University of Sheffield

Authorship Identification

Da Eun Kim

Supervisor: Dr Mark Stevenson

A report submitted in partial fulfilment of the requirements
for the degree of Bachelor of Science

in the

Department of Computer Science

December 9, 2019

Declaration

All sentences or passages quoted in this report from other people's work have been specifically acknowledged by clear cross-referencing to author, work and page(s). Any illustrations that are not the work of the author of this report have been used with the explicit permission of the originator and are specifically acknowledged. I understand that failure to do this amounts to plagiarism and will be considered grounds for failure in this project and the degree examination as a whole.

Name: Da Eun Kim

Signature: Da Eun Kim

Date: 09.12.2019

Abstract

Authorship Identification also known as Authorship Attribution is a task to identify the author of given documents. This problem is studied by many researchers and lots of approaches were proposed. Many of the studies use n-grams of characters, words frequencies and stemmed words. Also, SVM and Naive Bayes classification is widely used. This project is going to look through useful proposed approaches with a balanced dataset.

Contents

List of Figures

Chapter 1

Introduction

Authorship is to ensure the writers who have substantive contributions to the paper have credits as an author and to remind their role to have responsibility on their publication. The task of Authorship Identification is to find the likely authorship in the given documents. According to Bozkurt et al (2007), as the growth in Internet usage, information with unclear writers has been increased. This can be a problem since plagiarism, pseudonym malicious emails in these anonymous information can be a legal consequences. Also, Juola(2012) said it is applicable to detect disputed historical inquires, fake news, journalism and in forensic linguistics.

1.1 Background

The early period of the authorship identification studies, there were main methodological limitations during accuracy evaluations. For example, the data is too long (e.g. whole books) or not written in unified style and hard to compare among different methods because of the lack of suitable data. Also, the evaluation method was mainly visual inspection. However, a research on authorship identification moves forward as the research in machine learning, information retrieval and natural language processing increases. In addition, since the late 1990s the huge amount of electronic texts was generated through the Internet media usage such as emails and blogs. As the availability of electronic documents and performance of text processing, a various methods of acknowledge the author of a disputed text has become possible.
According to Stamatatos(2008), authorship identification problem has been presented in text representation and text classification. Also, it is multi-class single-label text categorization task in a machine learning.

The most common features in authorship identification studies are bag-of-words(Stamatatos, 2006), stylistic features(Zheng et al., 2006), and word and character-based n-grams (Peng et al., 2003; Juola, 2006). In addition, these features have been worked successfully.

1.2 Aim

As the availability of electronic documents and performance of text processing, a various methods of acknowledge the author of a disputed text has become possible. This project will develop a system to identify the author of a document based on the authors' writing style.

In addition, the project will develop a system to identify authorship on group of anonymous texts. As the project is about identify real writer, the features of writing style which is called stylometric features are the essential things to consider.

1.3 constraints

According to Sapkota et al.(2015) the length of the documents and the number of the candidate authors have effect on the accuracy of authorship identification approaches.

1.4 Report Structure

Chapter2 Literature Survey
This part will cover basic knowledge which is needed to understand general task of authorship identification.

Chapter3 Requirements and analysis
In this part, it will explain traditional prototype of authorship identification. In addition to that, this part will state the analysis of the possible system construction.

Chapter4 Progress
This part will describe how far the project has been reached.

Chapter5 Conclusion and plan
Final part will conclude the process I got so far and set a plan for the reset of the process of this project.

Chapter 2

Literature Survey

2.1 Language Model

Language model is a model that assigns a probability to a sentence which can be seen as a word sequence. There are two big approaches to handle language model; Statistical language model and artificial neural network model.

2.2 N-gram model

N-gram is one of a Statistical Language Model(SLM) based on the count words. However, not considering all the words in a sentence but only N words in a sentence. The limitation of SLM is when the training corpus has a sparsity problem. As the longer a target sentence, the more possibility not to have the sentence in the corpus. To solve this problem, by shortening the testing words, it can have higher probability to count the target words. In authorship identification studies, word-based and character-based n-gram has been used. 'N' denotes the number of words or characters to observe. However, n-grams also has limitation. The main problem arises in the part that n-gram ignores. For example, if a sentence and n are each given as 'markets across Eastern Europe' and word-based 2-grams, it does not consider the rest of the two words. Then this approach can have less accuracy than language model which considers the entire sentences.

2.3 CCAT dataset

In this project, two datasets will be used which are CCAT10 and CCAT50. CCAT contains corporate and industrial newswire stories. The topics which is covered in the corpus are the stock exchange, China's politics and business, airlines association, car etc. Also, some author indicate their publishing company at the bottom such as 'Prague NewsRoom' and 'Air Cargo NewsRoom'. CCAT has been used in several studies (Plakias and Stamatatos, 2008; Sapkota et al., 2011; Sari et al., 2017). According to Stamatatos(2017), CCAT is the combination of personal style and preferred thematic nuances which will be represented as features. Also, there are words both that are author-specific and frequently appear in multiple documents.

Figure 2.1: *CCAT dataset.*

	CCAT10	CCAT50
Number of authors	10	50
Total number of documents	1000	5000
Avgerage characters per doc	3089	2595
Average words per doc (including punctuations)	606.2	608.5
Average sentences per doc	21.4	21.7
Number of documents per author	100	100

Therefore, these words can be a useful features to classify the author for the specific corpus.

Corpus for authorship attribution has been covered certain text domains. For example, online newspaper articles, newswire stories, etc. CCAT corpus are especially a part of the Reuters Corpus Volume1 (Rose et al., 2002). As Figure 2.1, CCAT10 and CCAT50 corpus have 10 and 50 labelled authors with 100 texts per author. The corpus are divided into the same amount of training and test sets per author. Each author has the same distribution of the training and testing corpus; 50 texts each.

2.4 Feature Selection

Stamatatos(2008) states there are mainly five stylometric approaches; Lexical, Character, Semantic, Syntactic Features. Among these features, Lexical and character features are commonly used in authorship identification studies (Mendenhall, 1887). These approaches provide writers' preferences in topic and writing style with simple processes. For example, sentence and word count length or character n-grams.

2.4.1 Lexical Features

Token-based

Tokens such as a word, number, a punctuation mark. Also can be a sentence length counts and word length counts. These features are applicable to any languages.

Vocabulary richness

It measures the diversity of the word choices by authors. However, theses features have over text length problem. The number of words are increased if the corpus has long texts. Its size increased rapidly at first but gradually will increase relatively slow at last since there will be a word overlaps.

Vectors of word frequencies

By using bag-of-words which is a traditional text representation. Features using BoW are the most straightforward approach to represent texts. Also, it captures correlations between authors and topics effectively(Statamos, 2008; Sapkota, 2015).

Word n-grams

Word n-gram is to capture contextual information

Errors

It captures authors' idiosyncrasies such as spelling and formatting errors. These features can be collected by using spelling checkers. However, the corpus in this project is news articles so capturing spelling and formatting errors might not an effective way to apply.

2.4.2 Character Features

By regarding text as a sequence of characters, many character-level measures can be considered.

Character types

Character types such as the number of letters, digit counts, uppercase and lowercase characters count, letter frequencies, punctuation marks count, etc. (de Vel, et al., 2001; Zheng, et al., 2006) These approach is very simple and straightforward and according to the Grieve(2007), this is quite useful method to quantify the writing style. According to Stamatatos(2008), the advantage of n-gram is that it can collect nuances of style including lexical information and hints of contextual information. Also, it is not sensible to noise so it can more captures authors' writing characteristics such as typo.

Character n-grams

Especially character n-grams have been used largely since it is able to capture stylistic information. In addition to the author's writing style, this approach is straightforward to observe and capture features from documents. Also, it is commonly used for authorship identification studies (Sari et al., 2017).

2.5 Classifiers

2.5.1 Support Vector Machine

Scikit-learn.org(2019) explained that Support Vector Machine(SVMs) is used in a suervised learnging methods in classification, regression and outliers detection problems.
There are several pros and cons on this methods. The main advantage of SVMs is that it is effective in high dimensional spaces. For example, when the number of dimensions is more than the number of samples. However, if the number of features is much more than

Figure 2.2: *Confusion matrix for two classes.*

	Predicted Class 1	**Predicted Class 2**
Actual Class 1	True Positive	False Negative
Actual Class 2	False Positive	True Negative

the number of samples, over-fitting would be happened. Also, it is using a five-fold cross-validation which is very expensive calculation.

SVMs were desinged for binary classification. Therefore for multi-class classification problem, it should combine several binary SVMs or can direcrly learn a multi-class classifier.

2.5.2 Neural Network

A single hidden layer Feed-forward Neural Network model (FNN)and logistic regression were used in the recent studies; Sari et al (2018). Its parameters includes the number of neurons and tunes the dropout rates for each of the datasets.

2.6 Evaluation

There are many classification models and algorithms and we need to evaluate which construction and techniques are the best for the project. There are several ways to evaluate the performance depend on the dataset and method that the system uses.

2.6.1 Performance Measures

To evaluate a classifier, there are a matrix called confusion matrix. As figure 2.2, confusion matrix allows to show the performance of the algorithm. Each cell represents True Positive(TP), False Negative(FN), False Positive(FP) and True Negative(TN). Positive and negative stands for whether the system find class1 or not. True and False stands for whether the item is correctly positive or negative. A classifier's performance can be evaluated based on the followings; Accuracy, Precision, Recall and F-measure. Accuracy is the ratio of the system said that is correct over the size of the testing data. Recall is the ratio of the system actually guess the correct answer over the actual answer. Whilst precision is the ratio of the actual answer over the system said that is the answer. Finally the F-measure is defined as the harmonic mean of the precision and recall.

Chapter 3

Requirements and Analysis

This chapter will identify possible approaches. Then look through feasible feature selection and classifiers to implement authorship identification.

3.1 Design

Figure 3.1 shows the basic process of algorithm. To describe the process. The raw text from each authors goes through the preprocessing steps such as stemming. After preprocessing, extract the features from the given documents such as word frequencies and punctuation frequencies. With these features the system create a classifier to classify an author of the given text.

3.2 Implementation and evaluation

3.2.1 Instance-based approach

In this approach, each training text is a separate feature of an author's writing style. For this approach, vector space model, similairy model based and meta-learning model is appropriate. According to Stamatatos(2008), the large amount of the modern authorship identification approaches thend to have each training text sample as a one element which contributes to the attribution model. A typical architecture is described in Figure 3.1.

Figure 3.1: *Traditional prototype of authorship identification.*

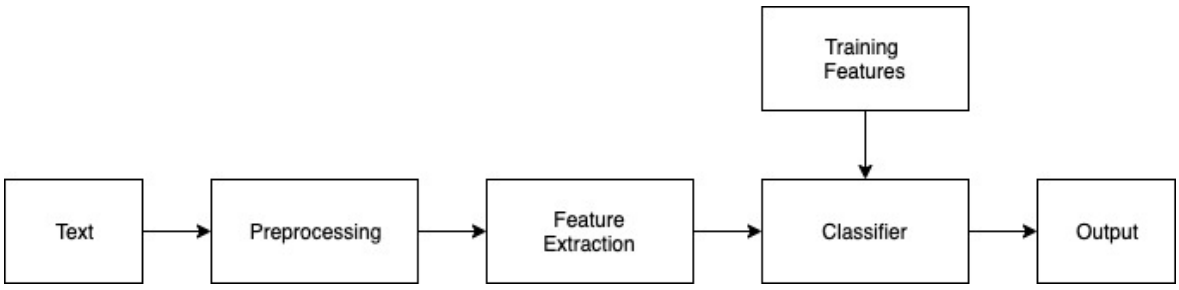

Figure 3.2: *A diagram showing instance-based approaches.*

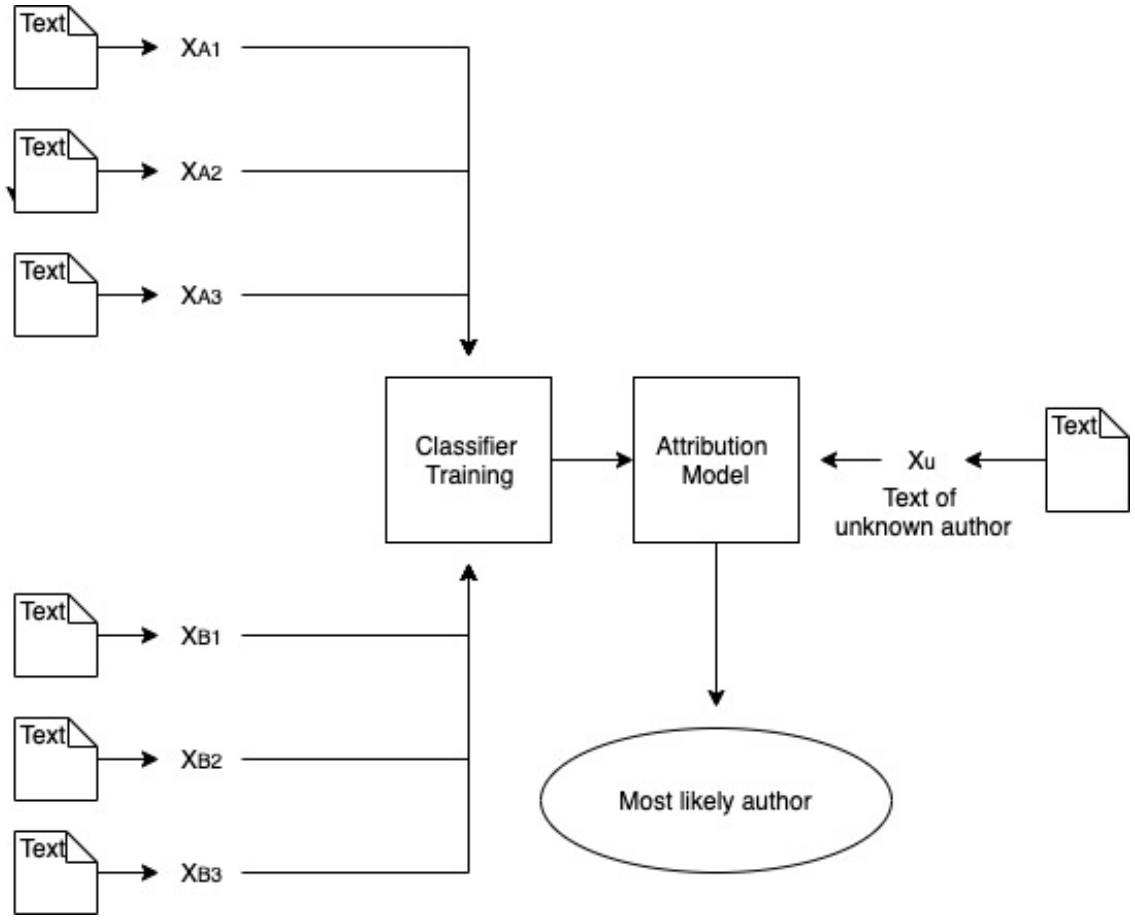

Figure 3.3: *A diagram showing profile-based approaches.*

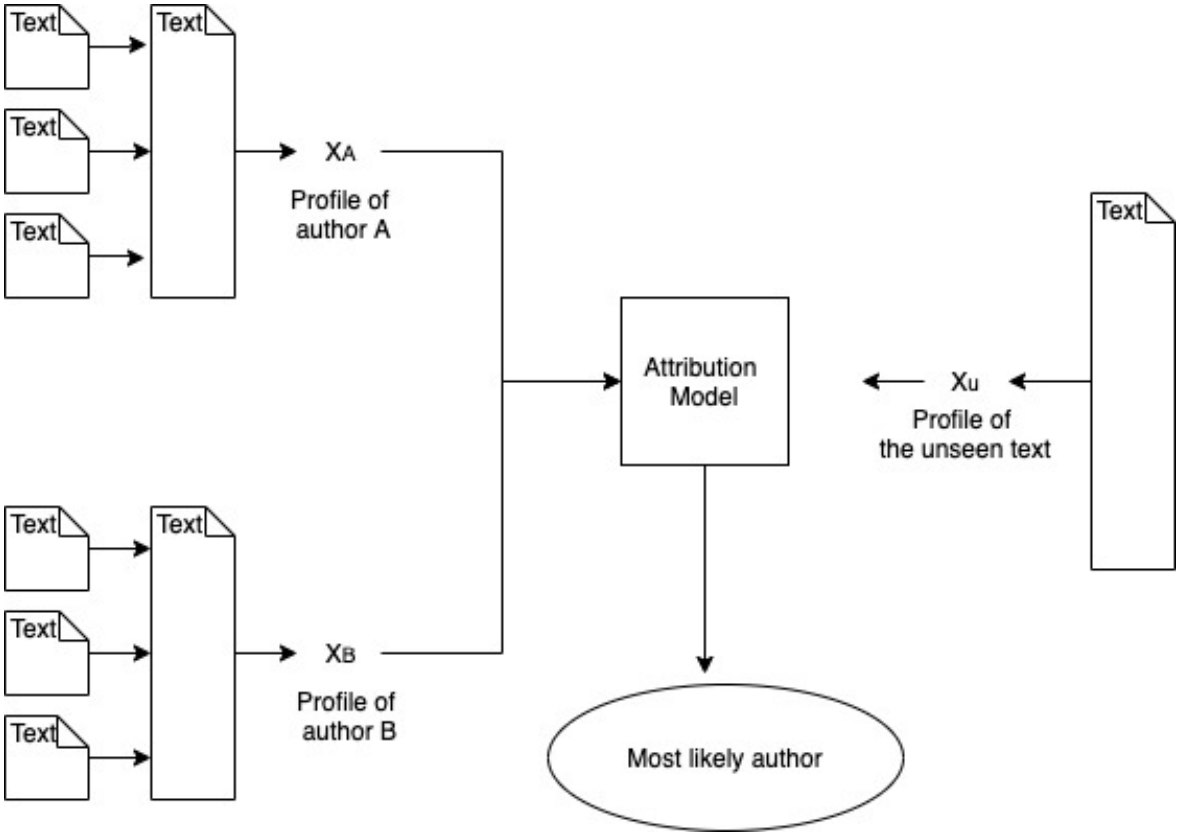

Figure 3.4: *Confusion matrix in authorship identification.*

		Prediction	
		Different authors	Same authors
Actual	Different authors	TP	FN
	Same authors	FP	TN

3.2.2 Profile-based approach

Profile-based approaches, was first proposed by (Mosteller Wallace, 1964). Its approach combines all training text per author and capture features from these text as a property of the author's style. Therefore, vectors consisting of the features will be the profile of an author. The idea is that calculate the distance of the profile of an unseen text and a profile of each author.

3.2.3 evaluation

The confusion matrix in authorship identification is as Figure 3.3(Chen et al, 2011). The Positive and negative in authorship identification denotes whether the author is different or same.

3.3 constraints

There is a sparsity problem in language model. The aim of the language model is to train as much as corpus to a machine and make the model can get close to the probability distribution of the real natural language. However, a vast amount of corpus is needed. Sparsity problem occurs when unavailable to observe and train enough data. As a result, the model cannot modeling a target language.

Chapter 4

Progress

4.1 Fundamental algorithm

The prototype of the system will be as Figure1. The method used in this project is to train a SVM classifier to identify the writing style of each author. This system is a skeletal structure for simple and brief authorship identification implementation. Firstly, preparing the training data which is CCAT10 and CCAT50. In this dataset there are folders which are named as authors' name. To Therefore the data is texts in the folder and labels are the folder name. Secondly, by using scikit-learn method, TfidfVectorizer, transforms the texts into a numerical representation, especially into a matrix of TF-IDF weight. Then classify the texts to the most similar author with SVM classifier. The scikit-learn provides Linear SVC(linear support vector classification) which is useful module to deal with large number of samples. So fit the attribution model to the given training data and predict on test files.

By implementing this process, the accuracy score for CCAT10 is 0.7444 and CCAT50 is 0.6828. The comparison of prediction and the true values for the CCAT10 is in Figure 4.1.

4.2 Implementation

In Figure 4.2, there are lists of the library and the toolkit used in the system. The NLTK and scikit-learn toolkit are used to implement tokenize and vectorize the texts. The process of tokenizing sentences into words is in the Figure 4.3. The tokenizer used here is WordPunctTokenizer() which keeps punctuation in the sentences. As denoting in the Chapter2, punctuation is an important feature in the authorship identification studies. Also, the vectorizer is TfidfVectoizer() which creates vectors with tf-idf weight. Finally, in figure 4.5, SVM using in here is LinearSVC() which has more flexibility in the choice of penalties and loss functions. To use an SVM to make predictions for sparse data, it should be fit on the training data.

Figure 4.1: *Prediction variance graph.*

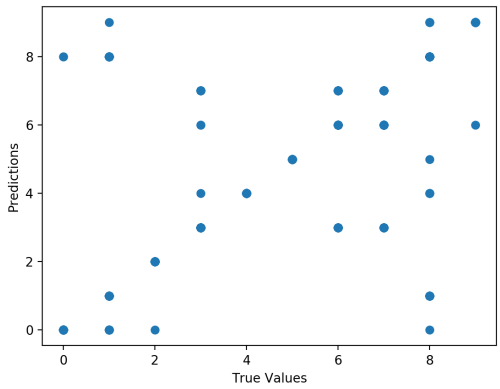

Figure 4.2: *toolkit and library used in the system.*

```
import glob, re
import matplotlib.pyplot as plt
import nltk
from collections import defaultdict
from nltk import WordPunctTokenizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.svm import LinearSVC
from sklearn.metrics import accuracy_score
```

Figure 4.3: *word tokenizer.*

```
wpt = WordPunctTokenizer()
words = []
all_words = []
for author in test_news:
    for file in test_news[author]:
        words.append(wpt.tokenize(test_news[author][file]))
for i in words:
    all_words += i
```

Figure 4.4: *Tf-idf vectorizer.*

```
vectorizer = TfidfVectorizer()
all_texts_vectors = vectorizer.fit(all_texts)
print(all_texts_vectors.vocabulary_)
all_texts_vectors = vectorizer.fit_transform(all_texts)
print(all_texts_vectors.toarray())
```

Figure 4.5: *Predict using svm.*

```
svm = LinearSVC()
svm.fit(x_train, train_authors)
```

Chapter 5

Conclusions and Plan

5.0.1 Conclusions

To implement stylometric features, basically a tokenizer is used to examine in word unit. For word frequencies stemmer and lemmatizer can be additionally used in preprocessing step.

5.0.2 Plan

The Gantt Chart below, Figure 5.1, is a schedule of the project plans for the rest of the academic period. The stream of this project will be like this, however plans can be changed as additional part can be added or delayed.

Figure 5.1: *Dissertation Project Plan.*

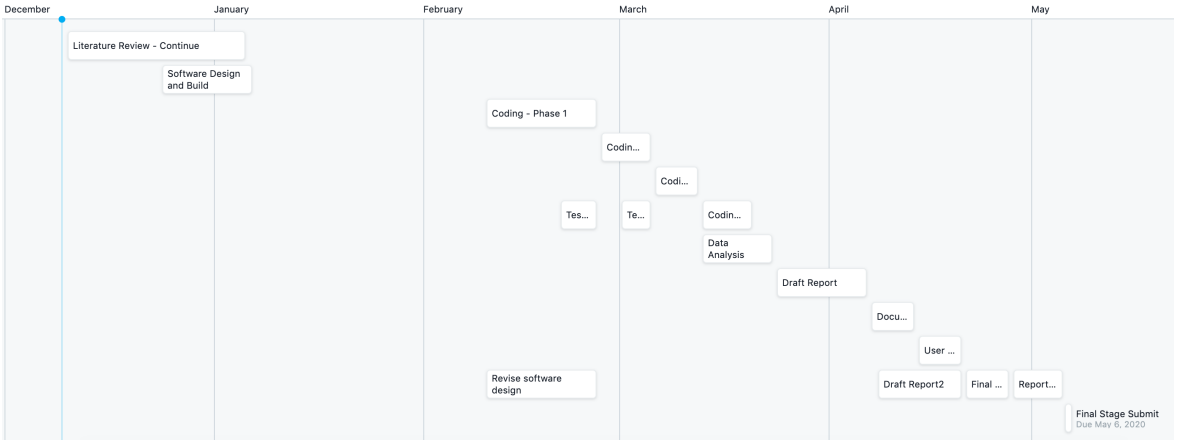

Figure 5.2: *Dissertation Project list.*

Dissetation_Project Plan

Name	Start Date	Due Date
Literature Review - Continue	2019-12-10	2020-01-05
Software Design and Build	2019-12-24	2020-01-06
Revise software design	2020-02-10	2020-02-26
Coding - Phase 1	2020-02-10	2020-02-26
Testing	2020-02-21	2020-02-26
Coding - Phase 2	2020-02-27	2020-03-05
Testing	2020-03-01	2020-03-05
Coding Integration	2020-03-06	2020-03-12
Coding Finalise	2020-03-13	2020-03-20
Data Analysis	2020-03-13	2020-03-23
Draft Report	2020-03-24	2020-04-06
Documentation	2020-04-07	2020-04-13
Draft Report2	2020-04-08	2020-04-20
User Guide	2020-04-14	2020-04-20
Final Testing	2020-04-21	2020-04-27
Report revise	2020-04-28	2020-05-05
Final Stage Submit		2020-05-06