Author Identification from Song Lyrics

Nazmun Nisat Ontika
Student ID: 012151006

A Thesis

in

The Department

of

Computer Science and Engineering

Presented in Partial Fulfillment of the Requirements

For the Degree of Master of Science in Computer Science and Engineering

United International University

Dhaka, Bangladesh

November, 2018

Approval Certificate

This thesis titled " **Author Identification from Song Lyrics**" submitted by Nazmun Nisat Ontika, Student ID: 012151006, has been accepted as Satisfactory in fulfillment of the requirement for the degree of Master of Science in Computer Science and Engineering on November, 2018.

Board of Examiners

1.

_____ Supervisor

Dr. Mohammad Nurul Huda
Professor & MSCSE Director
Department of Computer Science & Engineering (CSE)
United International University (UIU)
Dhaka, Bangladesh

2.

_____ Head Examiner

Dr. Khondaker Abdullah-Al-Mamun
Professor
Department of Computer Science & Engineering (CSE)
United International University (UIU)
Dhaka, Bangladesh

3.

_____ Examiner-I

Dr. Swakkhar Shatabda
Associate Professor
Department of Computer Science & Engineering (CSE)
United International University (UIU)
Dhaka, Bangladesh

4.

_____ Examiner-II

Dr. Dewan Md. Farid
Associate Professor
Department of Computer Science & Engineering (CSE)
United International University (UIU)
Dhaka, Bangladesh

5.

_____ Ex-Officio

Dr. Md. Abul Kashem Mia
Dean of School of Science & Engineering
United International University (UIU)
Dhaka, Bangladesh

Declaration

This is to certify that the work entitled "**Author Identification from Song Lyrics** " is the outcome of the research carried out by me under the supervision of Dr. Mohammad Nurul Huda, Professor and MSCSE Director, United International University (UIU), Dhaka, Bangladesh.

_____

Nazmun Nisat Ontika
Student ID: 012151006
MSCSE Program
Department of Computer Science and Engineering
United International University (UIU)
Dhaka, Bangladesh

In my capacity as supervisor of the candidate's thesis, I certify that the above statements are true to the best of my knowledge.

_____

Dr. Mohammad Nurul Huda
Professor and MSCSE Director
Department of Computer Science and Engineering
United International University (UIU)
Dhaka, Bangladesh

Abstract

Machine Learning (ML) tools have been used extensively in a wide variety of domains recently. Due the enormous amount of data being produced, machine learning techniques are being heavily used to make sense of data & derive meaningful results. Using machine learning tools, we can turn the data into knowledge.

Music is one of the truest forms of art. Bangladesh has a great history of music with a great tradition of song writing over centuries. Authorship attribution is the way of identifying the author from a linguistic corpus.

This paper demonstrates a guideline to identify the author of a Bengali song from the lyrics of that song using machine learning. This research work presents the first work on machine learning approach for author attribution from the lyrics of a song. Here six methods of machine learning are used for the author identification and high accuracies have been achieved from these methods. It is observed that Naïve Bayes method provides higher accuracy in comparison with the other methods.

Acknowledgement

I would like to start by expressing my deepest gratitude to the Almighty Allah for giving me the ability and the strength to finish the task successfully within the scheduled time.

This thesis titled “Author Identification from Song Lyrics” has been prepared to fulfill the requirement of MSCSE degree. I am very much fortunate that I have received sincere guidance, supervision and co-operation from various persons.

I would like to express my heartiest gratitude to my supervisor, Dr. Mohammad Nurul Huda for his continuous guidance, encouragement, and patience, and for giving me the opportunity to do this work. His valuable suggestions and strict guidance made it possible to prepare a well-organized thesis report.

At long last, my most profound appreciation and love to my parents for their help, support, and interminable love.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

Chapter 1

Introduction

1.1 Overview

Machine Learning is a branch of Artificial Intelligence (AI), which is concerned about learning from data. In recent years, there are enormous amounts of data being generated in different fields, like science, engineering, medicine etc. Machine learning tools are being heavily used in diverse fields. Machine learning algorithms help to gain insight from data. With machine learning algorithms & techniques, we can better understand the data and turn those to knowledge.

The application of machine learning is notable in every field nowadays. From education to entertainment, architecture to art we are using machine learning approaches. Music is one of the truest forms of art and machine learning is contributing a lot in music nowadays. There are some works for genre classification, mood classification but song lyrics area was not focused till now. Bangladesh has a great history of music with a great tradition of songwriting over centuries. However, it is being observed that people are more concern about the artist than the lyricist of a song. Hence, proper information about the author is rarely available. If we could not devise an effective method to know the related information of a songwriter from a song, many songwriters name will be lost from the modern history of music.

Authorship attribution is the way of identifying the author from a linguistic corpus [1]. In machine learning, author identification is a very active and important field to resolve plagiarism, email spam, online fraud and so on. During the last decade, there have been many works done by the researchers in this field. However, most of the time English was selected as the language for the research. However, in every language author attribution should be given more importance to keep the originality of the work.

A rightful author can be identified using a machine learning approach. Everybody's writing and speaking style are different. We all use our own knowledge to refer words in sentences in every language. Each language has its own structure and even for the same language, we use different grammatical styles to represent our sentences. For example, in English, in some

region, they call a particular piece of furniture like a sofa where other people call it a couch. There are words that are different but have similar meanings like, “Although” and “Though”. Same words are also common with different spellings like, “Elite” / “L33t” / “1337”. Moreover, there are also many ways to express similar ideas. For instance, “The pen is to the right of the notebook” or “The pen is at the notebook’s right”. So, these differences are how we can differentiate the authors in writing because every author has his or her own unique style of writing.

For example, if one lyricist said,

“একটুকু ছোঁয়া লাগে, একটুকু কথা শুনি

তাই দিয়ে মনে মনে রচি মম ফাল্গুনী”

And another author said,

“শাওন রাতে যদি স্মরণে আসে মোরে

বাহিরে ঝড় বহে নয়নে বারি ঝরে।”

Then how can we know which lyrics belong to which author?

Since we are more concerned about the lyricist, we have to consider the lyrics as our main resources. A lyric is separated from an audio song but it contains important information about the song and the author too. This paper demonstrates a guideline to identify the author of a Bengali song from the lyrics of that song using machine learning. This research work presents the first work on machine learning approach for author attribution from the lyrics of a song.

1.2 Author Attribution Problems

For the Bengali language, this is the very first model for author identification from Bengali song lyrics. Some key facts are behind this circumstance. One of the main reasons is a limited number of data-sets or corpus to drive more research on Bengali song lyrics or Bengali lyricists. Cleaning and reprocessing are essential for the available corpus. Bengali word spelling is a bit tricky than English, hence authentication is very much important regarding Bengali song lyrics. Another thing is, text syntax is different in the poems or lyrics from the

text in essays or stories because sometimes for rhyming purpose authors use short forms, are synonyms, a lot of punctuations, different spellings or words from different languages which make the syntax more difficult. The works which have been discussed in the related work section, are the ideal example of overcoming these shortcomings in author attribution in Bengali literature.

1.3 Motivation

In this study, we have used system literature review process and followed standard steps for searching, screening, data-extraction, and reporting. First of all, we tried to search for relevant papers, presentations, research reports and policy documents that were broadly concerned with author attribution. We identified appropriate electronic databases and websites. Conceivably important articles were recognized utilizing the electronic databases and websites, for example, IEEE Explore, ACM Portal, Springer Linker, Science Direct, and Google Search Engine. For the best and a consistent search, we applied a systematic search strategy where we derived keywords, queries, and phrases from the desired research question. After that, keywords were arranged into categories and we arranged the related keywords too. We have also used some facilities of the digital libraries, for example, sort by year etc. The search keywords were refined to include only those words which have produced successful results in our lab experiment.

Data collection was a great challenge for us. The Internet provides a great deal to find lyrics online. However, there is no reliable large data set for Bengali lyrics with proper author information. Although we have selected some great Bengali songwriters for experiments who have thousands of songs of their own but very few of those are available on the Internet. There are some websites for Bengali song lyrics as well as some mobile applications but some of those provide only the lyrics, without the related information about those songs, even without the name of the authors. Despite having difficulties to find resources, we have gathered the lyrics from online for each of the seven lyricists, that we mentioned and described later in section 3.2 and stored the lyrics of each song into a text file format and used that in our experiment.

1.4 Related Works

Author attribution has become a major research area from the 19th century. There are many great previous works on music, for example, mood detection, genre classification and author attribution [2], and some works are based on lyrics [3] [4] but not on lyricists. However, in the Bengali language, this area is a relatively new concern. There are not many works have been done to find an author. We found two major works in this field. One was done by Das and Mitra in 2011 where they experimented with 36 documents and tried to find author from three authors set [2]. Another work was done by Chakrabarty in 2012 where he considered 150 documents and tried to find author from three authors set too [3]. While Das and Mitra got 90% accuracy from their experiment by using the uni-gram method, Chakrabarty gained 84% accuracy by using Support Vector Machine (SVM). However, in both these experiments, they didn't have large data set to have any reliable conclusion. We found another work in Bengali literature where the authors had a large data set but focused only to identify from three authors and used passages as text [4]. Many researchers focused on author attribution works but they worked on to find artists, we found another work done by a student for his thesis titled as "Authorship Attribution of Song Lyrics" but because of having lack of information about the lyricists, he did not consider the original creator who wrote the lyrics of a song as the author of that song rather he considered the artist who played out that song as the creator of that song [5]. This is what we actually wanted to address, the lack of information about the authors or the lyricist who wrote a certain poem or a song.

1.5 Objectives

Our research focused to address authorship attribution by machine learning approach where we have applied six different methods and followed certain steps to test our methodology in the lab environment.

This research is conducted with these following objectives –

- Create Bengali corpora from song lyrics which are in text format.
- Extract the appropriate features of the text documents.
- Reduce the feature vector dimensionality.
- Select the best feature for author attribution.

- Build a machine learning classifier that can accurately predict the lyricist or author of that song from the lyrics.

1.6 Thesis Outline

In this paper we have discussed the following – background material, where we have discussed different machine learning tools; experiment setup, where we have described our data sets, feature extraction, feature selection and the process of classifier design; experiment analysis, where we have explained our experiment result and done the analysis and evaluated the performance we have gained. Lastly, we have mentioned the limitations we have faced during the experiment and the works we want to follow in the future.

Chapter 2

Background Study

2.1 Introduction

In this chapter, we will discuss different supervised machine learning algorithms. As we will see, each algorithm has some advantages & disadvantages. We will try to have a general idea about the comparative picture of different algorithms.

2.2 Machine Learning Tools

Machine learning is a subclass of software engineering that best in class from the examination of model affirmation and computational learning theory in mechanized thinking. Machine learning explores the advancement and examination of estimations that can pick up from and make desires on data. Such figuring work by building a model from point of reference commitments to demand to settle on data driven conjectures or decisions, instead of clinging to altogether static program rules.

Machine learning undertakings are commonly ordered into a few general classes, contingent upon the idea of the learning "flag" or "input" accessible to a learning framework. These are:

Supervised Learning: Supervised machine learning is the undertaking of construing a capacity from marked preparing information. The preparation information comprises of a lot of preparing precedents. In this learning, every model is a couple comprising of an input object (regularly a vector) and an ideal output value (likewise called the supervisory signal). This algorithm examines the preparation information and produces an induced capacity, which can be utilized for mapping new precedents. An ideal situation will take into account the calculation to accurately decide the class marks for concealed occasions. The info esteem is called preparing information or training data and has a known name or result, for instance, spam/not-spam or a stock expense at some random minute. A model is set up through a readiness system where it is required to make desires and is reconsidered when those gauges aren't right. The readiness method continues until the moment that the model achieves a perfect element of exactness on the planning data.

Unsupervised Learning: Unsupervised machine learning is the errand of gathering a capacity to depict concealed structure from "unlabeled" information (an arrangement or order is excluded in the perceptions). Since the models given to the student are unlabeled, there is no target assessment of the exactness of the structure that is yield by the significant calculation which is one method for recognizing unsupervised gaining from managed learning and fortification learning. A focal instance of unsupervised learning is the issue of thickness estimation in measurements, however unsupervised learning includes numerous different issues (and arrangements) including condensing and clarifying key highlights of the information. Info information isn't checked and does not have a known result. A model is set up by finishing up structures present in the info data. Demonstrate issues are alliance rule learning and grouping. Demonstrate computations are the Apriori calculation and k-implies.

2.2.1 Naive Bayes Classifier

Naive Bayes classifiers are a gathering of direct probabilistic classifiers subject to applying Bayes' theory with strong (naive) opportunity suppositions between the features.

Naive Bayes classifiers are significantly adaptable, requiring different parameters direct in the quantity of elements (highlights/indicators) in a learning issue. Most extraordinary likelihood getting ready should be conceivable by evaluating a closed edge explanation, which takes direct time, instead of by expensive iterative estimation as used for some unique sorts of classifiers.

Naive Bayes is a fundamental methodology for building classifiers where models that consign class imprints to issue events, marked as vectors of highlight esteems, where the class names are drawn from some constrained set. It's definitely not a singular figuring for getting ready such classifiers, nonetheless, a gathering of computations reliant on a regular rule which is, all Naive Bayes classifiers acknowledge that the estimation of a particular component is self-ruling of the estimation of some other component, given the class variable. For example, a characteristic item may be seen as an apple in case it is red, round, and around 10 cm in separation over. A simple Bayes classifier considers all of these features to contribute unreservedly to the probability that this natural item is an apple, paying little notice to any possible connections between the shading, roundness, and separation crosswise over features.

For a couple of sorts of probability models, blameless Bayes classifiers can be arranged gainfully in an oversaw getting the hang of setting. In various sensible applications, parameter estimation for Naive Bayes models uses the strategy for most outrageous likelihood; toward the day's end, one can work with the artless Bayes exhibit without enduring Bayesian probability or using any Bayesian procedures.

In spite of their credulous structure and clearly distorted suspicions, innocent Bayes classifiers have worked great in numerous mind-boggling true circumstances. In 2004, an examination of the Bayesian portrayal issue showed that there are sound speculative clarifications behind the clearly unimaginable ampleness of Naive Bayes classifiers. In any case, an exhaustive correlation with other arrangement calculations in 2006 demonstrated that Bayes order is done by different methodologies, for example, helped trees or arbitrary timberlands.

One clarification for the shockingly great execution of Naive Bayes in numerous spaces does not require correct circulation for grouping, just the correct choice limits.

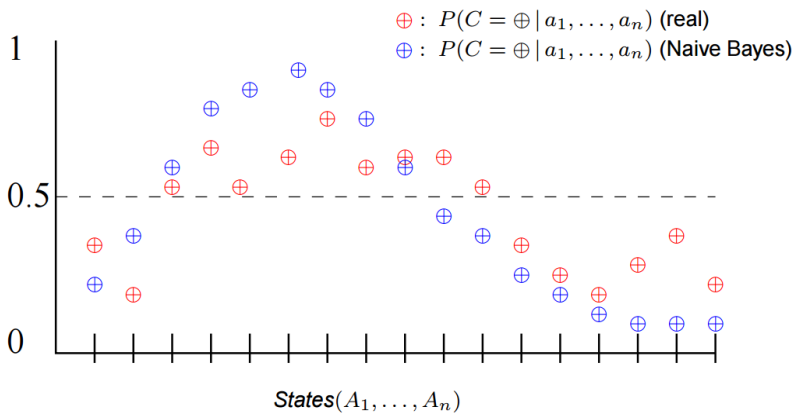


Figure 1: The Paradoxical Success Of Naive Bayes

In experiments, the autonomy supposition is regularly disregarded, yet Naive Bayes classifiers still will, in general, perform extremely well under this impossible suspicion. Particularly for little example sizes, guileless Bayes classifiers can beat the more amazing choices.

2.2.1.1 Advantages

- Easy to implement
- Requires a little measure of training data information to assess the parameters
- Good results obtained in most of the cases

2.2.1.2 Disadvantages

- Loss of accuracy due to conditional independence in the class
- Dependencies are found among the variables in practical cases
- Dependencies cannot be modeled by Naive Bayesian Classifier

2.2.1.3 Mathematical Formulation

Naive Bayes strategy is a set of supervised learning method dependent on applying Bayes' hypothesis with the naive supposition of autonomy between each pair of the features. Given a class variable y and a dependent feature vector x_1 through x_n, Bayes' hypothesis expresses the accompanying relationship:

$$P(y \mid x_1,\ldots,x_n) = \frac{P(y)P(x_1,\ldots,x_n \mid y)}{P(x_1,\ldots,x_n)} \qquad [2.1]$$

Using the naive independence assumption that

$$P(x_i \mid y, x_1,\ldots,x_{i-1},x_{i+1},\ldots,x_n) = P(x_i \mid y) \qquad [2.2]$$

for all i, this relationship is simplified to

$$P(y \mid x_1,\ldots,x_n) = \frac{P(y)\prod_{i=1}^{n} P(x_i \mid y)}{P(x_1,\ldots,x_n)} \qquad [2.3]$$

Considering $P(x_1,\ldots,x_n)$ is constant in the input, we can utilize the accompanying rule:

$$P(y \mid x_1,\ldots,x_n) \propto P(y)\prod_{i=1}^{n} P(x_i \mid y)$$

$$\Downarrow \qquad [2.4]$$

$$\hat{y} = \arg\max P(y)\prod_{i=1}^{n} P(x_i \mid y)$$

and we can use Maximum A Posteriori estimation to estimate $P(y)$ and $P(x_i \mid y)$, where $P(y)$ is the relative frequency of class y in the training set.

Different types of Naive Bayes classifiers vary for the most part by the suppositions they make with respect to the dissemination of $P(x_i \mid y)$.

2.2.2 Support Vector Machine

Support Vector Machines rely upon the possibility of decision planes that describe decision limits. A choice plane is one that isolates between a lot of articles having diverse class participation. A schematic precedent appears in the representation underneath. In this precedent, the articles have a place either with class GREEN or RED. The disengaging line describes a point of confinement on the right half of which all articles are GREEN and to the other side of which all things are RED. Any new article (white hover) tumbling to the privilege is marked, i.e., grouped, as GREEN (or delegated RED should it tumble to one side of the isolating line).

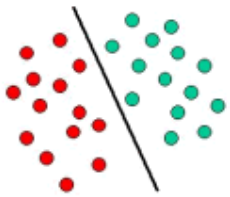


Figure 2: SVM Linear Classifier

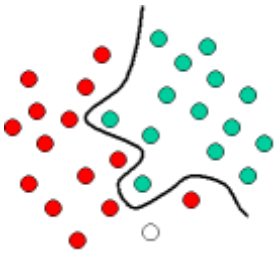


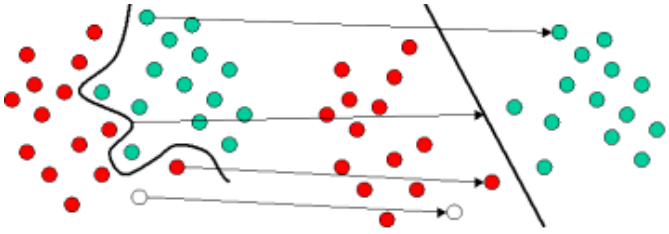
Figure 3: SVM Non-Linear Classifier

The above is an exemplary case of a direct classifier, i.e., a classifier that isolates a lot of items into their individual gatherings (GREEN and RED for this situation) with a line. Most characterization assignments, in any case, are not unreasonably straightforward, and frequently progressively complex structures are required so as to make an ideal partition, i.e.,

effectively group new items (test cases) based on the precedents that are accessible (train cases). This circumstance is portrayed in the delineation underneath. Contrasted with the past schematic, unmistakably a full division of the GREEN and RED articles would require a bend (which is more complicated than a line). Arrangement undertakings are dependent on attracting isolating lines to recognize objects of various class enrollments are known as hyperplane classifiers. Bolster Vector Machines are especially suited to deal with such errands.

The delineation beneath demonstrates the essential thought behind Support Vector Machines. Here we see the first items (left half of the schematic) mapped, i.e., revamped, utilizing a lot of numerical capacities, known as portions. The way toward reworking the articles is known as mapping (change). Note that in this new setting, the mapped items (right half of the schematic) is straightly divisible and, along these lines, rather than developing the mind-boggling bend (left schematic), we ought to just to find a perfect line that can disconnect the GREEN and the RED articles.

Figure 4: SVM Feature Mapping Using Kernel



2.2.2.1 Advantages

- Prediction accuracy is generally high
- Robust. Works when training examples contain errors
- Fast evaluation of the learned target function

2.2.2.2 Disadvantages

- Long training time
- Difficult to understand the learned function
- Not easy to incorporate domain knowledge

2.2.2.3 Mathematical Formulation

Given training vectors $x_i \in \mathbb{R}^p$, i=1,..., n, in two classes, and a vector $y \in \{1, -1\}^n$, SVM solves the following primal problem:

$$\min_{w,b,\zeta} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \zeta_i \qquad [2.5]$$
$$\text{subject to } y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i,$$
$$\zeta_i \geq 0, i = 1, ..., n$$

Its dual is

$$\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha - e^T \alpha \qquad [2.6]$$
$$\text{subject to } y^T \alpha = 0$$
$$0 \leq \alpha_i \leq C, i = 1, ..., n$$

where e is the vector of all ones, $C > 0$ is the upper bound, Q is an n by n positive semi definite matrix, $Q_{ij} \equiv y_i y_j K(x_i, x_j)$, where $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$ is the kernel. Here preparing vectors are verifiably mapped into a higher (possibly unbounded) dimensional space by the capacity ϕ.

The decision function is:

$$\text{sgn}(\sum_{i=1}^{n} y_i \alpha_i K(x_i, x) + \rho) \qquad [2.7]$$

2.2.3 Stochastic Gradient Descent

Stochastic Gradient Descent is a basic yet extremely effective way to deal with discriminative learning of straight classifiers under raised misfortune capacities, for example, (direct) Support Vector Machines and Logistic Regression. In spite of the way that SGD has been around in the machine learning system for a long time, it has gotten a great deal of thought just starting late with respect to gigantic scale learning.

SGD has been effectively connected to a huge scale and meager machine learning issues frequently experienced in content order and regular dialect preparing. Though the information is little, the classifiers in this module effectively scale to issues with more than 10^5 planning models and more than 10^5 features.

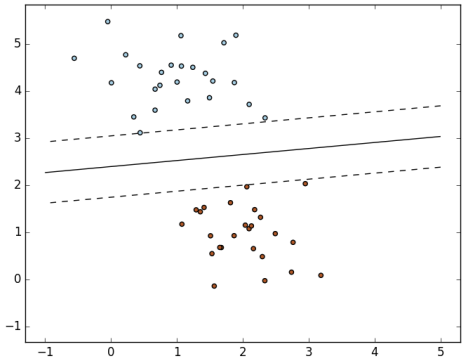


Figure 5: SGD Learning Routine

2.2.3.1 Advantages

- Efficiency.
- Easy to implement (great chances for code tuning).

2.2.3.2 Disadvantages

- SGD needs various hyper parameters, for example, the regularization parameter and the number of iterations.
- SGD is sensitive to feature scaling.

2.2.3.3 Mathematical Formulation

Given a set of training precedents $(x_1, y_1), \ldots, (x_n, y_n)$ where $x_i \in \mathbf{R}^n$ and $y_i \in \{-1, 1\}$, we will likely gain a linear scoring function $f(x) = w^T x + b$ with model parameters $w \in \mathbf{R}^m$ and intercept $b \in \mathbf{R}$. So as to make prediction, we basically take a gander at the indication of $f(x)$. A typical decision to locate the model parameters is by limiting the regularized training error by

$$E(w, b) = \frac{1}{n} \sum_{i=1}^{n} L(y_i, f(x_i)) + \alpha R(w) \qquad [2.8]$$

2.2.4 Passive Aggressive Algorithms

The passive-aggressive algorithms are a group of calculations for huge scale learning. They are like the Perceptron in that they don't require a learning rate. Be that as it may, in spite of the Perceptron, they incorporate a regularization parameter C.

2.2.4.1 Mathematical Formulation

$$\ell(\mathbf{w};(\mathbf{x},y)) = \begin{cases} 0 & y(\mathbf{w}\cdot\mathbf{x}) \geq 1 \\ 1 - y(\mathbf{w}\cdot\mathbf{x}) & \text{otherwise} \end{cases} \quad [2.9]$$

2.2.5 Ridge Regression

Generally, Tikhonov regularization technique is mostly used for regularization for those problems which are not well-presented. In measurements, the strategy is known as Ridge regression, and with numerous free disclosures, it is additionally differently known as the Tikhonov Miller technique, the Phillips Twomey technique, the compelled direct reversal technique, and the strategy for straight regularization. It is identified with the Levenberg Marquardt calculation for non-linear least-squares issues.

2.2.5.1 Advantages

- Multi-collinearity
- Biased but smaller variance and smaller Mean Squared Error
- Explicit Solution

2.2.5.2 Disadvantages

- Shrink coefficients to zero but cannot produce a parsimonious model
- Dimensionality can be reduced
- It can create high bias error

2.2.5.3 Mathematical Formulation

Ridge regression tends to a portion of the issues of Ordinary Least Squares by forcing a punishment on the extent of coefficients. The ridge coefficients limit a punished lingering sum of squares,

$$\min_{w} ||Xw - y||_2^2 + \alpha||w||_2^2 \quad [2.10]$$

Here, $\alpha \geq 0$ is a complexity parameter that controls the amount of shrinkage: the larger the value of α, the greater the amount of shrinkage and thus the coefficients become more robust to collinearity.

2.2.6 Perceptron Classifier

The perceptron is a calculation for managed learning of twofold classifiers (works that can choose whether info, spoken to by a vector of numbers, has a place with some explicit class or not). It is a kind of direct classifier, i.e. an arrangement calculation that makes its expectations dependent on a straight indicator work joining a lot of loads with the component vector. The calculation takes into consideration web-based learning, in that it forms components in the preparation set each one in turn.

The perceptron calculation goes back to the late 1950s; its first execution, in custom equipment, was one of the primary fake neural systems to be delivered. The Perceptron is a straightforward calculation reasonable for extensive scale learning. As a matter of course:

- It does not require a learning rate.
- It isn't regularized (penalized).
- It refreshes its model just on oversights.

The last trademark suggests that the Perceptron is somewhat quicker to prepare than SGD with the pivot misfortune and that the subsequent models are sparser. Two classes of focuses, and two of the endlessly numerous direct limits that different them. Despite the fact that the limits are at about right edges to each other, the perceptron calculation has no chance to get off picking between them.

2.2.6.1 Advantages

- In the event that an answer exists, it very well may be found
- Work best for well-defined problems
- If things go wrong the parameters can be adjusted
- Lots of training algorithms exist
- MLP works easily with continuous values
- Deals well with noise

2.2.6.2 Disadvantages

- NN does not contain an easily understood representation of their knowledge
- MLP confides in altogether on the calculations used to make it

- MLP does not scale well
- Once trained MLP is not updated without retraining
- Retraining does not preserve previous MLP knowledge
- Plan of data sources and yields can profoundly affect MLP achievement
- Inputs may require pre-processing
- Outputs may require post-processing
- To get the correct number of layers and neurons requires experimentation

2.2.6.3 Mathematical Formulation

Perceptron can be referred as an algorithm for learning a binary classifier. It is a function that maps its input x (a real-valued vector) to an output value f(x) (a single binary value):

$$f(x) = \begin{cases} 1 & \text{if } w \cdot x + b > 0 \\ 0 & \text{otherwise} \end{cases} \qquad 2.11$$

where w is a vector of real-valued weights, w. x is the dot product , $\sum_{i=1}^{m} w_i x_i$, where m is the number of inputs to the perceptron and b is the bias. The inclination moves the choice limit far from the source and does not rely upon any info esteem.

2.3 Conclusion

In this chapter, we discussed different types of supervised learning algorithms. Each algorithm has its own advantages & disadvantages. We wanted to compare our selected methods by its applications and usefulness. We have used different algorithms for our experiments to get comparative results of their performances on our lyrics datasets and based on the different results we can name the method by which we achieved the best result in our experiment.

Chapter 3

Experiments Setup

3.1 Introduction

In this chapter, we will discuss the settings we have used for our experiment. We will describe all the steps we followed in our experiment, how we collected our data, how we distributed them into our data sets, the process we picked to extract the features and which features we have considered. Then we will describe the feature selection process and how we designed our classifier for this experiment.

3.2 System Diagram

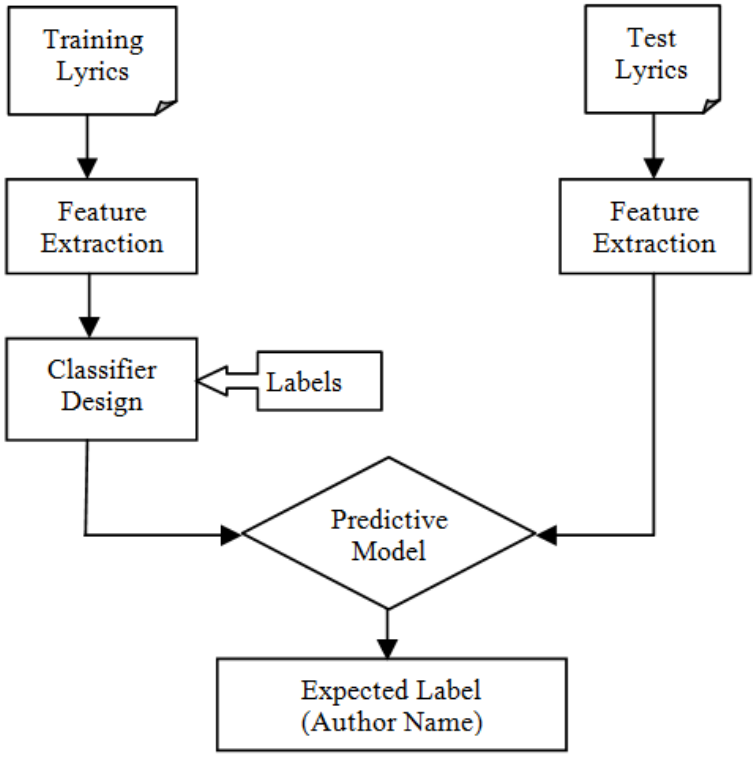


Figure 6: System Diagram For Author Identification

3.2 Data Collection

Data collection was a great challenge for us. The internet provides a great deal to find lyrics online. However, there is no reliable large data set for Bengali lyrics with proper author information. Although we have selected some great Bengali songwriters for experiments who have thousands of songs of their own but unfortunately, very few of those are available on the internet.

There are some websites for Bengali song lyrics as well as some mobile applications but some of those provide only the lyrics, without the related information about those songs, even without the name of the authors. Despite having difficulties to find resources, we have gathered the lyrics to all available songs by each of the seven lyricists and stored the lyrics of each song into a text file format.

For our research, we have made our dataset by parsing Bengali lyrics from various websites (5) and mobile applications (2 android apps) mentioned in Table 3.1. The authors and the number of their lyrics of songs that we are using in our dataset are listed in Table 3.2. For our experiment, we have selected seven great lyricists in Bengali literature and they are:

Kazi Nazrul Islam (1899 – 1976): He was a great Bengali poet, writer, musician, and revolutionary from South Asia. He is the national poet of Bangladesh. He wrote and composed music for approximately 4,000 songs, which collectively also known as Nazrul Geeti.

Rabindranath Tagore (1861 - 1941): He was a Bengali polymath from the Indian subcontinent, a Nobel winning poet, musician, and artist. He wrote and composed music for nearly 2,230 songs, collectively known as rabindrasangit (“Tagore Song”).

Fakir Lalon Shai (1772 - 1890): He was a prominent Bengali philosopher, Baul saint, and songwriter from the Indian subcontinent. It is estimated that Lalon composed nearly 2,000 to 10,000 songs. However, only about 800 songs from the vast collection of his songs, are considered authentic.

Shah Abdul Karim (1916 - 2009): He was an immensely popular Baul musician from Bangladesh who was well known as Sindhi Sufi poet. He wrote and composed over 1500 songs.

Gauriprasanna Mazumder (1924 - 1986): He was a Bengali lyricist and writer, known for his work in Bengali cinema in the black and white era.

Salil Chowdhury (1922 - 1995): He was an Indian music composer, lyricist, poet, playwright, writer and singer, who mainly composed for Bengali, Hindi, Malayalam film and other films.

Atulprasad Sen (1871 - 1934): He was a great Bengali composer, lyricist, and singer. He wrote 212 poems and most of the poems were used as songs.

Table 1: Data Sources

Websites	Mobile Applications
1. বাংলায় গানের কথা □Bangla Song Lyrics[1]	1. Lalon Song – লালন গীতি[2]
2. বাংলা লিরিক | Bangla Lyrics3	2. Bangla Lyrics[4]
3. Shah Abdul Karim[5]	
4. Collection of Nazrul Geeti[6]	
5. বাংলায় লিখি[7]	

Table 2: The Authors And Lyrics Count

Author	Lyrics Count
Kazi Nazrul Islam	95
Rabindranath Tagore	117
Fakir Lalon Shai	129
Shah Abdul Karim	37
Gauriprasanna Mazumder	24
Salil Chowdhury	30
Atulprasad Sen	69

[1] http://banglasonglyrics.com/
[2] FinalApps
[3] http://www.ebanglalyrics.com/
[4] TechTunes Mobile App
[5] http://shahabdulkarim.com/category/lyrics/
[6] https://collectionofnazrulgeeti.wordpress.com/
[7] http://preyrona.blogspot.com/2008/02/list-of-mass-songs-ganasangeet.html

3.3 Data Set Distribution

The authors and their lyrics of songs that we are using in our data set are listed in Table-I. In this study we have used a moderate data set of song lyrics in text files of seven renowned lyricists: Rabindranath Tagore (RT), Kazi Nazrul Islam (KNI), Atulprasad Sen (AS), Gauriprasanna Mazumder (GM), Fakir Lalon Shai (FLS), Shah Abdul Karim (SAK), and Salil Chowdhury (SC) of Bangladesh who have a great diversity in their writings.

There are two approaches to find authorship of a song using acoustical and verbal information. In this paper, we have selected the latter, specifically using the lyrics of songs. We have presented six methods Stochastic Gradient Descent (SGD) [13], Support Vector Machine (SVM) [14], Naive Bayes (NB) [15], Passive-Aggressive (PA) [16], Ridge Classifier (R) [17], and Perceptron Classifier (P) [18] to identify the author.

For our experiment, we have considered 501 songs of seven different authors and constructed three data sets. For each data set, we used 10-Fold cross-validation to distribute train.

Table 3: Data Distribution Of Data Set

Data Set Name	Authors	Lyrics Count
D2A	RT, KNI	212
D4A	RT, KNI, GM, SC	266
D7A	RT, KNI, GM, SC, AS, FLS, SAK	501

3.4 Feature Extraction

Feature extraction is a primary pre-processing step for machine learning works. By extracting features we can reduce the dimensionality of our data set. In our study, we represent each lyric of a song as a vector. In this work, we considered the following features:

3.4.1 Word Level N-Grams

N-gram is a neighboring sequence of n items, where the items can be phonemes, syllables, letters, words or base pairs according to the application from a given sequence of text or speech corpus. When the items are words, n-grams also called shingles. N-gram can be referred as "unigram" when the size is one, or "bigram" when the size is two, or "trigram"

when the size is three. Larger sizes are often referred to by the value of n, like, "four-gram", "ve-gram", and so on.
In our experiment, we used the bi-gram method for feature extraction.

3.4.2 Stop Words

Generally, stop words are a set of commonly used words in any language. However, stop words are considered very critical in applications because when we remove the stop words, we can focus on the important words. On the other hand, if we count the stop words, we can get a more accurate result by focusing on the vocabulary as a whole.

In our study, we have used stop words in separate ways. Firstly, we have not considered stop words and removed those from our data set and followed the rest of the steps. Secondly, we have accounted stop words as features and in this experiment. Moreover, in the latter situation, we have got a better result than the first one because stop words are important and meaningful in terms of lyrics.

3.5 Feature Selection

Feature selection is very important in preprocessing of text classification because it enhances the adaptability, productivity, and precision of a text classifier. Generally, a good feature selection technique ought to think about algorithm and domain properties [1]. The fundamental thought of feature selection is to choose a subset of features from the initial files. Feature selection is performed by keeping the words with the highest score according to a predetermined measure of the importance of the word [2]. The selected features retain original physical meaning and provide a better understanding of the data and learning process [3].

3.5.1 Frequency Based Feature Selection

Frequency-based feature selection is selecting the terms that are most common in the class. Frequency can be either defined as document frequency (the number of documents in the class c that contain the term t) or as collection frequency (the number of tokens of t that occur in documents in c).

3.5.1.1 Term Frequency

Term Frequency (TF), which estimates how regularly a term happens in a record. Since each archive is diverse long, it is conceivable that a term would seem significantly more occasions in long records than shorter ones. Hence, the TF is frequently divided by the report length as a method for normalization.

$$tf\,(t, d) = log(t\,f_{t,d} + 1) \tag{3.1}$$

For word *w* in our song collection,

$$TF(w) = \frac{Number\ of\ times\ a\ word\ (w)\ appears\ in\ a\ document}{Total\ number\ of\ words\ in\ that\ document}$$

3.5.1.2 Inverse Document Frequency

Inverse Document Frequency (IDF), estimates how essential a term is. While processing term frequency, all terms are considered similarly essential. In any case, it is realized that specific terms, for example, "is", "of", and "that", may show up a lot of times yet have little significance. In this manner, we have to load down the incessant terms while scaling up the uncommon ones, by processing the Eq-3.2.

$$idf\ (t, D) = log\frac{n}{x_t} \tag{3.2}$$

3.5.1.3 TF-IDF

TF-IDF is composed of two scores, term frequency (TF) and inverse document frequency (IDF). TF-IDF ensures that a term is an important indexing term for a document if it occı frequently in it. Then again, terms which happen in numerous reports are appraised less critical ordering terms because of their low IDF. There are many variants of TF-IDF. The accompanying basic variation was utilized in our analyses. 3.3

$$weight_{t,d} = \begin{cases} \log(tf_{t,d} + 1) \log \frac{n}{x_t} & \text{if } tf_{t,d} \geq 1 \\ 0, & \text{otherwise} \end{cases}$$

In our study, we have used TF-IDF as a feature selection method. There are multiple variants for TF-IDF but for our experimental setup we have used the formula mentioned below.

$$tfidf = TF\ *\ IDF \tag{3.4}$$

Thus, weight of each feature in a vector is defined as:

$$weight(w) = tfidf(w) \quad 3.5$$

Where *tfidf* is the frequency of word t in document d, n is the number of documents in the text collection and xt is the number of documents where word t occurs.

3.6 Classifier Design

We have designed our model into two consecutive parts. Firstly, in the training part, we extract features from training corpus and pass into one of the methods to make the classifier model. Lastly, in the detection part, our system gets input as lyrics to test identification model and generates the result.
We have designed a classifier using that six methods mentioned earlier to identify the author or lyricist of a song. Figure 6 illustrates the design and workflow of our classifier that we considered in the experiments.

We have considered the following setup for the six methods we have selected to design our classifier model for our experiments.

3.6.1 Stochastic Gradient Descent (SGD)

SGD is an optimization method and an effective approach to linear classifiers.
In our experiment, alpha, = 0.0001 to multiply the regularization term C, penalty = l2, loss = hinge loss for loss function, and maximum iteration = 50.

3.6.2 Support Vector Machine (SVM)

SVM classifier is a set of supervised learning methods used especially for classification and detection [4]. We have used the linear function as kernel function and one-vs-the-rest [5] method for multi-class classification where only one model is trained for only two classes. In our experiment we have kept the value of penalty parameter, C = 1.0 as default, penalty = l2, tolerance for stopping, tol = 0.0001, loss = square of the hinge loss, and maximum iteration = 1000.

3.6.3 Naive Bayes (NB)

It is a method based on applying Bayes theorem with the assumption of independence between the features. For our study, we have used Multinomial NB [6] for our multinomial distributed data, smoothing parameter = 0.01.

3.6.4 Ridge Regression

This is another generalized linear model that uses for some problems of Ordinary Least Squares.
For our system, we have used regularization strength, = 1.0, and tolerance for stopping was 0.001.

3.6.5 Perceptron

This generalized linear model uses for large scale learning. For our experimental setup, we have used = 0.0001 to multiply the regularization term, number of iteration = 50, verbose = 0 and eta, = 0.01 to multiply the updates.

3.6.6 Passive-Aggressive

This is another Generalized Linear model uses for large scale learning. In this research, we have used maximum step size, C = 1.0, number of iteration = 50, and verbose = 0 for verbosity level.
For all experimental models, our term was bi-gram and the highest value of TF and IDF is 1.0.

3.7 Conclusion

In this chapter we have discussed the complete process we followed for our experiment. We have discussed all the details in each stage we have gone through to complete our experiment.

Chapter 4

Experimental Analysis

4.1 Experimental Result And Analysis

Several experiments have been performed to detect the author from song lyrics. We have divided our experiments into three groups, where each group of experiments has done for each data set. Moreover, each group contains two experiments, one for with stop words and another one for without stop words. Hence, we have included six experiments here in total where we have considered the experiments for dataset D2A, D4A, and D7A while once considering the stop words and then without the stop words too and we have achieved different accuracy each time.

4.1.1 Experiment Using D2A Data Set

Table 5 displays the final author identification result from corpus by considering our D2A data set and stop words where Table 4 displays the result by considering the same data set but eliminating the stop words. From the comparison of both the tables, we can observe that for data set D2A Naive Bayes gives higher accuracy of 93.9% and 92% by considering the stop words and eliminating the stop words respectively. On the other hand, Perceptron gives the lowest accuracy of 85.9% while considering the stop words. Another interesting finding was, without Perceptron in Table 5, all the results were between 90 to 94 percent.

Table 4: Author Identification Result For D2A Data Set Without Stop Words

Classifier Name	Precision	Recall	F1 Score	Accuracy
Ridge Regression	0.873	0.873	0.873	0.873
Perceptron	0.869	0.869	0.869	0.869
Passive Aggressive	0.892	0.892	0.892	0.892
SVM	0.897	0.897	0.897	0.897
SGD	0.883	0.883	0.883	0.883
Naïve Bayes	0.920	0.920	0.920	0.920

Table 5: Author Identification Result For D2A Data Set With Stop Words

Classifier Name	Precision	Recall	F1 Score	Accuracy
Ridge Regression	0.901	0.901	0.901	0.901
Perceptron	0.859	0.859	0.859	0.859
Passive Aggressive	0.925	0.925	0.925	0.925
SVM	0.934	0.934	0.934	0.934
SGD	0.925	0.925	0.925	0.925
Naïve Bayes	0.939	0.939	0.939	0.939

4.1.2 Experiment Using D4A Data Set

Table 6, Table 7 expresses the result of author identification from corpus by considering our second dataset D4A. Here, we have achieved the most accurate result, 85% from Naïve Bayes in Table 7 by having the stop words in the corpus. Though the number of authors increased in D4A the number of the song lyrics were very low for the new two authors. Hence, this may be a reason for the decrement of the accuracies than D2A dataset. However, for Table 6, Regression gives the least accurate result, 73.4% by not having stop words.

Table 6: Author Identification Result For D4A Data Set Without Stop Words

Classifier Name	Precision	Recall	F1 Score	Accuracy
Ridge Regression	0.734	0.734	0.734	0.734
Perceptron	0.787	0.787	0.787	0.787
Passive Aggressive	0.771	0.771	0.771	0.771
SVM	0.742	0.742	0.742	0.742
SGD	0.778	0.778	0.778	0.778
Naïve Bayes	0.839	0.839	0.839	0.839

Table 7: Author Identification Result For D4A Data Set With Stop Words

Classifier Name	Precision	Recall	F1 Score	Accuracy
Ridge Regression	0.775	0.775	0.775	0.775
Perceptron	0.826	0.826	0.826	0.826
Passive Aggressive	0.834	0.834	0.834	0.834
SVM	0.764	0.764	0.764	0.764
SGD	0.838	0.838	0.838	0.838
Naïve Bayes	0.850	0.850	0.850	0.850

4.1.3 Experiment Using D7A Data Set

Table 9 demonstrates the consequences of the author identification by considering our third data set D7A and the most accurate result, 86.7% have gained from Naïve Bayes. Without the accuracy of Perceptron classifier, the other accuracies from the rest of the classifiers increased than D4A. Table 9 is for displaying result by accounting stop words and Table 8 is for showing result by not keeping stop words in the corpus. In Table 8 the results are lower than Table 9 because it eliminates the stop words.

Table 8: Author Identification Result For D7A Data Set Without Stop Words

Classifier Name	Precision	Recall	F1 Score	Accuracy
Ridge Regression	0.792	0.792	0.792	0.792
Perceptron	0.784	0.784	0.784	0.784
Passive Aggressive	0.825	0.825	0.825	0.825
SVM	0.774	0.774	0.774	0.774
SGD	0.829	0.829	0.829	0.829
Naïve Bayes	0.853	0.853	0.853	0.853

Table 9: Author Identification Result For D7A Data Set With Stop Words

Classifier Name	Precision	Recall	F1 Score	Accuracy
Ridge Regression	0.807	0.807	0.807	0.807
Perceptron	0.791	0.791	0.791	0.791
Passive Aggressive	0.835	0.835	0.835	0.835
SVM	0.776	0.776	0.776	0.776
SGD	0.845	0.845	0.845	0.845
Naïve Bayes	0.867	0.867	0.867	0.867

4.1.4 Experiment Conclusion

In our research, we have experimented our system by six different and effective classifiers on three different data sets in two separate ways. Comparatively, we have achieved the best results from Table 5 by Naive Bayes classifier where the accuracy rate is 93.9%.

4.2 Performance Evaluation

We have determined accuracy from Precision, Recall, and F1- Score [7].

True Positive (tp) = Number of lyrics assigned accurately to x

False Positive (fp) = Number of lyrics which does not belong to x but assigned to class x inaccurately

False Negative (fn) = Number of lyrics which is not assigned to x but belongs to class x

4.2. 1 Precision

Precision (P) [7] is the proportion of the predicted positive cases that were correct, as calculated using the eq-4.1.

$$precision = \frac{tp}{tp + fp} \quad 4.1$$

4.2.2 Recall

Recall (R) [7] is the proportion of positive cases that were correctly identified, as calculated using the eq-4.2.

$$recall = \frac{tp}{tp + fn} \quad 4.2$$

4.2.3 F1-Score

The traditional F-measure or balanced F-score or F1-score (F1) [7] is the harmonic mean of precision and recall. We can calculate it using eq-4.3.

$$f1\text{-}score = 2\left(\frac{precision * recall}{precision + recall}\right) \quad 4.3$$

F1 score reaches its best value at 1 and the worst score at 0.

4.2.4 Micro Averaging

There are different ways to do the average binary metric calculation. For example, macro, micro, weighted, samples etc. In our experiment, we have used the micro averaging method which gives an equal contribution to each sample-class pair to overall metric. In a multi-class classification scenario, micro-average is considered as the preferable one, if you suspect there might be a class imbalance or in multiclass classification where a

majority class is to be ignored. As we know micro-averaging in a multiclass setting will produce precision, recall, F1-score and that are all identical to accuracy [8]. Since we have used this technique, all our results that have been displayed by table 4.1 to 4.6 have the same precision, recall, F1 and accuracies.

To make it more explicit, consider the following notation:
y the set of predicted (sample, label) pairs
ŷ the set of true (sample, label) pairs

$$P(A,B) := \frac{|A \cap B|}{|A|}$$
$$R(A,B) := \frac{|A \cap B|}{|B|}$$
$$F_{ß}(A,B) := (1 + ß^2)\frac{P(A,B) * R(A,B)}{ß^2\, P(A,B) + R(A,B)}$$

Then the metrics are defined for micro-averaging as:

Precision: $P(y, \hat{y})$
Recall: $R(y, \hat{y})$

In other words, in a multi-class setting, we count all false instances which means every single negative will be a false negative for a class whereas, every single false prediction will be a false positive for a class.

4.3 Experimental Analysis

From our experiments, it is clear that fewer classes are easy to predict and that is the reason for great accuracies for fewer authors in Fig-4.1 but to build a proper model we need more authors. However, the number of authors is not the only factor that can affect in this model; we need a great number of lyrics for each author to build a successful predictive model. That is why the accuracies fall down drastically for 4 authors but then when the number of authors and their number of lyrics are increased at a moderate rate the accuracies are increased with that.

More features could lead to achieving higher accuracy. Other features such as parts-of-speech bi-gram, rhyme and style features, and sentimental analysis would be a great addition to reduce the error rate used in previous works.

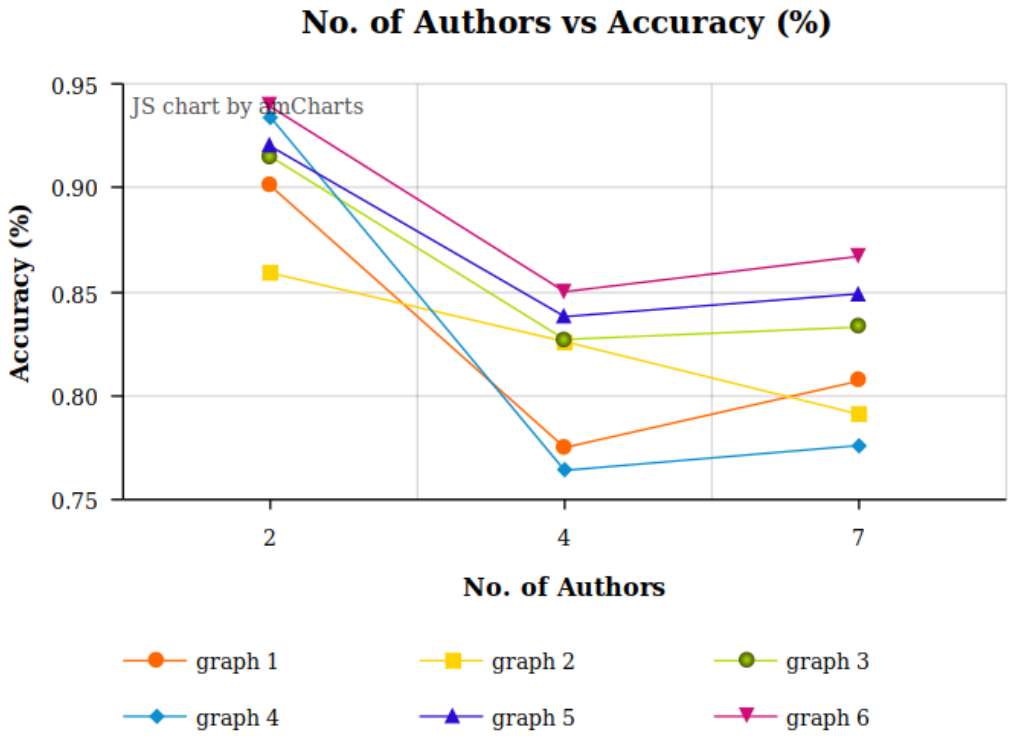


Figure 7: Number Of Authors VS Accuracy

4.4 Conclusion

In this chapter, we have discussed all the results we have gained from the different setup in our experiment. We have evaluated the performance that we got through the experiment. We have also analyzed the experiment and mentioned the accuracy we achieved.

Chapter 5

Conclusion & Future Work

5.1 Limitations

The performances of this thesis work are subject to the limitations mentioned bellow-

- **Corpus Availability:** Though Bengali is used by a large number of people, availability of Bengali corpus is not that high. For our experiment, we did not find any standard Bengali document corpus. So, we have to create our own corpus.

- **Noisy Corpus:** As we have to create our own corpus by parsing on-line newspapers, there were lots of noisy data in the corpus. In this study, we removed those noisy data sometimes manually and sometimes pragmatically.

- **Time Complexity:** Our system takes relatively much time to train and test the data because of considering all the features as a relevant feature in the feature selection step.

- **High Storage:** As we considered all the features, we needed more space to store our data and model which can be improved using dimensionality reduction algorithms.

5.2 Future Work

There are many ideas and improvements that have not focused on this experiment which may apply to improve the usefulness of the system. Followings are some which we would like to incorporate in our future work.

- **Dimensionality Reduction:** Dimensionality Reduction is referred to reduce the size of the feature vector. The main advantage of dimensionality reduction is that it reduces process time and also the storage required to store the feature vector. It

also sometimes helps to improve the performance of the Classifier. In the future, we would like to apply some dimensionality reduction algorithm such as Latent Dirichlet Allocation (LDA), Principal Component Analysis (PCA) etc.

- **Hybrid Classifier:** In the future, the authors would like to incorporate a hybrid system using multiple classification algorithms for evaluating the performance.

- **Multi-class Classification:** Sometimes a document can belong to multiple class which is not considered in this study. In the future, we would like to incorporate a multiclass classification.

- **Hierarchy Classification:** It is possible to divide a class into multiple child class. For example, Cricket, Football, Basketball etc. can be a child class of Sports class. For our further studies, we will try to classify the hierarchy classification of the documents.

5.3 Conclusion

We investigate the importance of lyricists. Bengali music has great potential but for lack of proper online archive of song lyrics, it's very tough to do work on lyrics.
Our research concludes the following information:

- Our project presents the first attempt to build author identification from the lyrics of a song.
- The most accurate result of getting the author name has achieved by Naive Bayes classifier.

In the future, we intend to explore more lyrics and develop other algorithms and techniques to obtain more effective results with more accuracy.

References

[1] X. S. D. x. Z. a. X. L. Z. q.Wang, "An optimal svm-based text classification algorithm," in *2006 International Conference on Machine Learning and Cybernetics*, Aug 2006.

[2] J. F. I. D. E. F. C. J. R. Elena Montañés, "Measures of Rule Quality for Feature Selection in Text Categorization," in *Springer Berlin Heidelberg*, Berlin, Heidelberg, 2003.

[3] H. L. a. H. Motoda, "Feature Extraction, Construction and Selection: A Data Mining Perspective," in *Kluwer Academic Publishers*, Norwell, MA, USA, 1998.

[4] J. S.-T. N. Cristiani, An introduction to support vector machines and other kernel-based learning methods, Cambridge university press, 2000.

[5] C. W. J. Weston, "Support vector machines for multi-class pattern recognition," *ESANN*, vol. 9, pp. 219-224, 1999.

[6] E. F. B. P. a. G. H. A. M. Kibriya, "Multinomial naive bayes for text categorization revisited," in *Australasian Joint Conference on Artificial Intelligence. Springer*, 2004.

[7] D. M. W. Powers, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *International Journal of Machine Learning Technology*, vol. 2, p. 37–63, 2011.

[8] "scikit-learn.org," [Online]. Available: https://scikit-learn.org/stable/modules/model_evaluation.html. [Accessed October 2018].

[9] T. Zhang, "Solving large scale linear prediction problems using stochastic gradient descent algorithms," in *Twenty-first International Conference on Machine Learning, ser. ICML '04*, New York, NY, USA, 2004.

[10] C.-C. C. a. C.-J. Lin. [Online]. Available: http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf.

[11] I. Rish, "An empirical study of the naive bayes classifier," in *IJCAI 2001 workshop on empirical methods in artificial intelligence*, 2001.

[12] J. K. S. S.-S. Y. S. K. C. O. Dekel, "Online Passive-Aggressive Algorithms," in *JMLR 7*, 2006.

[13] a. E. P. N. R. D. H. Smith, Applied regression analysis, 2 ed., vol. 3, New York:

Wiley, 1996.

[14] S. R. E. F. Y., "Large margin classification using the perceptron algorithm," in *Machine Learning*, 1999.

[15] E. Stamatatos, "A survey of modern authorship attribution methods," *J. Am. Soc. Inf. Sci. Technol. 60*, vol. 3, pp. 538-556, March 2009.

[16] M. Mara, "Artist Attribution via Song Lyrics," 2014.

[17] N. R. a. R. A. M. R., "Rhyme ans style features for musical genre claasification," in *9th International Conference on Music Information Retrieval (ISMIR'08)*, 2008.

[18] Ö. B. E. U. Đlker Nadi Bozkurt, "Authorship Attribution: Performance of various features and classification methods," [Online]. Available: https://users.cs.duke.edu/~ilker/papers/conference/iscs07.pdf.

[19] T. Chakraborty, "Authorship Identification Using Stylometry Analysis in Bengali Literature," 2012.

[20] S. D. a. P. Mitra, "Author Identification in Bengali Literary Works," in *Lecture Notes in Computer Science*, Berlin Heidelberg, Springer , 2011, pp. 220-226.

[21] L. Tarlin, "Authorship Attribution of Song Lyrics," 2017.

[22] S. S. L. a. A. B. Phani, "Authorship Attribution in Bengali Language," ICON, 2015.

Appendix A

List of Publications

International Conference Paper

- Nazmun Nisat Ontika, Fashiul Kabir and Mohammad Nurul Huda "Author Identification from Song Lyrics" accepted on First International Conference on Smart Technologies in Computer and Communication (SmartTech-2017), February 2017, Jaipur, India.

- Nazmun Nisat Ontika, Fashiul Kabir and Mohammad Nurul Huda "Author Identification from Song Lyrics" accepted for the student conference on 2017 IEEE International Conference on Imaging, Vision & Pattern Recognition (icIVPR), January 2017, Dhaka, Bangladesh.