtained some incomplete set of manuscripts by Cao, and tried to complete the novel based on what he had obtained. The style change is a result of the lack of genuine work by Cao as the story developed. A more plausible interpretation is that the last 40 chapters were written by someone such as Gao trying to imitate Cao's style, and over time the author became sloppier and returned more and more to his own style.



(a)

(b)

Figure 3.3: Experiment 3: (a) Mean cross validation error rate; (b) Values of SVM classifier on chapters 96-105, which correspond to the samples 31-50 in all 80 samples. Note two samples come from one chapter in this experiment.

3.3.5 Comparison with *Continued Dream of the Red Chamber*

It is worth mentioning that there are several other attempts to complete *Dream of the Red Chamber* from its first 80 chapters, among them is *Continued Dream of the Red Chamber* by Qi Zichen. Using the same features for building the classifier in Experiment 1, we can compute the Euclidean distances between all chapters and their distances of chapters from *Continued Dream of the Red Chamber*, see Figure 3.4. Surprisingly, although these features are obtained in favour of the differences between Cao and Cheng-Gao, they lead to even larger distance between the first 80 chapters and those chapters of *Continued Dream of the Red Chamber*. It obviously implies that the style of the 40 chapters by Cheng-Gao is more similar to the 80 chapters by Cao compared to *Continued dream of the Red Chamber*. Maybe that's why the Cheng-Gao version is more popular than other versions.

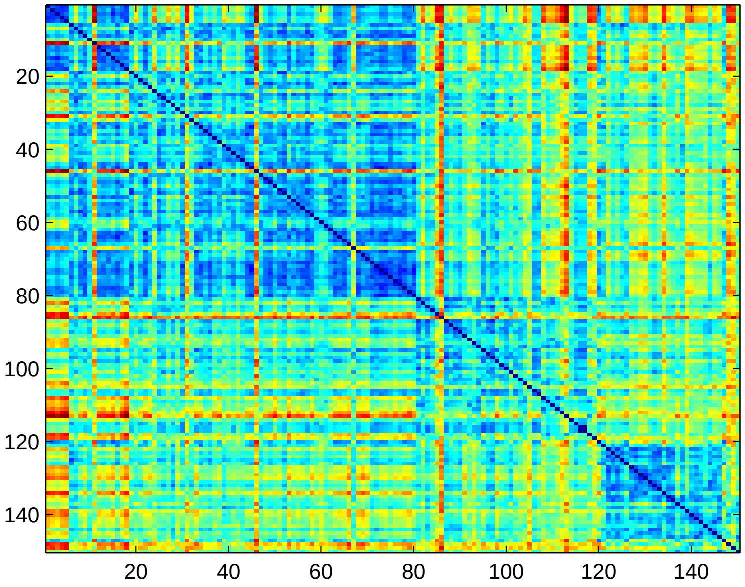


Figure 3.4: Distances between the first 80 chapters of the Cheng-Gao version, the last 40 chapters of the Cheng-Gao version, and 30 chapters of *Continued Dream of the Red Chamber*.

3.4 Case Study: Analysis of the other three Great Classical Novels

To further bolster the credibility of our approach we test our method on the other three Great Classical Novels in Chinese literature, *Romance of the Three Kingdoms*, *Water Margin*, and *Journey to the West*. Unlike *Dream of the Red Chamber*, there is no authorship controversy for these three novels. Thus if our method is indeed robust we should expect negative answers for the two-author hypotheses for all of them by finding no chrono-divides.

As with *Dream of the Red Chamber*, we split each of the three novels into training samples and test samples. Both *Romance of the Three Kingdoms* and *Water Margin* have 120 chapters. In both cases we designate the first 30 chapters and the last 30 chapters as the two classes of training data, and the middle 60 chapters as test data. For *Journey to the*

West the two classes of training data are the first and last 25 chapters respectively, with the middle 50 chapters as test data.

We use the same procedure to test for chrono-divides on the three novels. Compared to *Dream of the Red Chamber*, the selected features show much lower relative frequencies, indicating difficulty in differentiating between the writing styles. Table 3.2 show the relative frequencies (with $c = 1/30$) of the top 8 features for each of the four Great Classical Novels. Also of note is that in the case of *Water Margin*, 51 features are used to build a classifier from the 60 training data, which is clearly another strong indication of the difficulty.

Novel	Relative frequencies of top 8 features							
Dream of the Red Chamber	0.57	0.46	0.43	0.36	0.31	0.30	0.29	0.19
Romance of the Three Kingdoms	0.31	0.27	0.26	0.25	0.23	0.22	0.17	0.15
Water Margin	0.18	0.17	0.16	0.16	0.14	0.11	0.11	0.10
Journey to the West	0.03	0.03	0.02	0.02	0.02	0.02	0.02	0.02

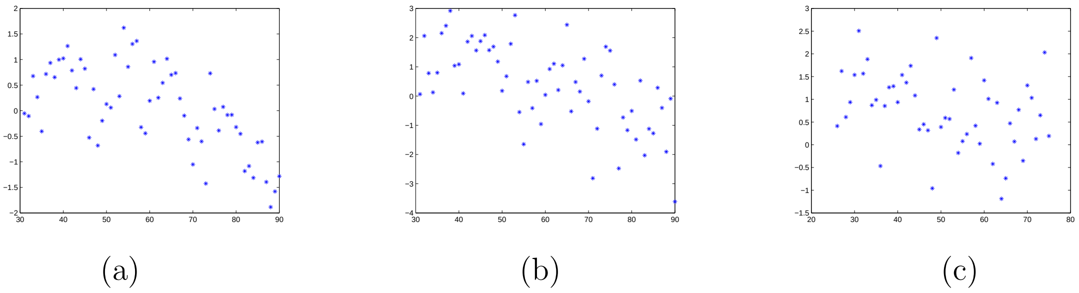
Table 3.2: Relative frequencies of the top ranked 8 features in each of the four Great Classical Novels.

Figure 3.5 plots the values from the classifiers for all three novels. In all cases the values fluctuate in such a way that it is quite clear that no chrono-divides exist, as expected.

This analysis shows that our approach can reliably reject the two-author hypothesis when it is false, lending further support to the effectiveness and robustness of our method.

3.5 Case Study: Chrono-divide of *Micro*

Micro, a techno-thriller published posthumously in 2011, is Michael Crichton's final novel. It was found on his computer upon his death in 2008 as an unfinished manuscript. Harper Collins commissioned science-writer Richard Preston to complete the novel from Crichton's notes and research. Although *Dream of the Red Chamber* is in Chinese, the principle of our



(a) (b) (c)

Figure 3.5: Classification results on the testing samples of the other three classical novels: (a) *Romance of the Three Kingdoms*; (b) *Water Margin*; (c) *Journey to the West.*

method should apply to books in other languages. *Micro* thus provides us with a good test example. In this case study, we will use our approach to confirm and detect the chrono-divide of *Micro.* We will also perform a different new test using classifiers built directly from other books written by Crichton and Preston for comparison. The new test serves both as a validation of our method and as a comparison. Note that the second option is not available for *Dream of the Red Chamber.*

In the direct classifiers test we use the other books written by Crichton and Preston to generate the training data. A total of 17 books written by Crichton and 2 books by Preston were used for training. The initial features consist of the frequency of 241 most frequently used words in these books. To build classifiers each book was divided into multiple pieces with each piece containing approximately 2000 words. The frequencies of the the 241 selected words of each piece form a sampling point. Overall 782 data points for Crichton and 104 data points from Preston were generated. To overcome the imbalance of the sampling points for Crichton and Richard, we only used 728 samples for Crichton and they are split into 7 subsets. Each subset is combined with the samples for Preston to form a training data set, from which we build a linear classifier. So totally 7 classifiers were constructed. Applying the classifiers to detect the chrono-divide in *Micro*, we chunk the book into 56 parts, each containing about 2000 words. Each part provides a testing sample point. We applied the 7 classifiers to this testing data. The average of the 7 classifiers are plotted in Figure 3.6. The result shows a break point at around the 15th-16th sample points.

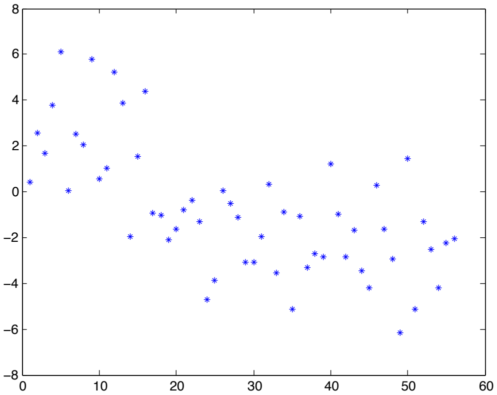


Figure 3.6: *Micro*: average values of the 7 classifiers.

We can now compare the above method to the earlier method for *Dream of Red Chamber*. We assume that, compared with the overall style of an author across multiple books, the style of the author in a single book would be more consistent. As a result we divided *Micro* into 112 parts of approximately 1000 words each. The most frequently used 265 content independent words from the book were used as the initial features. We use the first 22 sample points and the last 22 sample points as training and validation data and the middle 68 sample points as test data. The classification results are shown in Figure 3.7. A clear break point can be seen around the 29th-30th samples.

The two experiments confirmed the existence of a chrono-divide in *Micro*, and provide further evidence of the validity of our original approach for discovering and locating chrono-divides. As a by product, our results show that the change of authorship for *Micro* had occurred between 1/4 and 1/3 of the book. This is consistent with what Richard Preston had indicated in several interviews about the book.

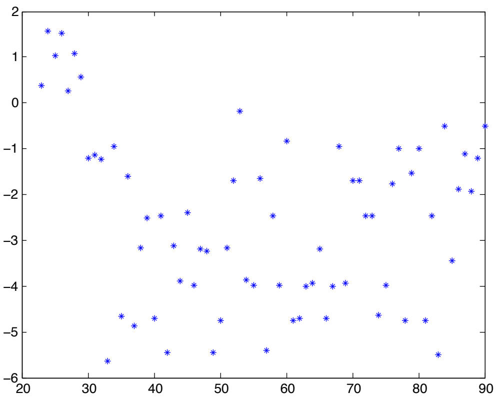


Figure 3.7: *Micro*: Classification result by the classifier obtained using the first 22 parts and the last 22 parts of the book as training samples.

3.6 Conclusion

Inspired by authorship controversy of *Dream of the Red Chamber* and the application of SVM in the study of literary stylometry, we have developed a mathematically rigorous new method for the analysis of authorship by testing for a chrono-divide in writing styles. We have shown that the method is highly effective and robust.

Applying our method to the Cheng-Gao version of *Dream of the Red Chamber* has led to convincing if not irrefutable evidence that the first 80 chapters and the last 40 chapters of the book were written by two different authors. Furthermore, our analysis has unexpectedly provided strong support to the hypothesis that Chapter 67 was not the work of Cao Xueqin either.

Applying our method to *Micro*, we are able to confirm the existence of chrono-divide and identify its location. It provides strong evidence for us to attribute the first 1/4 of the work to Michael Crichton and the left 3/4 to Richard Preston.

The robustness of our approach is also evidenced by its ability to reject the multiple author hypothesis when there is no chrono-divide, as we have done for the other three classical Chinese novels.

Chapter 4

Open Class Authorship Identification

4.1 Introduction

Who wrote the *Fighting for Our Lives*, who wrote the *Recipes for Disater* and who wrote *The Animated Skeleton*? There are numerous books that were published anonymously or under pseudonyms. Although authorship attribution has a long history, most of the studies are confined to the close authorship identification cases where the authors are limited to a small set of potential candidate authors. For example, the *Federalist* papers were known to be written by three authors (Hamilton, Madison and Jay), and it is only a matter of deciding which paper was written by which one of the three authors. In contrast, the aforementioned books didn't have any reliable set of "suspects", and the identification of their authorship becomes an open case. The close case identification problem is generally far less challenging. With a small set of potential authors it is often viable to simply look for the closest match in some features, and this is in fact how most of these close cases were studied. However, this approach cannot be applied if the number of candidate authors is very large (in some cases it will be the entire database). Thus for open case identification problems we will need completely new approaches, approaches that will allow us to reliably pick out the true author among a large number of authors. So far, there has not been a reliable method to handle open class authorship identifications.

Authorship attribution is one of the most prominent problems in the broad research area of literary stylometry, which has seen rather explosive growth in recent years. The basic idea behind the authorship attribution is that people can always find some measurements that can distinguish the styles of different authors. Although the style of an author in different books may vary due to different genres, it shouldn't differ too much. The difference in style between authors should be much larger compared to that within an author if we use proper measurements. Thus, with these measurements, to find the author of a book whose set of candidate authors is known, we can use some books from a candidate author as training samples and build classifiers between different authors. The purpose of first part of this chapter is to solve close class problems, where the set of candidate author is small. In another word, if one book is suspected to be written either by author A_1, A_2, or A_3, we can always find some features that can discriminate the style of these authors and attribute it to one of the candidate authors. This forms the basis for close case authorship identification.

One of the fundamental requirement for conducting open case identifications is the access to a large database of authors and their work. Fortunately, there are many databases that are available either in the public domain (e.g. Project Gutenberg) or through library subscription (e.g. the Hathi Trust collection). But even with a large database, finding the author of a books such as *Fighting for Our Lives* when we have no clue about its authorship is challenging. The authorship in an open class problem is much harder to study, and very few work has been done so far. Harder still is to be able to do it efficiently within a large database. In this chapter, we propose a new method to open class problems in authorship attribution using the idea of randomization together with the method for close class problems, by which we can detect the true author of any text in a large database of authors and do it efficiently. We can attribute the text to its true author if he or she is in the database, and otherwise we can conclude that the true author is not in the database. We will describe and demonstrate the reliability of our algorithm in the next sections. Our method is based on a new method for closed case identification and an author randomization technique. In the final part of the chapter we conduct three case studies by analyzing three books: *Cuckoo's Calling*, *To Kill a Mockingbird* and *Dreams from My Father*. All three books are involved in controversies, some of which are still ongoing. We show that our method can almost unequivocally settle the controversies.

4.2 Close Class Problems

4.2.1 Methodology

As we mentioned earlier, the objective of close case identification is to construct classifiers for these authors and find the best match among the group using certain metrics. Here we build SVM classifiers using frequencies of words as features. Instead of using only function words, we use the most commonly appeared words in the samples we are analyzing. This turns out to be more effective in our testing, and it is also pointed out in [7].

Let $A_1, A_2, \ldots, A_L$ be the authors in a group within which we would like to find the closest match for certain text in question. We break down all the training and testing text into *samples*. Each sample is a segment of text consisting of K words. In our study we have set $K = 2000$. This proves to work very well and smaller K such as $K = 1000$ still works well. This is important in situations where the availability of text by certain author is limited. A typical book is broken down into between 25 to 100 samples. The number of samples that *can be attributed definitively to a given author* thus depends on how prolific the author is. On average an author will have a corpus of between two hundreds to a thousand samples.

For $L > 2$, there are two popular ways we can perform this multi-class classification task. One way would be to build classifiers for each pair, and in essence conduct a series of head-to-head contest. The closest match would be the author with the most wins. One potential problem is that as L increases, the number of pairwise classifiers would increase in the order of L^2, making it computationally less efficient. Here we employ the *one-vs-rest* method, in which we build L classifiers $f_1, f_2, \ldots, f_L$. The classifier f_j is built using author A_j as one class and all other authors grouped into a single class. This is shown to perform as well as the pairwise approach and in our testing, it actually outperforms slightly in our setting.

Assume that the text in question is broken down into N samples $X_1, X_2, \ldots, X_N$. We now apply the classifiers f_j to each sample X_n, where f_j is the sign function of decision function 2.11. The test is being matched with author j_0 (maximum win)if

$$j_0 = \mathrm{argmax}_j \sum_n f_j(X_n).$$

In our implementation we also gain some improvement by running several folds of classifiers and take their average to minimize the chance for outliers, with matching being decided by a voting scheme such as maximum win or BORDA. We shall call q_j/N the *matching rate* for author A_j, where q_j is the number of samples among $\{X_n\}$ that have matched with author A_j. The author with the highest matching rate is declared to be the closest match within the group.

4.2.2 Case studies

Case study 1 **– Cuckoo's Calling.** A controversy receiving a great deal of media attention in the summer of 2013 was the rumor alleging that the author Robert Galbraith of the crime fiction *Cuckoo's Calling* was actually the pseudonym for J. K. Rowling. In an effort to "out" Rowling, as a story in the National Geographic detailed, the *Sunday Times* sent the book along with books by Rowling and three other crime fiction authors to Patrick Juola and Peter Millican, two prominent experts in authorship attribution, for comparison. Both researchers were able to identify Rowling as the closest match among the 4 authors. But to highlight the general difficulty of true authorship identification, the story also mentioned that Juola "wasn't totally confident in the result. After all, he had no way of knowing whether the real author was somebody who wasn't in the comparison set of books who happened to write like Rowling does."

We have tested *Cuckoo's Calling* within a group of L candidates, where L can be 2, 3 or more and one or two books from each author are used in constructing the classifiers. To make our result comparable, the books we choose are similar to the ones that Juola and Millican had used in their training process. To show the accuracy of our classifiers, we have also tested other books by all the authors. The matching rates are listed together with that of *Cuckoo's Calling.* We have conducted several groups of experiments and obtained the following results (see tables (4.1 4.2 4.3 4.4)). As one can see, although there are some fluctuations when different training samples are used, the tested books from each author matched their own work with very high matching rates. Furthermore, all the tables here indicate that *Cuckoo's Calling* is the closest match to the work of J. K. Rowing. We can easily draw the same conclusion as Juola and Millican did, namely if Robert Galbraith is among the authors in the training list, then it is the pseudonym for J. K. Rowling.

Testing \ Training	The Casual Vacancy-J.K. Rowling	The Private Patient-P.D.James
Cuckoo's Calling	72/72	0/72
Deathly-JKR	92/100	8/100
Death-PDJ	1/45	44/45

Table 4.1: *Cuckoo's Calling*: Classification result by the classifier obtained using one book for each of the two authors as training samples.

Testing \ Training	The Casual Vacancy J.K.Rowling	The Private Patient P.D.James	No More Dying R.Rendell	Dead Beat V.McDermid
Cuckoo's Calling	65/72	0/72	1/72	6/72
Deathly-JKR	64/100	1/100	8/100	27/100
Death-PDJ	0/45	44/45	0/45	0/45
Some Lie-RR	2/28	0/28	25/28	1/28
Kick Back-VM	0/37	0/37	0/37	37/37

Table 4.2: *Cuckoo's Calling*: Classification result by the classifier obtained using one book for each of the four authors as training samples, group I.

Testing \ Training	Half-Blood J.K.Rowling	Cold Blood Capote Truman	Unbearable Lightness Alexander.McCall	Murder of Roger Agatha Christie
Cuckoo's Calling	57/72	11/72	3/72	1/72
Deathly-JKR	92/100	8/100	0/100	0/100
Prayer-CT	0/22	22/22	0/22	0/22
Lady-AM	1/32	0/32	31/32	0/32
Appoint-AC	5/26	0/26	4/26	17/26

Table 4.3: *Cuckoo's Calling*: Classification result by the classifier obtained using one book for each of the four authors as training samples, group II.

Case study 2 **– To Kill a Mockingbird** *To Kill a Mockingbird* was a southern drama by Harper Lee published in 1960. A year later it won the Pulitzer Prize and had quickly become one of the classics in American literature. Ever since its first publication, the authorship of the book has been a subject of controversy because this was the one and only one published work by Harper Lee until 2013. Some found it hard to believe that an unknown author would write a single and great novel and then stop writing. Rumors persisted the true authors was Truman Capote, a childhood friend of Lee and the author of

Testing \ Training	J.K.Rowling	P.D. James	Ruth Rendell	Val MacDermid
Cuckoo's Calling	72/72	0/72	0/72	0/72
Chamb-JKR	42/42	0/42	0/42	0/42
Light-PDJ	1/60	59/60	0/60	0/60
Crack-VM	0/37	0/37	1/37	36/37

Table 4.4: *Cuckoo's Calling*: Classification result by the classifier obtained using two books for each of the four authors as training samples.

In Cold Blood, despite the denials of both Lee and Capote. Pearl Belle, a famous literary critic and editor in Cambridge, Massachusetts, exposed that Capote implied to her that he penned or heavily edited the book. In 2001 Jim Gilbert, an Alabama writer, added to the rumor by claiming that *To Kill a Mockingbird* was the work of Capote after he compared it with *In Cold Blood* in terms of literary structure. But Harper Lee has her own defenders as well. Dr. Wayne Flyt, a retired professor of history from Auburn University, claimed that Harper Lee is the true author of this book after analyzing the voices of the characters and found the styles to be quite different from the style of Capote.

To solve this authorship mystery, we have compared *To Kill A Mockingbird* with the work by Truman Capote using the proposed method and author randomization for close class problems. We have trained the classifier using one book by each of the five candidate authors (see table 4.5). We have then tested *To Kill A Mockingbird* together with the other books by the selected candidate authors, the matching rates are shown in table 4.5. The results show that the testing works from candidate authors matched correctly in high rates, while the matching rate of *To Kill A Mockingbird* is distributed almost randomly in the three of the five authors. Thus, we conclude that it is unlikely that *To Kill A Mockingbird* was written by Capote Truman.

Case study 3 – **Dreams From My Father.** *Dreams from My Father* is a memoir by Barack Obama published in 1995. In 2008 William Ayers stirred up a controversy by

	A Darker Domain	The Executioners Song	In Cold Blood	The Murder of Frorence	Midnight Garden
Testing \ Training	Val McDermid	Mailer Norman	Capote Truman	Douglas Preston	John Berendt
To Kill a Mockingbird	13/49	3/49	15/49	1/49	17/49
Bleeding-VM	63/66	2/66	0/66	0/66	1/66
Castle-MN	9/75	53/75	3/75	10/75	0/75
Voices-CT	1/27	0/27	24/27	0/37	2/27
Prayer-CT	1/22	0/22	21/22	0/22	0/22

Table 4.5: *To Kill A Mockingbird*: Classification result by the classifier obtained using one book for each of the five authors as training samples.

"confessing" that he was the true author of Obama's memoir. Some of Obama's critics were eager to point out that Obama didn't possess the writing skill to write this best-selling book because he couldn't even write a 30 second speech (his 2012 acceptance speech was written by Bill Clinton). His defenders, on the other hand, were firm believers that Obama himself was the true author. After reading the book, Jack Cashill argued that *Dreams from My Father* was thematically and semantically too close to Ayers's earlier memoir, *Fugitive Days*. Patrick Juola couldn't make the conclusion and said: "The accuracy simply isn't there" using his tools.

To investigate the Ayers' "confession", we constructed classifiers within a small group, which included the book *Fugitive Days*, *Audacity of Hope* by Obama, and other two books from other authors, and tested the matching for *Dreams From My Father* and other books known to be by the chosen authors. As we can see in table 4.6, we matched all the books to the true authors with high accuracy, while the best match for *Dreams From My Father* was with Obama's *Audacity of Hope*. This result leads us to conclude that Ayers likely lied in his "confession".

	Audacity of Hope	Fugitive Days	A Distant Mirror	The Story of My Life
Testing \ Training	Barack Obama	Bill Ayers	Barbara Tuchman	Clarence Darrow
Dreams-Obama	54/74	10/74	3/74	7/74
Confession-Ayers	1/42	40/42	0/42	1/42
Folly-Tuchman	2/80	0/80	78/80	0/80

Table 4.6: *Dreams From My Father*: Classification result by the classifier obtained using one book for each of the four authors as training samples.

4.3 Open Class Problems

Open class problems are to check whether the author is in the training author pool or not, if yes, who it is. Large candidate group and efficient algorithm to deal with big data are the two main challenges in solving open problems. We will move one step further in authorship attribution and reliably detect whether the author is in the training author pool or not. Our approach is based on the method in solving close class problem and the idea of randomization. The idea is that if the author is in a small group, we will always get high matching rate for the true author when we construct classifier in the group. When we separate candidate authors into small groups randomly, the true author would always obtain high matching rate. By recording the occurrence of the authors who get the highest matching rate, we would obtain the high frequency of the true author. If we fix author A and construct classifier by randomly choosing other author in the training many times, the average matching rate for author A should be much higher if A is the true author. If the average matching rates for all the candidate authors are not high enough, then we can conclude that the true author is not in the candidate set.

The open class authorship identification problem is divided into several components. Here we break down these components and discuss their details.

4.3.1 Database and data preparation

Naturally, to perform open class authorship identification, one would need a sizable digital database of authors. Fortunately this is not a problem today. The publicly available Project Gutenberg has over 45,000 titles as of May 2014. More impressive is the Hathi Trust collection, which has over 10 million titles and is available from several university libraries.

The Hathi trust contains both public domain and copyrighted titles. It is an ideal database around which to build a large scale authorship stylometry analysis project. For author in the training, we need enough samples to train the classifiers. thus, We don't use all the authors in the mentioned database. We just choose some authors who have at least 3 books and more than 100 samples (of length 2000) from these books. Finally, we got a data base which contains 200 authors and each author has at least 3 books which contain at least 100 samples.

Since our method is based on supervised machine learning, training samples in the database will be used to train for classifiers. In theory we can use all available samples in the training process. In practice, we noticed that results do not improve significantly with more than 100 samples per author. In fact we obtain good results even with only 30-60 training samples per author. By randomly fixing one book as test and constructing the classifier between the true author and others, we obtain the average matching rate for the true author by repeating this process 50 times. Theoretically, the writing style of the authors can be better reflected if the training samples are chosen from different books by each author rather than from one or two whole books by the authors. The following two tables 4.7 and 4.8 show that when the sample size increase from 30 to 90, the average true matching rate is getting higher. They also verified that when the training samples are from different books rather than one or two books by the author, the true matching rate is better. Thus, all our case studies in this part have used 90 training samples from different books by each author.

4.3.2 Methodology

Improved Reliability and Robustness through Author Randomization. Good results can often be attained if there is certainty that the author of the text in question is

	L=2	L=3	L=4	L=5
30	0.93	0.87	0.86	0.84
60	0.94	0.93	0.89	0.88
90	0.94	0.93	0.90	0.90

Table 4.7: Matching rate for multi-class classifications trained by randomly chosen books by each author.

	L=2	L=3	L=4	L=5
30	0.95	0.92	0.91	0.88
60	0.96	0.95	0.93	0.92
90	0.97	0.96	0.94	0.93

Table 4.8: Matching rate for multi-class classifications trained by randomly chosen samples by each author.

indeed in the group (such as in the case of *Federalist* papers). The problem is that the closest match can always be found even when the real author of the text in question is in fact not a member of the group. So in general one often can only claim to have found the closest match within the group without truly identifying the authorship.

Our SVM approach isn't immune to this concern. In fact, it can happen when in a small group a matching rate of over 90% is obtained by someone other than the true author of some text in question. This usually occurs when the true author is not in the group. So how can one effectively minimize such false matching?

A key ingredient for our method is *author randomization*, which proves to be a highly effective way to identify a false matching. Here for each *potential candidate* author we perform a series of small group classifications against randomly chosen authors from a database. More precisely, let A_1 be the author we wish to test for matching for the text samples $X_1, X_2, \ldots, X_N$. We now randomly select authors $A_2, \ldots, A_L$ for a database and perform the classification. This is done over several trials. In the end we obtain an average matching rate for author A_1. The outline of our algorithm for open class problems is as following:

1 Initialize the data, which contains at least 3 books by each author in the database.

2 Split each book into parts of K words (usually we set $K = 2000$), and extract the features for each part. We randomly choose one book by each author in the database as test.

3 Randomly divide the authors into groups of L_1 authors, and construct classifier for each group. Record the authors who receives the highest matching rates.

4 Repeat T_1 times step 3 and record the authors having the highest matching rates.

5 Remove the authors whose average matching rate is below the threshold S_1. Those authors who are still left are deemed as potential true authors, or "suspect authors".

6 Fix each suspect author from step 5 and randomly choose $L - 1$ authors from the database and construct a classifier. Record the matching rates for the suspect authors.

7 Repeat T_2 times step 6 to obtain the average matching rate for each suspect author.

8 If the maximum average matching rate from step 7 is above some threshold S_2, we identify that author as the true author. Otherwise the true author is not in the dataset.

We have done extensive test of this approach, and the table 4.9 shows the matching rates for both authors and non-authors. The last row denotes the maximal average matching rate for a non-author in our test. As one can see, the average matching rates of the true authors are above 90% even for 5-class classification, while the average matching rate of the non-author is always on average. What's more, even the maximal average matching rate for a non-author is not even close to the average matching rate for the true author. This is perhaps the most important attribute that allows us to identify authorship using our approach.

	L=2	L=3	L=4	L=5
Author	0.94	0.93	0.90	0.90
Non-author	0.48	0.32	0.24	0.20
Max non-author	0.70	0.62	0.59	0.51

Table 4.9: Matching rate for multi-class classifications by randomly chosen books by each author.

	L=2	L=3	L=4	L=5
Author	0.97	0.96	0.94	0.93
Non-author	0.51	0.34	0.25	0.20
Max non-author	0.66	0.59	0.52	0.49

Table 4.10: Matching rate for multi-class classifications with randomly chosen samples by each author.

The importance of the above results is that it provides an extremely robust way to not only tell us which author is the best match for a given body of text, but also how well the matching is. The gap between matching rate for true authors and non-authors is so significant that it can be scaled up into a legitimate open class authorship identification method, which we discuss next.

Open Class Authorship Identification. To scale our method for close class problems up for open class authorship identification, we divide our database of authors into small groups $G_1, G_2, \ldots, G_M$ of L authors (we choose L to be 4 or 5. It works for L as large as 10). Within each group G_m we build classifiers, which can have several folds as mentioned earlier. For more robustness we may also establish several such groupings from random assignments. Note that this is perhaps the most computationally demanding part with a large database, but it can be done entirely offline as a onetime task.

Given a body of text in question and its samples $X_1, \ldots, X_N$, we feed the samples into the classifiers for each group. Each group will have an author that has the highest matching rate, but even the highest matching rate can be low. We only select those whose matching rates exceed a threshold (e.g. 50%) as candidates for the next round of testing. In theory, the true author should always get the highest matching rate in all groups, while other authors won't always obtain high matching rate. Assume that after this initial round we are left with a group of potential authors $B_1, B_2, \ldots, B_q$ for the text in question. If q is still rather large, as it is possible given the size of the database we hope to build, we would then treat $B_1, B_2, \ldots, B_q$ as a separate database of authors and iterate this process until a small number of candidate authors $C_1, \ldots, C_p$ remain.

Now here the author randomization method shows its power. For each remaining candidate author C_j, we randomly add $L - 1$ authors from the entire database to form a group G of L authors. Within G a matching rate for C_j is obtained. We repeat this process for C_j many times to obtain an average matching rate for C_j. We identify the author among C_j with the highest average matching rate as the author of the text in question, provided that the average matching rate is higher than certain threshold. If even the highest average matching rate is low, we may conclude that the text is written by an author outside our database.

In the following case studies we set $L_1 = 4$ or 5, $T_1 = T_2 = 50$, $S_1 = 0.5$ or 0.6. We also set $S_2 = 0.8$.

4.3.3 Case studies

We have built a test database consisting of 200 authors and each author has more than 100 samples. This database, although small, already allows us to test our method and study some of the real world controversies. We are working with the Michigan State Library to significantly augment the size of the database. It is conceivable that using the Hathi trust we can build a database consisting of several hundreds of thousands of authors.

Case study: **Cuckoo's Calling.** As a test case we ran our open class method on *Cuckoo's Calling* using the database we had built. The result is quite strikingly even though our database is small. In the first stage where we divide the authors into groups of length 5, so there are now 40 groups. We record the author in each group with the highest matching rate. But of the 40 leaders in each group only 7 has a matching rate over 60%. If we lower the threshold to 50% then there are 16 candidates remaining. However, if we do this 50 times then only 3 candidates have matching rate of 50% or higher in at least 30 trials (not surprisingly Rowling passed the threshold in all trials.). We also tried a more robust approach in the first stage, in which we group the authors and keep those as potential authors (or suspect authors). The next stage is author randomization, where we fix each suspect author and randomly choose $L-1$ authors from the non-suspect authors and build the classifiers. We obtained the average matching rate for each classifier by repeating T_2 times in this randomization process ($T_2 = 50$). Table 4.11 shows that the matching rate for each suspect author is higher than the average. Still the matching rate in $T_2 = 50$ repeats for Rowling was much higher and we can easily and almost definitively identify Rowling as the true author.

master authors	frequency	L=2	L=3	L=4	L=5
80	50/50	0.96	0.93	0.91	0.91
163	34/50	0.64	0.58	0.50	0.41
46	33/50	0.65	0.57	0.44	0.33
83	28/50	0.63	0.56	0.42	0.31
174	28/50	0.62	0.53	0.40	0.32

Table 4.11: *Cuckoo's Calling:* master suspected authors and the average matching rate.

***Case study 2* – To Kill a Mockingbird.** As another case study, we have applied our method of author randomization to study the authorship of *To Kill A Mockingbird*. We test the book for matching against Capote and $L - 1$ other randomly selected authors from our database. Additionally we have also tested *Answered Prayers* by Capote in parallel. The trial is repeated 50 times. As we can see in table 4.12, the average matching rate of *To Kill a Mockingbird* with Capote is lower than 50%. In contrast, the matching rate for *Answered Prayers* is always above 80%. The result shows virtually irrefutably that Capote did not write *To Kill a Mockingbird*.

test books \ matching rate	L=2	L=3	L=4	L=5
To Kill a Mockingbird	0.49	0.40	0.39	0.36
Voices-CT	0.89	0.84	0.85	0.81
Prayer-CT	0.98	0.96	0.95	0.96

Table 4.12: *To Kill a Mockingbird:* the average matching rate.

***Case study 3* – Dreams From My Father.** To strengthen our conclusion, we have again applied our method by matching up *Dreams From My Father* against William Ayers and $L - 1$ randomly selected authors. This is repeated 50 times. Testing results in table 4.13 are the average matching rate for the 50 trials. The average matching rate for Ayers is lower than 25%. In contrast in the same test but with *Public Enemy: Confessions of An American Dissident* by Ayers in place of *Dreams from my Father*, we are able to obtain over 98% average matching rate! Constructing the classifiers using other books by Ayers yield very similarly high matching rates, which are always above 95%. Changing the training book for Ayers gives almost the same results, where the matching rate for *Dreams from My father* against Ayers has always stayed below 50%. This is virtually irrefutable evidence that Ayers *lied* in his “confession” about writing Obama’s autobiography.

As we can see in the results of the three experiments, our method for open class problems

test books \ matching rate	L=2	L=3	L=4	L=5
Dreams-Obama	0.24	0.25	0.23	0.23
Confession-Ayers	1.0	0.98	0.99	0.99

Table 4.13: *Dreams From My Father:* the average matching rate.

can always detect the true author successfully, although the matching rates of the true authors vary in different experiment. When the true author is not in the candidate set like *To Kill a Mockingbird*, the results from our method can lead us claim that the true author is not in the candidate set.

4.4 Conclusion

We studied the close class authorship attribution problems using machine learning techniques and studied the authorship problems of *Cuckoo's Calling*, *To Kill a Mockingbird* and *Dreams From My Father*. All the experiments showed high matching rate for the true authors.

We proposed a new algorithm to solve open class authorship attribution problems by extending the method for close class problems and randomizing authors in a big author pool. By comparing the matching rate of true authors and non-authors, we showed that this algorithm is very reliable and stable. To further verify our result, we studied the three particular cases as in the close case.

Applying our methods to *Cuckoo's Calling*, we can confirm that it is written by J.K.Rowling and published with his pseudonym Robert Galbraith with strong evidence. Studying the author of *To Kill a Mockingbird* with our methods, we concluded that its author is not Capote Truman, although we can't conclude it is by Harper Lee because no training samples from her. Furthermore, we can tell that Ayers lied to the media that it was he who wrote *Dreams From My Father* by comparing the matching rates between books by Obama and Ayers.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Friederike Antosch. *The diagnosis of literary style with the verb-adjective ratio.* New York, 1969.

[2] Yan Anzheng. One piece of evidence that chapters 64 and 67 are not the original version. *Journal of Xianyang Normal University*, 24:3, 2009.

[3] Ronald E Bee. Statistical methods in the study of the masoretic text of the old testament. *Journal of the Royal Statistical Society. Series A (General)*, pages 611–622, 1971.

[4] Ronald E Bee. A statistical study of the sinai pericope. *Journal of the Royal Statistical Society. Series A (General)*, pages 406–421, 1972.

[5] Barron Brainerd. On the distinction between a novel and a romance: a discriminant analysis. *Computers and the Humanities*, 7(5):259–270, 1973.

[6] Barron Brainerd. Weighing evidence in language and literature: A statistical approach. *AMC*, 10:12, 1974.

[7] John F Burrows. Word-patterns and story-shapes: The statistical analysis of narrative style. *Literary and linguistic Computing*, 2(2):61–70, 1987.

[8] John F Burrows. an ocean where each kind...: Statistical analysis and some major determinants of literary style. *Computers and the Humanities*, 23(4-5):309–321, 1989.

[9] B.C. Chan. Authorship of the dream of the red chamber: A computerized statistical study of its vocabulary. *Dissertation Abstracts International Part A: Humanities and[DISS. ABST. INT. PT. A- HUM. & SOC. SCI.]*, 42(2):1981, 1981.

[10] Koby Crammer and Yoram Singer. On the learnability and design of output codes for multiclass problems. In *Proceedings of the Thirteenth Annual Conference on Computational Learning Theory*, COLT '00, pages 35–46, San Francisco, CA, USA, 2000. Morgan Kaufmann Publishers Inc.

[11] A. de Morgan. Letter to rev. heald 18/08/1851. In S. Elizabeth and D. Morgan, editors, *Memoirs of Augustus de Morgan by his wife Sophia Elizabeth de Morgan with Selections from his Letters.* London: Longman's Green and Co, 1851/1882.

[12] Joachim Diederich, Jörg Kindermann, Edda Leopold, and Gerhard Paass. Authorship attribution with support vector machines. *Applied intelligence*, 19(1-2):109–123, 2003.

[13] Thomas G. Dietterich and Ghulum Bakiri. Solving multiclass learning problems via error-correcting output codes. *arXiv preprint cs/9501101*, 1995.

[14] Alvar Ellegård. *A Statistical method for determining authorship: the Junius Letters, 1769-1772*, volume 13. Göteborg: Acta Universitatis Gothoburgensis, 1962.

[15] Jillian M Farringdon, Andrew Queen Morton, Michael G Farringdon, and M David Baker. *Analysing for Authorship: A Guide to the Cusum Technique.* University of Wales Press Cardiff, 1996.

[16] Li Guoqiang and Li Rui-fang. Study Based on Statistics of word Frequency Research on Only Author of the" Dream of the Red Chamber"[J]. *Journal of Shenyang Institute of Chemical Technology*, 4, 2006.

[17] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1):389–422, 2002.

[18] D.I. Holmes and J. Kardos. Who was the author? an introduction to stylometry. *CHANCE-BERLIN THEN NEW YORK-*, 16(2):5–8, 2003.

[19] Chih-Wei Hsu and Chih-Jen Lin. A comparison of methods for multiclass support vector machines. *Neural Networks, IEEE Transactions on*, 13(2):415–425, 2002.

[20] Matthew L Jockers and Daniela M Witten. A comparative study of machine learning methods for authorship attribution. *Literary and Linguistic Computing*, page fqq001, 2010.

[21] Patrick Juola. Authorship attribution. *Foundations and Trends in information Retrieval*, 1(3):233–334, 2006.

[22] Vlado Kešelj, Fuchun Peng, Nick Cercone, and Calvin Thomas. N-gram-based author profiles for authorship attribution. In *Proceedings of the conference pacific association for computational linguistics, PACLING*, volume 3, pages 255–264, 2003.

[23] Bradley Kjell. Authorship attribution of text samples using neural networks and bayesian classifiers. In *Systems, Man, and Cybernetics, 1994. Humans, Information and Technology., 1994 IEEE International Conference on*, volume 2, pages 1660–1664. IEEE, 1994.

[24] Bradley Kjell. Authorship determination using letter pair frequency features with neural network classifiers. *Literary and Linguistic Computing*, 9(2):119–124, 1994.

[25] Bradley Kjell, W Addison Woods, and Ophir Frieder. Information retrieval using letter tuples with neural network and nearest neighbor classifiers. In *Systems, Man and Cybernetics, 1995. Intelligent Systems for the 21st Century., IEEE International Conference on*, volume 2, pages 1222–1226. IEEE, 1995.

[26] Moshe Koppel, Shlomo Argamon, and Anat Rachel Shimoni. Automatically categorizing written texts by author gender. *Literary and Linguistic Computing*, 17(4):401–412, 2002.

[27] Andrew Queen Morton. *Literary detection: How to prove authorship and fraud in literature and documents*. Bowker London, 1978.

[28] Frederick Mosteller and David Wallace. Inference and disputed authorship: The federalist. 1964.

[29] A. Pawlowski. Wincenty lutoslawski-a forgotten father of stylometry. *Glottometrics*, 8:83–89, 2004.

[30] Cao Qingfu. The last 40 chapters of dream of the red chamber were not written by cao xueqin comparison of function words, phrases and chapter titles in the first 80 chapters and the last 40 chapters. *A Dream of Red Mansions*, 01:288–319, 1985.

[31] Yu Qingxiang. Application of statistics in dream of the red chamber. *Journal of University of Politics*, 76:303–327, 1998.

[32] J. Rudman. The state of authorship attribution studies: Some problems and solutions. *Computers and the Humanities*, 31(4):351–365, 1997.

[33] Efstathios Stamatatos. A survey of modern authorship attribution methods. *J. Am. Soc. Inf. Sci. Technol.*, 60(3):538–556, March 2009.

[34] Fumitake Takahashi and Shigeo Abe. Optimizing directed acyclic graph support vector machines. *Artificial Neural Networks in Pattern Recognition (ANNPR)*, pages 166–173, 2003.

[35] V Vapnik. Estimation of dependencies based on empirical data. 1979.

[36] Vladimir Naumovich Vapnik and Vlamimir Vapnik. *Statistical learning theory*, volume 2. Wiley New York, 1998.

[37] Zhang Wei-dong and Liu Li-chuan. Investigation of the linguistic styles of the first 80 chapters and the last 40 chapters of dream of the red chamber. *Journal of Shenzhen University*, 1, 1986.

[38] Jason Weston, Chris Watkins, et al. Support vector machines for multi-class pattern recognition. In *ESANN*, volume 99, pages 219–224, 1999.

[39] Hu Xiangfeng, Wang Yang, and Wu Qiang. Multiple authors detection: a quantitative analysis of *Dream of the Red Chamber*. *Advances in Adaptive Data Analysis*, 6(4):Article ID 1450012, 18 pages, 2014.

[40] Li Xianping. A new hypothesis on the writing and publication of dream of the red chamber. *Fudan Journal of Social Science Edition*, 5, 1987.

[41] G Udny Yule. On sentence-length as a statistical characteristic of style in prose: With application to two cases of disputed authorship. *Biometrika*, pages 363–390, 1939.