Faculty of Natural and Mathematical Sciences
Department of Informatics

Bush House, 30 Aldwych
Strand Campus
London WC2B 4BG
United Kingdom
https://www.kcl.ac.uk/nms/depts/informatics

KING'S College LONDON

7CCSMPRJ

Individual Project Submission 2019/20

Name: Tzu-Han Lin

Student Number: 1905667

Degree Programme: MSc Data Science

Project Title: Fake News Detection by Tweet Text

Supervisor: Dr Frederik Mallmann-Trenn

Word count: 9831

RELEASE OF PROJECT
Following the submission of your project, the Department would like to make it publicly available via the library electronic resources. You will retain copyright of the project.

☒ **I agree to the release of my project**

□ **I do not agree to the release of my project**

Signature: Tzu Han Lin

Date: 21/08/2020

Department of Informatics
King's College London
United Kingdom

7CCSMPRJ MSc Project

FAKE NEWS DETECTION BY TWEET TEXT

Student Name: Tzu-Han Lin
Student Number: 1905667
Degree Programme: MSc Data Science

Supervisor's Name: Dr Frederik Mallmann-Trenn

This dissertation is submitted for the degree of MSc in Data Science

ABSTRACT

This research focuses on tweets on fake news detection. In this research, tweets from suspended accounts are defined as fake news. The suspended data come from Twitter's report, and unsuspended data is gotten from tweepy API. At first, there are two datasets used, China and Russia's tweets from suspended accounts. Both of them happen in May 2020. After finding topics from a suspended dataset, topics are used for searching results of Twitter. The results are unsuspended datasets.

There are three different types of features used. The first one is the sentiment behind a tweet; the second is hashtag uses. Finally, is tweet's time. As a result, the dataset with Chinese topic has outstanding performance on the k-Nearest Neighbour (k-NN) model. Moreover, the feature used in this model are "polarity score", "subjectivity score", "repeated hashtags", "frequent hashtags", "hour", "minute" and "day". About the dataset with Russian topic, Support Vector Machine (SVM) has the best performance. The used features are "polarity score", "subjectivity score", "hour", "minute" and "day".

ACKNOWLEDGMENTS

Throughout the writing of this thesis, I have received a great deal of support and assistance.

I would first like to thank my supervisor, Frederik Mallmann-Trenn, for giving suggestions on this project. You encourage me and give me support when I cannot find directions.

I would like to acknowledge PhD student, Cristina Gava, for advising and providing a variety of views. You always provide new points and solve my problems.

Also, I would like to thank my family for their support and counsel. You make me focus on my research without any worries.

Finally, I would like to thank my father. Even you passed away, every time I feel anxious, I would remember the word you talked to me. You give me the energy to move on and inspire me.

TABLE OF CONTENTS

NOMENCLATURE

FN	**False Negative**	**15**
FP	**False positive**	**15**
FPM	**Frequent Pattern Mining**	**12**
GNB	**Gaussian Naïve Bayes**	**13**
k-NN	**k-Nearest Neighbour**	**13**
LR	**Logistic Regression**	**14**
NLP	**Natural Language Processing**	**9**
NN	**Neural Network**	**14**
SGD	**Stochastic Gradient Descent**	**14**
SVM	**Support Vector Machine**	**14**
TN	**True Negative**	**15**
TP	**True Positive**	**15**

LIST OF FIGURES AND TABLES

Table

Plot

1 INTRODUCTION

1.1 Overview

There are five parts in the introduction section. First, we talk about the impact of fake news, and then narrow our scope and focus on specific countries and period. Next, the relevance and objective of the research is provided. Finally, the structure of this research is briefly explained.

1.2 Impact of Fake News

Nowadays, social media substitute mails and become a standard approach for communication or get in touch with each other. According to statistical data from website – statisa.com, the worldwide active social media population achieve 3.81 billion on 18 May 2020. Moreover, Twitter has 386 million users. On the first hand, by posting our viewpoints or sharing others opinion, social media can bring convenience and intimate connection to our life. On the other hand, there may be some adverse effects, such as cyber-attack, social media anxiety, or fake news crisis.

Fake news is a state-of-the-art problem on social media. In general, fake news is divided into two different aspects. First, some fake news cause audiences to believe hoaxes or lead misinformation happen by manipulating information content or images. For instance, in the 2016 United States presidential election, Donald Trump's first election victory. For a reason, that result of this election is far away from mainstream media prediction.

Second, when misinformation and fake news remarkable increased, more audiences are frauded, which lead them to make wrong decisions. "It is not just making people believe false things – it is also making them less likely to consume or accept information." (David A. Graham 2019). In a long-term, people figure out that it is too hard to recognise between flaw and truth. Thus, they may do not trust the information they received, even over-question it. When human do not have a suitable and dependable method for receiving recent news, lots of problems will appear. Consequently, it may make tasks more complicated while the government try to announce significant decisions, such as mask-wearing policy during COVID-19 pandemic period.

As stated above, the noisy of false information – fake news not only decrease the velocity of information but lead some misjudgement and erroneous decisions. In order to deal with flaw and hoax, fake news detection becomes an enormous mission, if researchers can identify a pattern of fake news spread or features of sockpuppet, then companies who run these social media can detect hoaxes and ban them.

1.3 Focus

There are lots of things happened in 2020. The most important affair is the coronavirus, as known as COVID-19. On account of this pandemic disease, most countries are lockdown from March to May. When countries gradually lifting lockdowns, some issues come. For instance, the relationship between China and America become much more intense. At first, China claimed that coronavirus comes from America. Then American president, Donald Trump, said he had evidence which points out China should take responsibility for COVID-19. Until now, they closed each other consulate in Huston, America and Chengdu, China.

Issues have existed between democratic and republic countries for more than a decade. Not only the intensive relation between China and America but also the relation between Russia and other countries get lots of attention. Both China and Russia are representative republic countries in the world. Therefore, this research focuses on China and Russia, and the period is May 2020.

1.4 Relevance of Research

In previous research, they discuss the categories of fake news, label information as five degrees of correction, research on the user profile, or figure out correlations between speakers' profile and accuracy of their content.

In this research, it focuses on one of the popular social media, Twitter. Based on previous research, some of them discuss about the profile of information providers, and some use Natural Language Processing (*NLP*) to find out which method has the highest accuracy. This research concentrate on the content of tweets.

The suspended account data provided by Twitter includes account information and tweets belong to those accounts which have been terminated by Twitter. With these data, we can compare with the data of unsuspended accounts.

1.5 Objective

The main question of this research is how to detect a tweet comes from a suspended or an unsuspended account.

First of all, this research assumes suspended accounts may be potential fake news spreader. Then, it works on tweets data. The main feature focuses on texts, hashtags and tweeting time. After getting those features, features are used for combination building by different feature selection methods, then see combinations' performance under classifiers.

1.6 Structure

There are nine chapters in this research. Chapter 1 introduces the background knowledge of fake news. Then, chapter 2 provides background knowledge of this research. In this chapter, lots of academic knowledge which are used in this research are provided. The related works are included in chapter 3. In this part, summarise the results or methodologies from previous research is the main point.

Chapter 4 is the methodology. First of all, this chapter talks about methodologies in each procedure. Then, there is a limitation as a final part. In each procedure part, tools and materials are mentioned. Moreover, approaches to using tools are discussed in this part. In Chapter 5 and 6, results and discussion are mentioned. There are two parts of the results in chapter 5. One is the result with Chinese topic, and the other one is with Russian topic.

Furthermore, some plots are displayed for observing the pattern of different classifiers. The primary method for evaluating performance is confusion matrix, and indexes are recall, precision and F1-score. The discussion part discusses some problems of this research and suggestion for further research. The seventh chapter describes the summaries of results, then compare them.

Last but not least, the last two chapters are references and appendixes. Essays which are mentioned in this research are listed in the reference section. Appendixes are used for storing some additional results.

2 BACKGROUND

2.1 Overview

The background section provides some professional knowledge of this research. In this research, six academic pieces of knowledge are used. Those knowledge relate to a tweet analysis. First, Natural Language Processing (NLP) is used for the text of a tweet. The Frequent Pattern Mining (FPM) is used for hashtags in a tweet. The rest four parts, feature selection, cross-validation, classifier and confusion matrix is about building classifiers and analysis results.

2.2 Natural Language Processing (*NLP*)

Natural language processing (*NLP*) is a computer program, which is used for connecting the interactions between computers and natural human languages. *NLP* involves speech recognition, natural language understanding, and natural-language generation. This research adopts two of *NLP* techniques, topic analysis and sentimental analysis.

2.2.1 Topic Analysis

Topic analysis is one of *NLP* techniques. With topic analysis, we can figure out the topics of texts. Moreover, the meaning of texts is extracted automatically. Topic analysis can be split into topic modelling and topic classification. With topic modelling, it can deduce patterns of texts and put similar expressions into a cluster. In topic modelling, topic tags do not be defined, in other words, it distributes texts with similar content and produce some tags used in texts of this topic, but no specific topic tags. Different from topic modelling, topic classification requires topics before analysis. The approach of topic classification is tagging different texts as the class they belong to. In this research, topic analysis means topic modelling.

The approach of topic analysis clusters texts into different topics is by counting words and grouping words which is in the same patterns. For instance, the related words of Food may have greasy, yummy, unpalatable or flavourless. With calculating word frequency and grouping related words, we can cluster texts into different groups.

2.2.2 Sentimental Analysis

Same as topic analysis, the sentimental analysis is one of *NLP* techniques. In the topic analysis, it takes words from each text to do cluster. When it comes to sentimental analysis, it determines a piece of text (or sentences) rather than a word. As humans, there is a large collection called "sentiment library" for helping the machine to recognise sentiment behind every sentence. There are two types of words in sentiment library, adjectives and phrases; every word in the library has its original score. In every sentence, each word has its score. This score not only comes from the library but also is impacted by its position in the sentence. Therefore, every sentence has a score which counts by words' scores.

In a sentimental analysis, there are two main parts, polarity and subjectivity. The polarity can score a sentence from -1 to +1. The -1 means very negative, and +1 means very positive. The subjectivity can assist machine to judge a sentence is more subjective or more objective. The score of subjectivity judgement is from 0 to +1, the score closer to +1, then the sentence more subjective.

2.3 Frequent Pattern Mining (*FPM*)

The frequent pattern mining is an analytical process. This process can assist users in finding frequent patterns, associations, or causal structures from datasets. In business, frequent pattern mining is used in a lot of aspects, such as basket data analysis, cross-marketing and selling, catalogue design, or medical treatments.

However, the FPM for this research does not relate to business products. This research adopts the concept behind the basket data analysis to find out the frequency of association between hashtags. In basket data analysis, it focusses on purchased products in a single order, then finds out the association that appear much more times. For instance, in table 1, the most frequent two-product association is juice and crisp. In this research, we change products into hashtags.

order no.	**Product**			
	Oil	**Juice**	**Crisp**	**Orange**
1	Yes	Yes	Yes	No
2	Yes	No	No	No
3	No	Yes	Yes	Yes
4	No	No	Yes	No
5	No	Yes	Yes	Yes

Table 1 Example of frequent pattern mining

2.4 Feature Selection

Feature selection is a process to select features automatically or manually. Selected features contribute most to the prediction output (Shaikh, 2018). If selecting irrelevant features, the accuracy of models is impacted. The excellent feature selection can reduce overfitting, improve accuracy and reduce training time.

2.5 k-fold Cross-Validation

Cross-validation is used to evaluate model performances by resampling. The k-fold cross-validation means data is split into k sets, each run takes one set as a test set, and other k-1 sets are training sets. Thus, there are k runs in the k-fold cross-validation. It means there are k outputs. For instance, if k=5, there are 1000 data. Then, in 5-fold cross-validation, it has 5 sets, and each set has 200 data. In each run, there is a test set; in other words, every 200 data represent a test set in each run under this example.

2.6 Classifier

Classifiers are used for classifying data by different features. Every model has at least one feature and at most one target. In supervised learning, models have at least one feature to assist classifiers in finding out the target. Thus, the model without a target is not suitable for supervised learning. In unsupervised learning, models can only have features. With unsupervised learning, classifiers split the data with a similar pattern into the same cluster.

In this research, there are more than one features and one target. Furthermore, there are seven classifiers be selected. All of these classifiers are used for supervised learning, which means the results of a classifier is to predict a specific target.

2.6.1 k-Nearest Neighbour (*k-NN*)

A k-nearest neighbour (*k-NN*) is a type of instance-based learning, or lazy learning (k-nearest neighbours algorithm, n.d.). The basic concept of *k-NN* is distance. After calculating the distances between the predicted point and every training point, *k-NN* selects the first k nearest points, then do plurality vote to decide which class is predicted point belongs to.
For instance, in table 2, if k=3 and the point we want to classify is (1,1). As table 2, the point is classified as class 0.

Data points	Class	Distance with (1,1)
(0, 0)	1	$\sqrt{2}$
(2, 3)	0	$\sqrt{5}$
(3, 1)	0	$\sqrt{4}$
(4, 4)	1	$\sqrt{18}$
(2, 2)	0	$\sqrt{2}$

Table 2 Example of the k-NN classifier

2.6.2 Gaussian Naïve Bayes (*GNB*)

In a Naive Bayes classifier, it assumes every feature is independent of any other feature. The method that Naive Bayes used is maximum likelihood. It means which class has the highest likelihood, then that class is chosen. Moreover, the likelihoods of classes are counted from the possibility of each feature.

In this research, we assume that features are Gaussian. Therefore, Gaussian Naive Bayes (*GNB*) is implemented for classification. The following is the function of the likelihood of features:

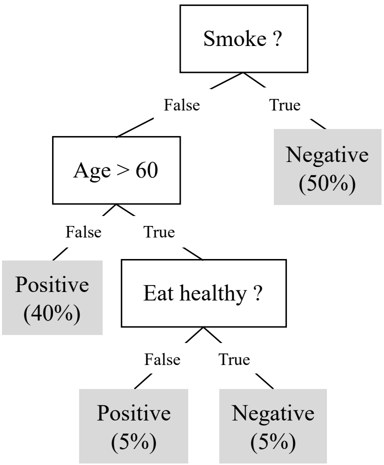
$$\mathcal{P}(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} exp\left(-\frac{\left(x_i - \mu_y\right)^2}{2\sigma_y^2}\right)$$

In this function, parameters σ_y and μ_y are estimated using maximum likelihood (Naive Bayes, n.d.).

2.6.3 *Decision Tree*

A *Decision Tree* is an approach used for supervised learning. The goal of a *Decision Tree* is to create a model which can predict the value of a target variable based on several input variables (Decision tree learning, n.d.). In other words, the *Decision Tree* classifier can predict by several features.

The following is a typical *Decision Tree*:



Decision Tree classifier has two main types of trees, classification tree and regression tree. First one is used for predicting the class. Then, the other one is for predicting a numerical value. This research uses a classification tree.

2.6.4 Support Vector Machine (*SVM*)

SVM is a kind of supervised learning. SVM analyses data by classification and regression analysis. The object of the *SVM* is to find a hyperplane in an N-dimensional space that distinctly classifies the data points (Gandhi, 2018). For instance, if our model has 5 features, then the *SVM* finds a hyperplane in a four-dimensional space. Every two features become x-axis and y-axis in a plane. The value of every two-feature become a point on that plane. Moreover, support vectors are data points that are closer to the hyperplane. Moreover, these points influence the position and orientation of the hyperplane.

When using SVM, there is a decision boundary. This boundary could maximise margins between different class. It means it could distinguish different classes by a decision boundary. Moreover, the support vectors are used for support SVM find the decision boundary.

2.6.5 Neural Network (*NN*)

NN is a series of algorithms which imitates the operation of the human brain. This approach can recognise underlying relationships in a dataset. In a simple *NN*, there are three essential layers, input layer, hidden layer and output layer. First, data is put into an input layer classified by different features. Then, the results of the input layer are conveyed to suitable neuron in the hidden layer. After calculating from hidden layers, results are put into the output layer to produce final results.

2.6.6 Logistic Regression (*LR*)

LR is a statistical model. It is used for predicting targets which only have two possible values. Different from linear regression, the *LR* can deal with more than two variables, which means if a model with a feature and a target, the linear regression is used. This model uses a logistic function to make a prediction.

2.6.7 Stochastic Gradient Descent (*SGD*)

Stochastic Gradient Descent (*SGD*) is one type of Gradient Descent. Gradient Descent is a learning algorithm. A gradient is the slope of a function. It measures the degree of change of a variable in response to the changes in another variable (Roy, 2020). There are three types of Gradient Descent, Batch Gradient Descent, Stochastic Gradient Descent and Mini-batch Gradient Descent.

In Stochastic Gradient Descent, it selects sample randomly without using the whole dataset. The function of *SGD* is:

$$\theta_j = \theta_j - \alpha(\grave{y}^i - y^i)x_j^i , \qquad i \, \epsilon \, [0, n + 1]$$

In this function, y means prediction (or target), x is inputs (or features) for prediction, α is learning rate, and θ_j is a feature from the training set. Therefore, θ_j keeps updating n times. Moreover, n is the times that θ_j become converge ($\grave{y}^i = y^i$).

2.7 Confusion Matrix

A confusion matrix is a summary of prediction results on supervised learning. There are two categories in a confusion matrix, actual output and prediction. When prediction equal to output, then this is a correct prediction. If not equal, then is false prediction. Moreover, results can be split into four different sides, based on the results of prediction is True or False. Table 3 is a demonstration of a confusion matrix.

		Actual output	
		True	**False**
Prediction	**True**	True Positive	False Positive
	False	False Negative	True Negative

Table 3 Example of Confusion matrix

Data in True Positive (*TP*) means the prediction is right, and the result of prediction is True. False positive (*FP*) means the prediction is wrong, and the result of the prediction is True. When a data with wrong prediction and the result of prediction is False, then it is counted into False Negative (*FN*). Finally, the data with the right prediction and is predicted as False, and then it belongs to True Negative (*TN*).

After using a confusion matrix, we receive four numbers. There is a data count on each side. With the calculation of performance correction, there are lots of indexes calculated and evaluated, such as accuracy, recall.

3 RELATED WORK

3.1 Overview

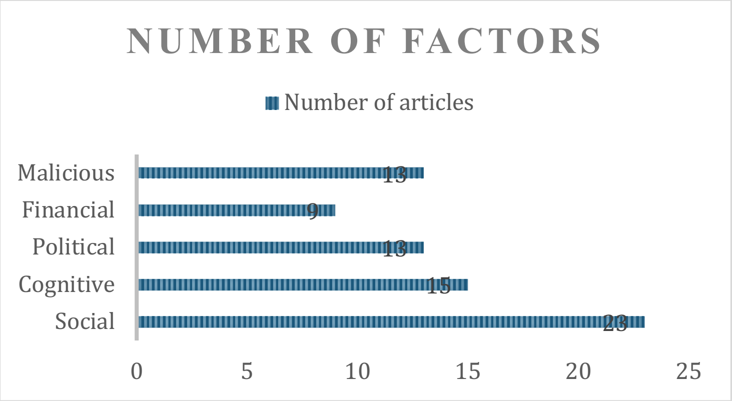
This section displays some previous research. The first part gives categories of fake news detection. Also, there is some research on the number of existing fake news research. Then a dataset called LIAR and related research is mentioned. Since this research uses Natural Language Processing (*NLP*), the research about using *NLP* on fake news detection is provided in the next part. Finally, there is a research focus on social media user pattern and fake news.

3.2 Categories of Fake news detection research

According to the research which relate to categories of fake news, we split them into two aspects. One is categories of fake news, the other is categories of algorithm for fake news detection.

For categories of fake news, in previous research taken by Marlie Celliers and Marie Hattingh (2020, p. 225-231) divide fake news into five main factors. There are social, cognitive, political, financial and malicious factors. Social factors mean users refer to social media as a tool to assist them in getting approval from a specific social group. When more and more users share posts in the same social environment, the effect of information is amplified rapidly. This situation also called conformity and peer influence. Cognitive factors may take advantage of users since they are limited by their knowledge or make them ignore the content of information. When fake news produce is attributed to the political factors, the false political statement might make candidates benefit from affecting voters' decisions or opinions. About financial factor, suppliers provide advertises to earn money because of financial reasons. Those advertise may direct users to websites which contain false information. Malicious factor means fake news is created because of bad intention.

According to research, 23 of 38 articles mention social factors. Then, 15 of 38 mention cognitive factors. Political factors are highlighted in 13 articles, financial factors are referred in 9 of 38 articles. Lastly, malicious factors are mentioned in 13 articles. The amount of each factor is displayed in plot 1.



Plot 1 Number of factors mentioned in previous researches

About categories of algorithm for fake news detection, Álvaro Figueira and Luciana Oliveira (2017, p. 820) found that there are two general approaches for fake news detection, Human intervention and Using algorithm. About human intervention, users can verify the information by fact-checking organisations, such as the Washington Post and Snopes.com. Most fake news is spread by algorithms, in contrast, algorithms can solve this crisis by fake news detection. Targets of algorithms for detecting fake news are based on three different parts, content orientation, diffusion dynamic of messages, and a group of features feeding.

3.3 *LIAR* dataset and relative research

According to the research from Wang (2017), it produced a benchmark dataset called *LIAR* for detecting fake news automatically. In this dataset, there were six-way multiclass provided, pants-fire, false, barely- true, half-true, mostly-true, and true. First, the researcher created a dataset which was selected from POLITIFACT.COM's API[1]. This dataset included posts from online social media. Then, they tested the dataset by five baselines: a majority baseline, a regularized logistic regression classifier (*LR*), a support vector machine classifier (*SVM*), and a convolutional neural network (C*NN*s) and a bi-directional long short-term memory networks model (*Bi-LSTMs*). Then, testing which model had well-performed with text-only data.

As a result, the *LR* and *SVM* models obtained significant improvements, *Bi-LSTMs* had the over-fitting problem, and the C*NN*s had the highest score in all models. Therefore, the researcher used C*NN* combine text with other meta-data to improve performance of C*NN*. The meta-data had the subject, speaker, job, state, party, context and credit history. Finally, they found out there was a significant improvement while using text with all meta-data.

Based on the research in the last paragraph, research (Long, Lu, Xiang, Li, & Huang, 2017) used *LIAR* datasets and long short-term memory network (*LSTM*) to evaluate the performance. This research focused on speaker profile information. The result of this research, "credit history" is the most crucial factor. Moreover, the first four higher combinations are:

I. Credit history
II. Credit history + Job title
III. Credit history + Job title + Location of speech
IV. Credit history + Job title + Location of speech + Party affiliation

3.4 Natural Language Processing (*NLP*) on fake news detection

Shlok Gilda (2017) used a dataset from OpenSources.co and applied two *NLP* techniques: term frequency-inverse document frequency (*TF-IDF*) of bi-grams and probabilistic context-free grammar (PCFG), to detect about 11,000 articles. Moreover, they tested their dataset by five algorithms: Support Vector Machines (*SVM*), Stochastic Gradient Descent (*SGD*), Gradient Boosting, Bounded Decision Tree and Random Forest. In this research, they found out that TF-IDF of bi-grams fed into the *SGD* model can identify results with 77.2% accuracy.

[1] http://static.politifact.com/api/ v2apidoc.html

3.5 Relationship between user patterns and fake news believers

Kai Shu et al. (2018) had research which focuses on profile features of users. They divide profile features into explicit and implicit features.

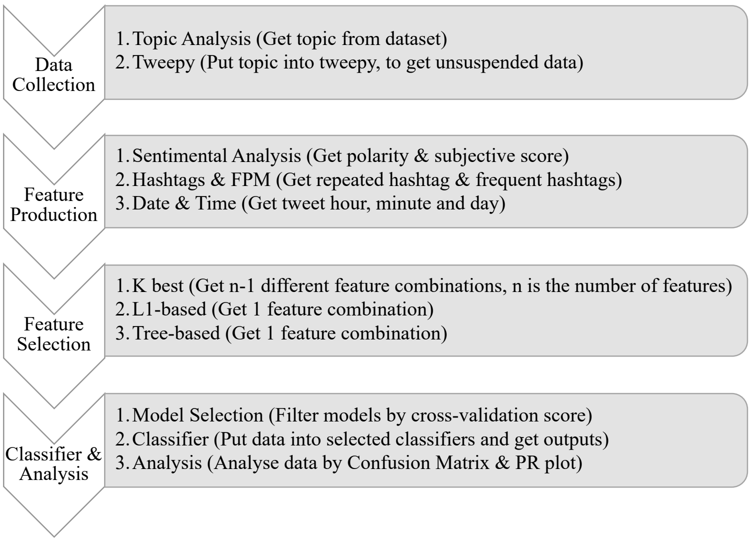
Explicit features include the profile-related, content-related, and network-related field. Profile-related field is user description, the content-related field is about user activities and network-related field relatives to users' social networks. Implicit features can be split into gender and age and personality. Personality is much complicated; in this research, they adopt the Five-Factor Model for personality classifying.

According to the result of this research, these two types of features can use for distinguishing fake news or real news believers.

4 METHODOLOGY

4.1 Overview

This research focuses on quantitative methods. Scores, ranks or categories represent the values in this research. As mentioned in the outline, there are five sections under methodology. They include four principal procedures and a limitation. The procedure is data collection, feature production, feature selection and classifier analysis. Plot 2 is the primary process of this research.


Data Collection
1. Topic Analysis (Get topic from dataset)
2. Tweepy (Put topic into tweepy, to get unsuspended data)
Feature Production
1. Sentimental Analysis (Get polarity & subjective score)
2. Hashtags & FPM (Get repeated hashtag & frequent hashtags)
3. Date & Time (Get tweet hour, minute and day)
Feature Selection
1. K best (Get n-1 different feature combinations, n is the number of features)
2. L1-based (Get 1 feature combination)
3. Tree-based (Get 1 feature combination)
Classifier & Analysis
1. Model Selection (Filter models by cross-validation score)
2. Classifier (Put data into selected classifiers and get outputs)
3. Analysis (Analyse data by Confusion Matrix & PR plot)

Plot 2 Research Process

This research has two categories of data. There are suspended and unsuspended accounts which are distinguished by account status. These two categories are combined into one data for analysis demand. As mention before, China and Russia are the two data we used. Therefore, there are four data at first. After combining, there are only Chinese topic and Russian topic dataset.

4.2 Procedure: Data Collection

There are two types of data should be collected, tweets from suspended and unsuspended accounts.

4.2.1 Tweets of suspended accounts

With suspended data, Twitter reports the accounts they terminated every two or three months at Twitter transparency website[2]. The data in this report have both account information and tweets of suspended accounts. Therefore, we can get complete information on suspended accounts from this report. Moreover, this report categorises data as different countries, months and years.

[2] https://transparency.twitter.com/en/information-operations.html

4.2.2 Tweets of unsuspended accounts

For unsuspended accounts, Twitter provides developers with different API, and *tweepy API* is the one we used. In *tweepy*, the search function is the primary way for data collection. When collecting tweets data, *Cursor()* is the function used. Moreover, this function requires a keyword. The result is the same as putting a keyword on Twitter's search bar. Moreover, the keyword which should use for getting tweets data is about the topic of content. Therefore, topic analysis is used.

Before analysis, data should be cleaned. In this step, we use three tools. There are *tweet-preprocessor*, *spacy* and *nltk*. About *tweet-preprocessor*, the PyPl website[3] mentions that this package is used for cleaning URLs, hashtags, mentions, reserved words and emojis.
For tokenising the text, spacy can assist in making each sentence become word corpus. With *nltk*, it can provide stopwords corpus which includes most of English common words. For topic analysis, ordinary words may impact our results. Furthermore, *nltk* provides *lemmatisation*. *Lemmatisation* is the process of reducing the number of words into a single word by combining common words (Farhad Malik, 2019).

The central part for topic analysis comes from the article written by Susan Li (2018). In this article, she uses three main packages -- *gensim*, *pickle* and *itertools*. The *gensim* package is designed to handle extensive text collections using data streaming and incremental online algorithms (Wikipedia, 2019). With the function of *gensim* package, it can help for topic analysing. The other two packages are used for visualising the results of the topic analysis.

First, we use *gensim* for distinguishing tweets into different clusters; every cluster represents a topic and includes main keywords. In this research, consider the relevance of keywords and tweets, we distinguish tweet text into three clusters. In other words, there are three different topics in each dataset. Besides, to avoid there may do not have enough suitable results, the number of keywords for each topic is two. Therefore, there are three topics in each dataset, and each topic includes two keywords. The outputs of topic analysis are displayed as follow.

There are two keywords for each dataset. Thus, we combine two keywords as one search word.
For instance, keywords for the search function in China dataset are
Topic 1 = ["milesguo" , "wengui"]
Topic 2 = ["milesguo" , "china"]
Topic 3 = ["world" , "epidemic"]

The search words for the search function in Chinese topic dataset are "milesguo+wengui", "milesguo+china" and "world+epidemic". Search results are the same as putting these three search words into the Twitter search bar. Since the policy of Twitter, we can only get tweets that are not older than seven days. We collect data from 08 July 2020 to 23 July 2020. Finally, we label data as suspended and unsuspended. Then, combine the suspended and unsuspended dataset.

4.3 Procedure: Feature Production

There are at least six features in each dataset. They include polarity scores of tweet texts, subjectivity scores of tweet texts, day of tweets, hour and minute users tweeted, repeated hashtags and hashtags with high frequency.

[3] https://pypi.org

In order to get polarity and subjectivity scores, the sentimental analysis is applied. When doing sentimental analysis, *textblob* is used. In *textblob* syntax, we can get polarity and subjectivity of text which is put into syntax. The polarity and subjectivity are displayed in numeric numbers. For polarity, the number is between -1 to 1, 0 means a neural statement. If a number less than 0, it means the statement is negative.

In contrast, when a statement is positive, the number of polarities is larger than 0. For subjectivity, the range is between 0 to 1. If a number is closer to 1, then the statement is more subjective. After cleaning the text of tweets, we put outputs into *textblob* algorithm and receive scores.

With day, hour and minute, the chosen package is *datetime*. By using this function, we can select the year, month, date, day, hour and minute of tweets. In this part, because the suspended and unsuspended use different periods, one is May 2020, and the other is July 2020, if we use the year as a feature, it is no differences between two labels. Moreover, if we use the month as a feature, there may be over-fitting. Since the date for collecting unsuspended data concentrate on one or two weeks, it also has possibilities of over-fitting. To sum up, the features related to time are the hour, minute and day.

With features of hashtags, there are two types. One is repeated tags, if a tweet uses a tag for more than once, the value of repeated tags is "True", otherwise it is "False". The second feature is the tag or tag combination, which has a high frequency. To get the hashtags or combination of hashtags which appear more times, we should do Frequent Pattern Mining (*FPM*). For Frequent Pattern Mining, the *prefixspan* package is used. The *apriori* function in this package can calculate the frequency of hashtags or combination from a two-dimension array. Moreover, *min_support* in this function is used to filter hashtags that appear times higher than *min_support*. We set minimum support as 0.1. Which means if any hashtags or hashtag combination appear more than once in every ten tweets, it or they become features.
Following is a table of features and targets, and the cell below them are value type.

Features							**Target**
polarity number	subjectivity number	repeated hashtags	frequent hashtags	day	hour	minute	suspend
[-1,1]	[0,1]	{True, False}	{True, False}	[1,7]	[0,23]	[0,59]	{True, False}

Table 4 Features and targets of tweets research

4.4 Procedure: Feature Selection

In feature selection, we use feature selection syntax from *sklearn*. Because of the rule of feature selection syntax, values of data should larger than 0. Thus, we should scale data before selecting features. Scaling can modify ranges of numeric data, and change values by means and variances. The one for modifying numeric data is *MinMaxScaler()*. After transforming, the values diminish into a new setting range.

Sklearn package provide three different ways to select features, K-best, L1-based and tree-based.

K-Best

K best removes features which do not have K highest scores. For instances, there are five features in total. If we set K as three, then K best estimators return features with three highest scores. In other words, they discard the other two low-score features. In the K best section, we try every possible number for K. For instance, if there are four features in total, the number put into K is four, three or two. Moreover, each function builds a distinct combination. Therefore, in this case, there are three combinations.

L1-based

L1-based estimators can be used for helping reduce the dimension of data. *SelectFromCombination()* function in L1-based estimators select features with non-zero coefficients.

Tree-based

With tree-based estimators, it can be used to estimate the importance of features, especially impurity-based features. After estimating importance, tree-based can discard irrelevant features, then output the array with values of essential features.

4.5 Procedure: Classifier and Analysis

This research is supervised learning. It means we should use classifiers to predict that one account is suspended or not. Each of them has different features but the same target -- suspend or unsuspended. Main classifiers come from *sklearn* package. For supervised learning, we choose seven classifiers, *k-NN*, *GNB*, *Decision Tree*, *SVM*, *NN*, *LR* and *SGD*.

First of all, we scale the data. As we used in feature selection, *MinMaxScaler()* is the function for scaling. However, different from the feature selection part, the reason for scaling is to downsize the data. With downsizing, classifiers use less time to run algorithms.

In order to select models, cross-validation is used. With cross-validation, it can filter classifiers or combinations by their performances. In this research, we put classifiers into function *cross_val_score()* which is from *sklearn* packages. In this function, we can set a number for cv. The number means the times of splitting and running data. For instance, if we set cv as 5, which means this syntax splits data 5 times each time they have different split. Every split has a score. With running *cross_val_score()*, the gotten output is a list of 5 scores. In this research, we set cv as 10.

The second step would be split into two parts. One is selecting models by classifiers' performance, and the other is by combinations' performance. In the first one, we should calculate the cross-validation scores of each classifier under every combination. If we select models by combinations' performances, then the calculation of cross-validation is used for selecting which combinations are suitable for a specific classifier. Regularly, every combination may use separate classifiers, and every classifier would select suitable combinations.

Next, we calculate the mean and standard of each model, then filter them by average mean of all models and the average standard deviation of all scores. The mean scores of classifiers should be higher than the total average score. Moreover, their standard deviation is used for analysis the model stability.

About classifiers, parameters selection is an essential job. With results of the cross-validation score, parameters of the well-performed and stable combinations are selected. Followings are the selected parameters of classifier.

k-NN: In *k-NN* classifier, we set the number of neighbours as 5.
GNB: There is no particular parameter for *GNB*.
Decision Tree: In the *Decision Tree*, the criterion is "entropy", and max depth of trees is 4.
SVM: The kernel of the function of the *SVM* is "poly".
NN: For the *NN*, the parameter of activation is "identity”, and solver is "adam".
LR: In *LR*, we set the random state as 0.
SGD: About *SGD*, China and Russia have different parameters. First, both of them use "hinge" as the loss function. The dataset of China performs well when the penalty is "elasticnet". For Russia, "l2" has better performance.

Next, a precision-recall curve is produced for each classifier under each combination. This curve can assist us in identifying the quality of a classifier. Moreover, the *Decision Tree* classifier would output the picture of the *Decision Tree*.
Finally, a confusion matrix is used to analyse the results of tweets. A confusion matrix has four results, *TP*, *FP*, *TN* and *FN*. The *TP* means a tweet is predicted come from a suspended account, and it has been suspended. *FP* means a tweet is assumed from a suspended account, but it did not. Tweets in *TN* and *FN* are predicted as unsuspended. True one is unsuspended, and False one is suspended. In other words, the prediction of *FN* is wrong.
By calculating right or wrong prediction and positive or negative result, we can get four indexes, accuracy rate, recall rate, precision rate and F1-score. With these four scores, we can recognise which classifier has better performance on different datasets. Additionally, the combination with better performance is found out.

Table 5 is the table for displaying materials in each procedure.

Procedure	**Package**
Data Collection	*tweepy*, *tweet-preprocessor*, *spacy*, *nltk*, *gensim*, *pickle*, *itertools*
Feature Production	*textblob*, *datetime, prefixspan*
Feature Selection	*sklearn*
Classifier	*sklearn*

Table 5 Materials in each procedure

4.6 Limitation

There is some limitation in this research. It can be discussed in two parts.
First, due to Twitter's policy, the tweets from unsuspended accounts centralise on the date that we run the API. Second, because *NLP* focuses on dealing with English data, we remove the tweets with other languages. However, the hashtags in tweets may still have other languages.

5 Results

5.1 Overview

The results of tweets are the output of topic analysis from data collection, sentimental analysis and Frequent Pattern Mining from feature production, outputs of feature selection, and the main results of well-performance classifiers and combinations. Moreover, some plots are used for comparison.

5.2 Chinese Topic

5.2.1 Topic Analysis

In the data collection part, there are three topics and keyword combination for putting into the search function. In China dataset, three topics and keywords of each topic are

Topic 1 = ["milesguo" , "wengui"]
Topic 2 = ["milesguo" , "china"]
Topic 3 = ["world" , "epidemic"]

The percentage of topics are 38.9% of tokens for topic 1, 31.7% of tokens for topic 2, and 29.4% for topic 3.

In appendix 1 is the distribution of keywords in the first topic. In the 1000 tweets we randomly select from data, more than 250 tweets mention "milesguo", and 200 of 250 are distributed in this topic. Moreover, more than 100 tweets under this topic also mention "wengui".

In appendix 2 is the second topic. The first keyword is the same as topic one, is "milesguo". Furthermore, the one combines with "milesguo" is "china". The distribution of topic 2 is much more equal. In appendix 3 is the final topic. For the first keyword "world", 70 of 100 tweets which have this keyword are in this topic. Also, around 60 tweets under this topic talking about "epidemic".

5.2.2 Sentimental Analysis

The sentimental analysis is for detecting sentiment behind texts in each tweet. After getting the scores, we calculate some statistic information. About the polarity scores, the average score of tweets is 0.58002; most of the tweets use neural statement; there are 19308 (44%) tweets are neural. In other words, although most tweets do not be detected using positive or negative statements, the absolute values of positive statements which has the number between 0 to 1 are higher than others. The maximum and minimum scores of tweets are 1 and -1.
About the subjectivity scores, the average score is 0.31868. Most statements are objective, the number of them is around 29930 (68%). The fewest one is the neural statement. Moreover, the maximum and minimum subjectivity scores are 1 and 0.

5.2.3 Frequent Pattern Mining (*FPM*)

After frequent pattern mining, "郭文贵" is the selected hashtag with 0.11 frequency. Therefore, this hashtag is a feature, and the value of this feature is True or False.

5.2.4 Feature Selection

There are three methods for feature selection, K best, L1-based and tree-based methods. Because there are 7 features, The K best feature selection has 6 combinations. Therefore, there are 8 combinations in total. The selected features of each method are displayed in table 6.

Combination	Features						
	polarity number	subjectivity number	repeated hashtags	frequent hashtags	day	hour	minute
All (k=7)	v	v	v	v	v	v	v
k=6		v	v	v	v	v	v
k=5			v	v	v	v	v
k=4			v	v	v	v	
k=3			v	v		v	
k=2				v		v	
L1-based	v	v	v	v	v	v	v
Tree-based	v	v				v	v

Table 6 Model selection by classifier performances (China)

As a result of the K best method, "frequent hashtags" and "hour" are first-two essential features. On the other hand, the "polarity number" and "subjectivity number" have less critical. Which means, K best method does not think the sentiment of text can cause an account to be suspended. According to the result of L1-based approach, all features may correlate with suspension decision. Finally, the tree-based approach points out the only two things which cause an account be suspended are sentiment and tweet time.

5.2.5 Classifier Results

First of all, we make the model selection by cross-validation scores. About model selection, we use two different aspects. Classifier performances select one, and combination performances select the other.

5.2.5.1 By classifier performances

The results of model selection by classifier performances are displayed in table 7.

Combination	Classifier						
	k-NN	*GNB*	*Decision Tree*	*SVM*	*NN*	*LR*	*SGD*
All (k=7) L1-based	v			v	v	v	v
k=6	v			v	v	v	v
k=5	v			v	v	v	v
k=4	v			v	v	v	v
k=3	v		v				v
k=2	v		v				v
Tree-based	v						

Table 7 Model selection by classifier performances (China)

All combinations select the *k-NN* classifiers. Then, the *SGD* is selected by six combinations without the tree-based combination. Four combinations select the *SVM*, *NN* and *LR*, and all of them have combinations, all features, k=6, k=5 and k=4. The *Decision Tree* has two combinations. Surprisingly, the *GNB* classifier is not selected by any combinations.

Between combination k=3 and k=4, there is a significant change in classifier performance. The only different feature between these two combinations is "day". Therefore, "day" may play a crucial role in classification.

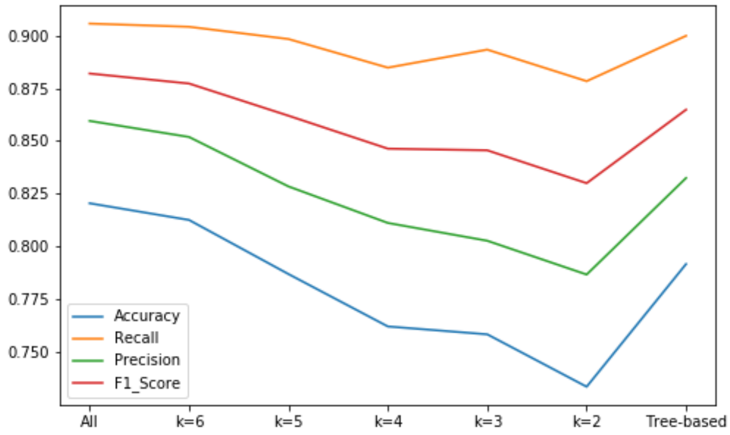
The following part talks about the *k-NN* classifier and *SGD* classifier which are well-performed classifiers. Moreover, we mention the extraordinary results.

k-NN
In our results, there are four indexes used, accuracy, recall, precision and F1-score. Table 8 displays the indexes of the *k-NN* classifier.

Combination	**Index**			
	Accuracy	**Recall**	**Precision**	**F1-Score**
All / L1-based	0.82	0.91	0.86	0.88
k=6	0.81	0.9	0.85	0.87
k=5	0.79	0.89	0.82	0.86
k=4	0.76	0.89	0.81	0.85
k=3	0.76	0.89	0.8	0.85
k=2	0.73	0.88	0.79	0.83
Tree-based	0.79	0.9	0.83	0.87

Table 8 Performance of k-NN classifier (China)

The combination with all features (k=7) has well-performed on *k-NN* classifier. It has not only the highest accuracy, but also the other three indexes are highest. Plot 2 is the line plot of combination performance of *k-NN* classifier.



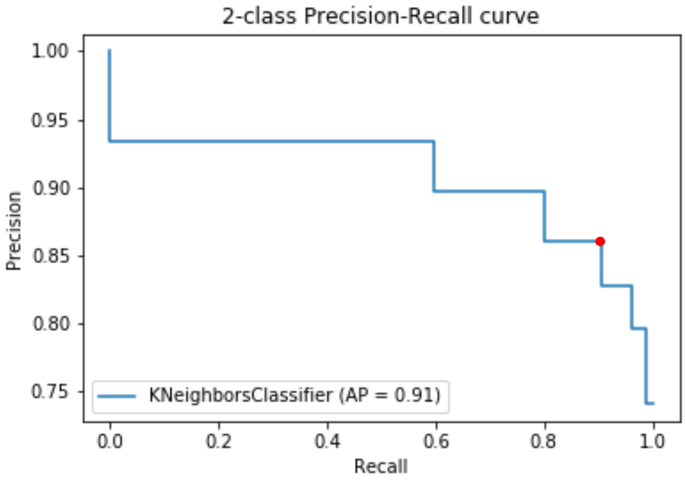
Plot 3 Performance of k-NN under every combination (China)

In this plot, there are four combinations have accuracy around 0.8. These are all (or L1-based), k=6, k=5 and tree-based combinations. When considering features in these combinations, all of them have "hour" and "minute" features. Thus, these two features may be the primary factors for causing higher accuracy on the *k-NN* classifier. Table 9 is the confusion matrix of the *k-NN* classifier. Moreover, we only display values in four combinations which have the score of accuracy around 80%.

Combination	**Confusion Matrix**			
	TP	***FP***	***TN***	***FN***
All / L1-based	5942	971	1327	619
k=6	5932	1031	1267	629
k=5	5894	1220	1078	667
Tree-based	5904	1188	1110	657

Table 9 Confusion Matrix of k-NN with well-performed combinations (China)

According to the results of the *k-NN* confusion matrix, numbers of *TP* and *FN* are stable. The differences in the results of these four combinations are numbers between *FP* and *TN*. Therefore, the feature in each combination may cause *k-NN* classifier has an unstable prediction on unsuspended accounts. The two combinations which have the most significant difference are all and k=5 combination. There are two features in all combinations and not in the k=5 combination, "polarity number" and "subjectivity number". Furthermore, k=6 and tree-based combinations have at least one of these two features. Therefore, the reason for unstable prediction might attribute to these two features.
Consider the well-performed combination in the *k-NN* classifier is all. Plot 3 is the precision-recall curve of this combination.



Plot 4 Precision-Recall curve of k-NN with all combination (China)

The output of precision and recall indexes are 0.86 and 0.91. Therefore, the point of our result is the red point on plot 4.

SGD

The last one is demonstrated in this section is *SGD* classifier. Except for tree-based combinations, the *SGD* classifier is selected by others. It means its performance is better than most of the other classifiers under each combination. Table 10 displays the values of indexes in combinations.

Combination	Index			
	Accuracy	**Recall**	**Precision**	**F1-Score**
All / L1-based	0.78	0.9	0.81	0.86
k=6	0.74	0.96	0.76	0.85
k=5	0.77	0.9	0.82	0.85
k=4	0.77	0.88	0.82	0.85
k=3	0.77	0.9	0.81	0.85
k=2	0.74	1	0.74	0.85

Table 10 Performance of SGD classifier (China)

According to the above table, the all, k=2, k=4, and k=5 has the highest accuracy, recall, precision or F1-score. Therefore, the confusion matrix in table 11 can assist the selection of the well-performed combination.

Combination	Confusion Matrix			
	TP	***FP***	***TN***	***FN***
All / L1-based	5916	1349	949	645
k=5	5875	1320	978	686
k=4	5746	1228	1070	815
k=2	6561	2298	0	0

Table 11 Confusion Matrix of SGD classifier (China)

With confusion matrix, we can figure out that the combination k=2 does not predict any negative prediction, even it gets the highest recall, the precision is the lowest one. Consider the overall scores of indexes, k=5 is better than k=4. Therefore, all and k=5 would be the well-performed combinations in the *SGD* classifier.

5.2.5.2 By combination performances

The results of model selection by combination performances are displayed in table 12.

Classifier	Combination						
	k=7 (All) L1-based	**k=6**	**k=5**	**k=4**	**k=3**	**k=2**	**Tree-based**
k-NN	v	v	v				v
GNB							v
Decision Tree					v	v	v
SVM					v	v	
NN		v	v	v	v	v	
LR			v		v	v	
SGD			v		v	v	

Table 12 Model selection by combination performances (China)

Most classifiers select the k=3 and k=2 combinations. The combination with all features or the first four features (k=4) is selected by one classifier, the *k-NN* or *NN*. The *NN* classifier selects combinations more than other classifiers; it selects five classifiers. The following part would talk about classifier performance under k=2 and k=3 combinations and combination with five-best features. Furthermore, although the model has a combination with all features and *k-NN* has the highest cross-validation score, this model has been mentioned in the last section. For this section, it does not be mentioned again.

Combination with the first two and three features (k=2 and k=3)

Table 13 demonstrates the index results of k=2 and k=3 combinations. The left one belongs to k=2, and the right one is k=3's results.

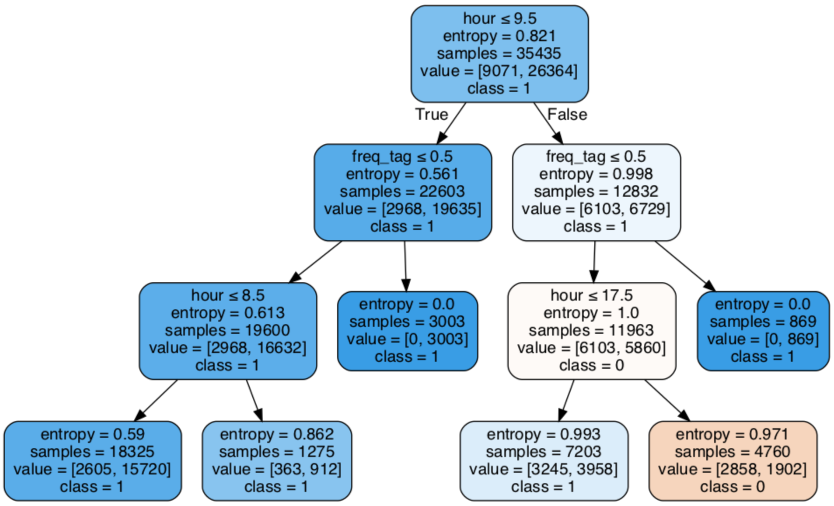
Classifier	Index							
	Accuracy		Recall		Precision		F1-Score	
Decision Tree	0.77	0.77	0.96	0.93	0.96	0.8	0.86	0.86
SVM	0.77	0.77	0.93	0.93	0.93	0.8	0.86	0.86
NN	0.77	0.77	0.93	0.92	0.93	0.81	0.86	0.86
LR	0.77	0.77	0.93	0.92	0.93	0.81	0.86	0.86
SGD	0.77	0.77	0.93	0.92	0.93	0.81	0.86	0.86

Table 13 Performance of k=2 and k=3 combination (China)

According to table 13, the results of k=3 combination with *LR* are the same as ones with *SGD* and *NN*. Furthermore, the results of the *Decision Tree* and *SVM* are the same. In k=2 combination, the *SVM*, *NN*, *LR*, and *SGD* have entirely the same results.

The accuracy and F1-score have no differences between the two combinations. When k=3, the performance of classifiers is same as k=2 or worse. The result shows that the combination of k=2 has a higher performance than k=3. The different feature between them is the "repeated hashtag", so maybe whether tweets have hashtags more than once do not help to figure out an account is suspended or not.

Moreover, the *Decision Tree* classifier gets the best performance with the combination of k=2. Plot 4 displays the *Decision Tree* of this model -- *Decision Tree* with the k=2 combination.



Plot 5 Decision Tree of Decision Tree classifier with k=2 combination (China)

In picture #, the tweet that is tweeted after 17 and without frequent hashtags "郭文贵" much possible be unsuspended. Moreover, if a tweet is tweeted before 9 or after 9 but has frequent hashtags may be suspended.

5.2.5.3 Comparison

With results in last two sections, there are four models selected. There is all combination with *k-NN* or *SGD* classifier, the *SGD* classifier with k=5 combination, and combination with the first two features in the *Decision Tree* classifier.
Back to see the cross-validation scores, table 14 is the mean and standard deviation of scores which repeatedly split dataset and calculate accuracy for ten times.

Models		**Cross-Validation**	
Classifier	**Combination**	**Mean of scores**	**Standard deviation of scores**
k-NN	all	0.79	0.03
Decision Tree	k=2	0.76	0.03
SGD	all	0.77	0.03
SGD	k=5	0.77	0.04

Table 14 Cross-validation results of well-performed models in each section (China)

According to table 14, the model with all combination and the *k-NN* classifier has the highest mean of scores. Moreover, three models have a minimum standard deviation. Thus, the model with all features and the *k-NN* classifier has an outstanding performance in China dataset.

5.3 Russian Topic

When it comes to Russia's result, there are still four parts of outputs, topic analysis, sentimental analysis, frequent pattern mining and the analysis of classifier performances.

5.3.1 Topic Analysis

First, three topics and keywords of each topic are

Topic 1 = ["people" , "football"]
Topic 2 = ["girl" , "naked"]
Topic 3 = ["followers" , "stats"]

For topic 1, it covered 28.9% of Russia data. The percentage of topic 2 and 3 are 42.7% and 28.4%. Topic 2 is the main topic in this dataset.
Appendix 3 is the distribution of the first topic. Most texts in this topic are about "people" and "football". Both of these two words have around 30 tweets. About the second topic, the output is displayed on appendix 4. most tweets in this topic talk about "girl" and "naked". The numbers of tweets about these two keywords are between 60 to 80. Even topic 3 only occupies 28.4%, appendix 4 points out that most tweets with "followers" and "stats" are distributed in this topic.

5.3.2 Sentimental Analysis

There is statistic information of polarity and subjectivity. About the polarity scores, the average score of tweets is 0.09174. It is slightly close to neural. However, polarity scores demonstrate that around 42% of tweets in these files are positive. Moreover, only 17% of the tweets are negative. The maximum and minimum are 1 and -1.

About the subjectivity scores, the average score is 0.35078. Same as China, most statements are objective. The number of objective statements is 34074, around 63%. On the other hand, the least one is neural statements, with less than 5%. Like China, the maximum and minimum subjectivity scores are 1 and 0.

5.3.3 Frequent Pattern Mining (*FPM*)

About frequent pattern mining, the most frequent hashtag of Russia is "forinnovations". However, the frequency of this hashtag is 0.01, which means it only appears one time for every one hundred tweets.

Consider the frequency is not high enough. Therefore, in Russia dataset, there is no frequent hashtag feature.

5.3.4 Feature Selection

There are three methods for feature selection, K best, L1-based and tree-based methods. Because there are 6 features, The K Best feature selection has 5 combinations. Therefore, there are 7 combinations in total. The selected features of each method are displayed in table 15.

Combination	**Features**					
	polarity number	**subjectivity number**	**repeated hashtags**	**day**	**hour**	**minute**
k=6 (All)	v	v	v	v	v	v
k=5		v	v	v	v	v
k=4		v		v	v	v
k=3				v	v	v
k=2				v	v	
L1-based	v	v		v	v	v
Tree-based		v		v	v	

Table 15 Feature selection under each combination (Russia)

As table 15, in the K best method, the least important feature is "polarity number". It means the polarity of text may not impact on suspension decision. Then, is "repeated hashtags" and "subjectivity number". Thus, features related to time are more important than other types of features. Except for the "repeated hashtags", other hashtags are selected by the L1-based method. It means there is no apparent correlation between "suspend" and "repeated hashtags". The tree-based method displays that "subjectivity number", "day" and "hour" have higher importance than other three features.

5.3.5 Classifier Results

Same as China, results of classifier performances are divided into two parts, by classifier performances and combination performances.

5.3.5.1 By classifier performances

The results of model selection by classifier performances are displayed in table 16.

Combination	**Classifier**

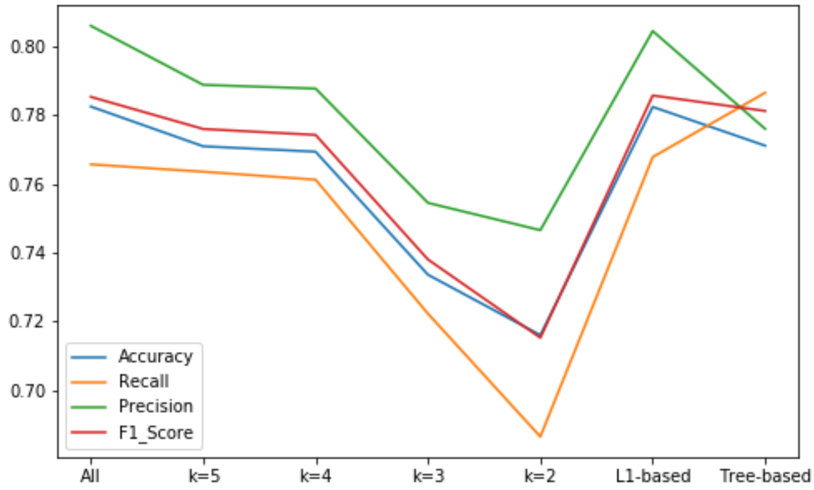
	k-NN	***GNB***	***Decision Tree***	***SVM***	***NN***	***LR***	***SGD***
k=6	v			v			
k=5	v			v			
k=4	v	v		v			
k=3	v						
k=2	v	v	v				
L1-based	v			v			
Tree-based	v	v					

Table 16 Model selection by classifier performances (Russia)

In table 16, the *k-NN* is the selected classifier every combination. Four combinations select the *SVM*. The combination k=4, k=2 and tree-based select *GNB* as one of the classifiers. Only k=2 select *Decision Tree* as the classifier. Surprisingly, the *NN*, *LR* and *SGD* do not be selected by any combination.

k-NN

According to table 16, the *k-NN* is selected by every combination. Moreover, the outputs with the highest accuracy, precision and F1-score also belong to the *k-NN*. The *k-NN* can be one of the significant classifiers in Russia dataset.

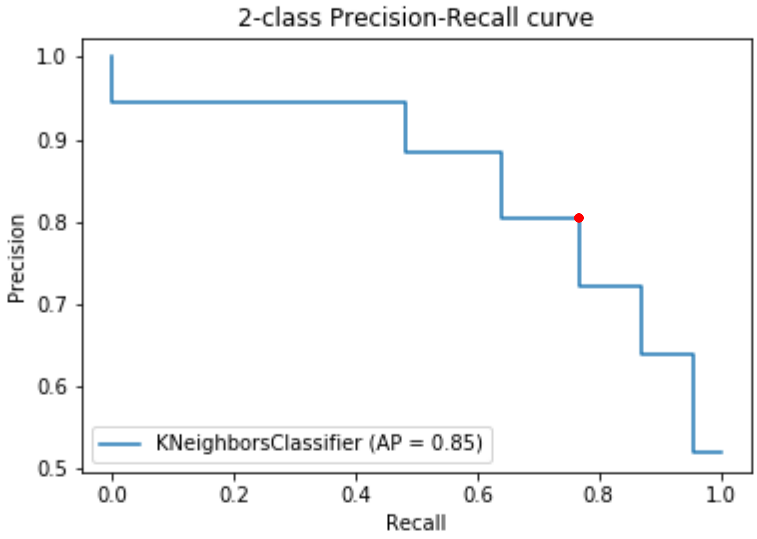


Plot 6 Performance of k-NN under every combination (Russia)

Combination	**Index**			
	Accuracy	**Recall**	**Precision**	**F1-Score**
All	0.78	0.77	0.81	0.79
k=5	0.77	0.76	0.79	0.78
k=4	0.77	0.76	0.79	0.77
k=3	0.73	0.72	0.76	0.74
k=2	0.72	0.69	0.75	0.72
L1-based	0.78	0.77	0.8	0.79
Tree-based	0.77	0.79	0.78	0.78

Table 17 Indexes performance of k-NN classifier (Russia)

Plot 6 and table 17 are performances of the *k-NN* classifier. About indexes of performances, the combination with all features has the highest accuracy and precision. Moreover, the L1-based combination has the highest F1-score, and tree-based occupies recall. Although the combination with all features gets the highest scores in two indexes, in plot #, it illustrates that indexes of L1-based combination are not only higher, but minimal differences. Thus, the well-performed combination with the *k-NN* is L1-based combination.

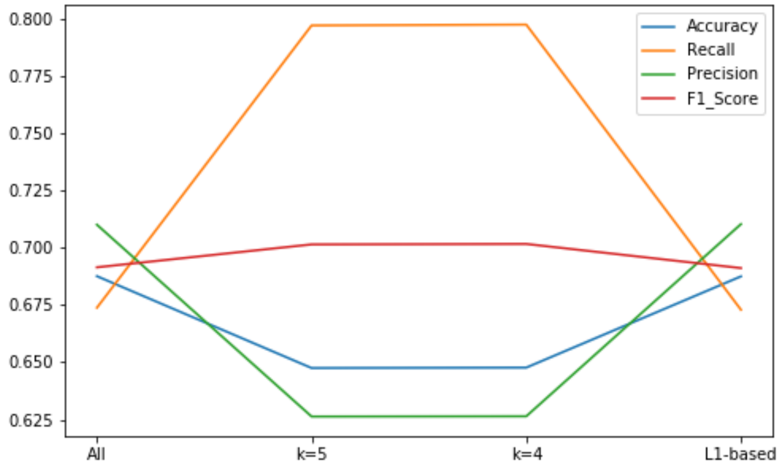


Plot 7 Precision-Recall curve of k-NN under the well-performed combination (Russia)

Plot 7 is the precision-recall curve of L1-based combination with *k-NN* classifier. The red dot on the curve is the result of our model (L1-based with *k-NN* classifier). About the precision-recall curve, the pattern is concaved and closes to the upper right angle. Thus, this model is a useful test. Moreover, the red dot is close to the steep side. It means this result is well.

SVM

According to the report presented in table 16, the *SVM* also can be one of the significant classifiers in Russia dataset. The combinations which select *SVM* as classifier are all, k=5, k=4 and L1-based. Plot 7 illustrates the performances of these combinations.



Plot 8 Performance of SVM under every combination (Russia)

In plot 8, all and L1-based combinations have similar results, then k=5 and k=4 have similar ones. Both of these two groups of combinations have a difference on the same feature, "repeated hashtag". In other words, the tweet has repeated hashtag or not only cause a slight effect on the results of the *SVM* in Russia dataset. Compare to L1-based and k=4 combinations which are the ones without less important feature "repeated hashtag" in two groups, we can figure out that the L1-based combination has "polarity number", but k=4 does not have. Thus, the primary reason that the performance difference between these two groups is "polarity number". If considering "polarity number" as a feature, recall and F1-score decreases, but precision and accuracy rises. Moreover, values of indexes become more concentrated.
Table 18 is the confusion matrix of k=4 and L1-based combination.

Combination	**Confusion Matrix**			
	TP	***FP***	***TN***	***FN***
k=4	4487	2676	2529	1141
L1-based	3787	1545	3660	1841

Table 18 Confusion matrix of SVM with k=4 and L1-based combinations (Russia)

Table 18 points out the combination k=4, as known as combination without "polarity number", has more positive prediction than the L1-based combination. That is the main reason for the recall values of k=4 combination is higher than L1-based.

5.3.5.2 By combination performances

The results of model selection by combination performances are displayed in table 19.

Classifier	**Combination**						
	k=6	**k=5**	**k=4**	**k=3**	**k=2**	**L1-based**	**Tree-based**
k-NN	v	v	v			v	v
GNB			v			v	v
Decision Tree	v	v	v			v	v
SVM	v	v	v			v	v
NN	v	v	v			v	v
LR	v	v	v			v	v

SGD	v	v	v			v	v

Table 19 Model selection by combination performances (Russia)

Table 19 shows that combination with k=2 and k=3 do not be selected by any classifiers. Moreover, the rest of the combinations are selected by almost all of the classifiers. Consider to features in each combination, features in k=3 and k=2 combinations make less help on distinguishing a tweet is from a suspended or unsuspended account. In other words, "day", "hour" and "minute" may not help classifiers to recognise accounts status.
According to section a), the *NN*, *LR* and *SGD* do not perform well in any combinations. Therefore, we discuss classifier performances under k=4, L1-based and tree-based combinations.

K=4, L1-based and tree-based combination
First of all, there is a table which can show the performance of three combinations under four classifiers *k-NN*, *GNB*, *Decision Tree* and *SVM*.

Classifier	**Combination**	**Index**			
		Accuracy	**Recall**	**Precision**	**F1-Score**
k-NN	k=4	0.77	0.76	0.79	0.77
	L1-based	0.78	0.77	0.8	0,79
	Tree-based	0.77	0.79	0.78	0.78
GNB	k=4	0.66	0.69	0.66	0.68
	L1-based	0.67	9,73	0.66	0.7
	Tree-based	0.66	0.69	0.66	0.68
Decision Tree	k=4	0.71	0.59	0.8	0.68
	L1-based	0.7	0.63	0.75	0.68
	Tree-based	0.71	0.59	0.8	0.68
SVM	k=4	0.65	0.8	0.63	0.7
	L1-based	0.69	0.67	0.71	0.69
	Tree-based	0.64	0.81	0.62	0.7

Table 20 Performances of k=4, L1-based and tree-based combination (Russia)

In table 20, the k=4 and tree-based combinations have similar patterns. In *Decision Tree* classifier, three combinations have similar results on accuracy and F1-Score, but the recall and precision are different between L1-based and the other two combinations. The L1-based one is more balanced; it means the gap between its recall and precision is smaller than other combinations.

When looking to the performance among different classifiers, the well-performed one is the *k-NN* classifier, and second to fourth ones are the *Decision Tree*, *SVM* and *GNB*. However, when looking to the cross-validation results in table 21, it points out that the *Decision Tree* classifier is unstable with these three combinations, despite the lowest mean of scores and highest standard deviation.

Furthermore, the most stable model is L1-based combination with the *SVM* classifier. Although the mean of scores of this model is not the highest, the gap between the highest one, 0.65, is only 0.01. The model of L1-based combination with the *SVM* classifier can be counted as one of the well-performed classifiers.

Classifier	Combination	Cross-Validation scores	
		Mean	Standard deviation
k-NN	k=4	0.63	0.085
	L1-based	0.65	0.08
	Tree-based	0.62	0.075
GNB	k=4	0.61	0.055
	L1-based	0.61	0.065
	Tree-based	0.61	0.06
Decision Tree	k=4	0.57	0.095
	L1-based	0.59	0.085
	Tree-based	0.57	0.095
SVM	k=4	0.6	0.065
	L1-based	0.64	0.04
	Tree-based	0.58	0.08

Table 21 Cross-validation scores of k=4, L1-based and tree-based combination (Russia)

5.3.5.3 Comparison

Compare to results in the last two section. There are two well-performed models selected. One is L1-based combination with *SVM*, and the other is L1-based combination with the *k-NN* classifier. Table 22 is the cross-validation score of two selected models.

Models		Cross-Validation	
Classifier	Combination	Mean of scores	Standard deviation of scores
k-NN	L1-based	0.65	0.08
SVM	L1-based	0.64	0.04

Table 22 Cross-validation results of well-performed models in each section (Russia)

According to table 22, the L1-based combination is more stable with the *SVM* classifier, but the accuracy is higher when with *k-NN* classifier. Consider the confidence interval of results; the *k-NN* model has a lower percentage of confidence. Thus, the well-performed model in Russia dataset is L1-based combination with the *SVM* classifier.

6 DISCUSSION

6.1 Overview

Two parts are discussed in this section. One is about data collection, and the other part talks about further research.

6.2 Limitation of Data Collection

In this research, despite Twitter's policy, the data of unsuspended accounts might concentrate on the same week or have a similar topic and text. These possibilities may cause output slightly inaccurate. Moreover, because unsuspended tweets are released in these days, although we assume those tweets do not be suspended, they may have possible to be suspended in future. The topic analysis is one of the approaches to get the data of unsuspended accounts; there may have other approaches to decrease over-similarity of tweets.

6.3 Suggestion for further research

In this research, we only list the results of well-performed models. In some models, accuracy and stability are not high enough. In further research, accuracy and stability can be improved by selecting different features or using other methods to collect data.

7 CONCLUSION

7.1 Overview

In the conclusion section, we divide them into three parts. One is a summary of Chinese results, and second is a summary of Russian topic. Finally, a comparison of two summaries is described. First of all, the central question of this research is "How to detect a tweet comes from a suspended or an unsuspended account?". Answers include in the following two sections.

7.2 Chinese Topic

According to our results, the main topics of China suspended accounts are "milesguo+wengui", "milesguo+china" and "world+epidemic". In the sentimental analysis, the average polarity score of China dataset is 0.58002, and most of the statements in this dataset are neural. Moreover, the average subjectivity score is 0.31868, and most statements are objective. With frequent pattern mining, "郭文贵" is the frequent hashtag. It has a 0.11 frequency.

In feature selection, there are three approaches, K best, L1-based and tree-based. With k best feature selection, the importance of features from highest to lowest are "hour" "frequent hashtags", "repeated hashtags", "day", "minute", "subjectivity number" and "polarity number". In L1-based selection, since every feature correlates with a decision of suspension, all features are selected. In tree-based selection, the features with high importance are "polarity number", "subjectivity number", "hour" and "minute".

The central part of this research is classifier and analysis. After analysis and comparison, the well-performed model in China dataset is the combination of all features using the *k-NN* classifier. The confusion matrix of this model: 5942 (67%) data is *TP*, 971 (11%) is *FP*, 1327 (15%) is *TN*, and 619 (7%) is *FN*. In other words, the accuracy of this model is 82%. Furthermore, when counting the outputs of the confusion matrix, we can get recall is 91%, precision is 86%, and F1-score is 88%. Moreover, the combination with "hour" and "frequent hashtags" has outstanding performance in the *Decision Tree* classifier.

For answering the main question, we can detect a tweet come from a suspended or an unsuspended account by the *k-NN* classifier. Seven things can assist in our prediction; First, the polarity and subjectivity scores of the text. Second, the existence of repeated hashtags and the hashtag "郭文贵". Then, the day, hour and minute of the tweet.

7.3 Russian Topic

The three main topics of Russia suspended accounts are "people+football", "girl+naked" and "followers+stats". In the sentimental analysis, the average polarity scores are 0.09174. Moreover, most tweets content are positive. About the subjectivity scores, the average is 0.35078; most of the statements are objective. The frequent pattern mining of hashtags displays that there are no apparent patterns exist in hashtags of the Russia dataset.

In feature selection part, the features sort by the importance from k best approach is "hour", "day", "minute", "subjectivity number", "repeated hashtags" and "polarity number". In the L1-

based approach, the only unselected feature is "repeated hashtags". Moreover, with the result of the tree-based method, the essential features are "subjectivity number", "day" and "hour". The well-performed model in Russia dataset is L1-based combination with the *SVM* classifier. The mean accuracy score is 0.64, and the standard deviation is 0.04. For answering the main question, we can detect sources of a tweet by the *SVM* classifier and putting five things into the selected classifier—the polarity and subjectivity scores of texts, and the day, hour and minute of posting time.

7.4 Comparison

According to the final results of the last two sections, there are some different and similar parts. The similar part is that both of their outstanding models have features—"polarity number", "subjectivity number", "day", "hour" and "minute". There are three different parts in two results. First, Russian dataset does not have a frequent hashtag. Second, they use different classifiers on the well-performed model. Last but not least, the feature combination of Chinese topic's dataset uses more features than Russian dataset on the well-performed model.

8 References

Celliers, M., & Hattingh, M. (2020). A Systematic Review on Fake News Themes Reported in Literature. *Conference on e-Business, e-Services and e-Society* (pp. 223-234). Pretoria, South Africa: Springer, Cham.

Conroy, N. J., Rubin, V. L., & Chen, Y. (2015). *Automatic Deception Detection: Methods for Finding Fake News.* London, Ontario, CANADA: Language and Information Technology Research Lab (LIT.RL).

Decision tree learning. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Decision_tree_learning

Figueira, Á., & Oliveira, L. (2017). The current state of fake news: challenges and opportunities. *International Conference on ENTERprise Information Systems* (pp. 817-825). Barcelona, Spain: Procedia Computer Science.

Gandhi, R. (2018, 6 7). *Support Vector Machine — Introduction to Machine Learning Algorithms*. Retrieved from Medium: https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47

Gilda, S. (2017). Evaluating Machine Learning Algorithms for Fake News Detection. *2017 IEEE 15th Student Conference on Research and Development (SCOReD)* (pp. 110-115). Pune, India: IEEE.

Granik, M., & Mesyura, V. (2017). Fake News Detection Using Naive Bayes Classifier. *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)* (pp. 900-903). Kiev, Ukraine: IEEE.

k-nearest neighbors algorithm. (n.d.). Retrieved from Wikipedia: https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

Long, Y., Lu, Q., Xiang, R., Li, M., & Huang, C.-R. (2017). Fake News Detection Through Multi-Perspective Speaker Profiles. *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)* (pp. 252-256). Taipei, Taiwan: Asian Federation of Natural Language Processing.

Marco L. Della Vedova, Eugenio Tacchini, Stefano Moret, Gabriele Ballarin, Massimo DiPierro, & Luca de Alfaro. (2018). Signals, Automatic Online Fake News Detection Combining Content and Social. *2018 22nd Conference of Open Innovations Association (FRUCT)* (pp. 272-279). Jyvaskyla, Finland: IEEE.

Naive Bayes. (n.d.). Retrieved from Sckit learn: https://scikit-learn.org/stable/modules/naive_bayes.html

Roy, R. (2020, 05 16). *ML | Stochastic Gradient Descent (SGD)*. Retrieved from GeeksforGeeks: https://www.geeksforgeeks.org/ml-stochastic-gradient-descent-sgd/

Shaikh, R. (2018, 10 28). *Feature Selection Techniques in Machine Learning with Python*. Retrieved from Medium: https://towardsdatascience.com/feature-selection-techniques-in-machine-learning-with-python-f24e7da3f36e

Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017, August 7). *Fake News Detection on Social Media: A Data Mining Perspective.* Retrieved from SIGKDD Explorations: https://www.kdd.org/

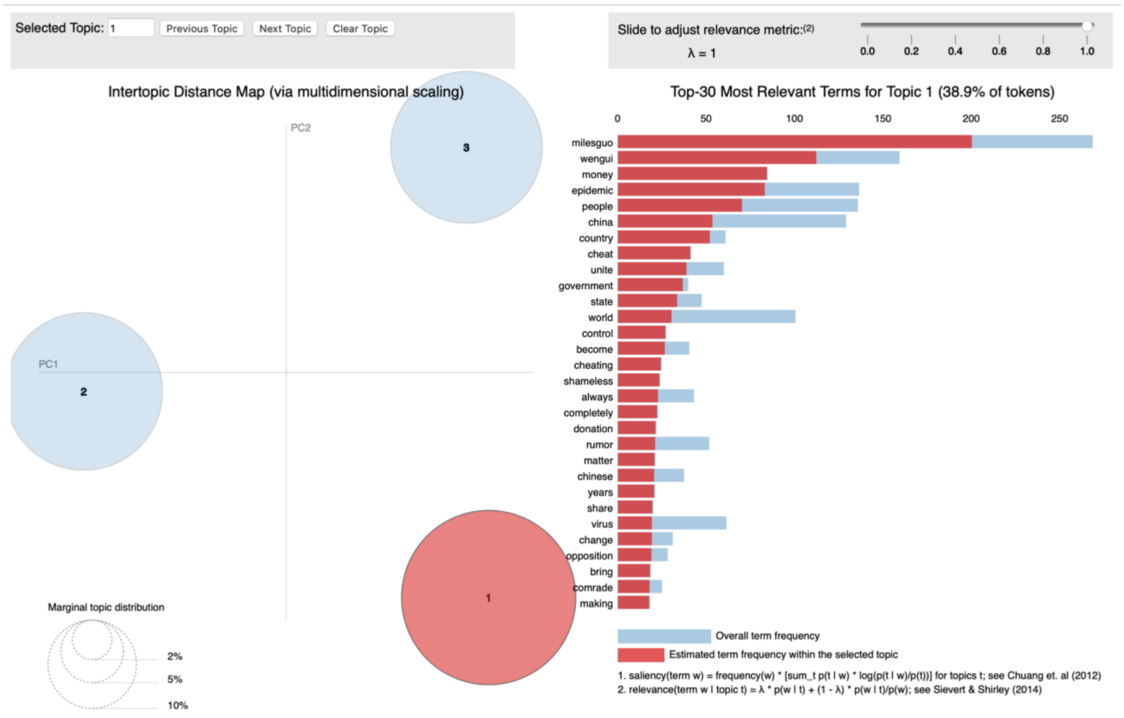
Shu, K., Wang, S., & Liu, H. (2018). Understanding User Profiles on Social Media for Fake News Detection. *2018 IEEE Conference on Multimedia Information Processing and Retrieval* (pp. 430-435). Miami, FL, USA: IEEE.

Wang, W. Y. (2017). *"Liar, Liar Pants on Fire": A New Benchmark Dataset for Fake News Detection.* Santa Barbara: Department of Computer Science University of California.

Yang, S., Shu, K., Wang, S., Gu, R., Wu, F., & Liu, H. (2019). Unsupervised Fake News Detection on Social Media: A Generative Approach. *The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)* (pp. 5644-5651). Hawaii, USA: Association for the Advancement of Artificial Intelligence.

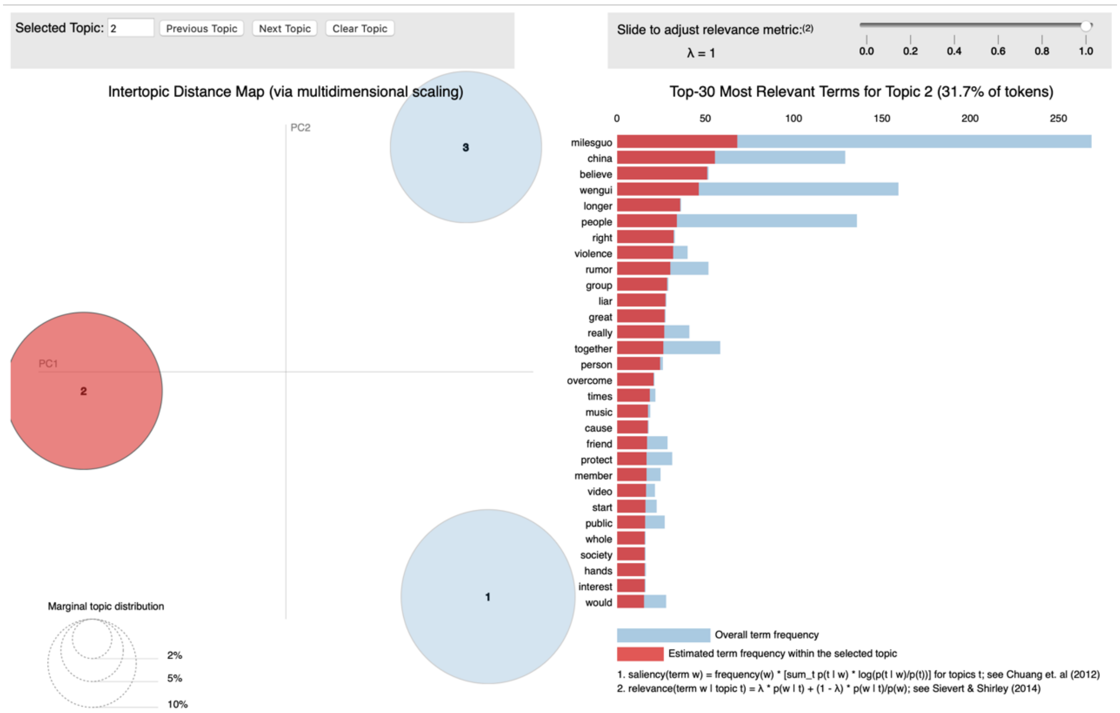
9 APPENDICES

9.1 Appendix: Results of Topic Analysis in Chinese Datasets



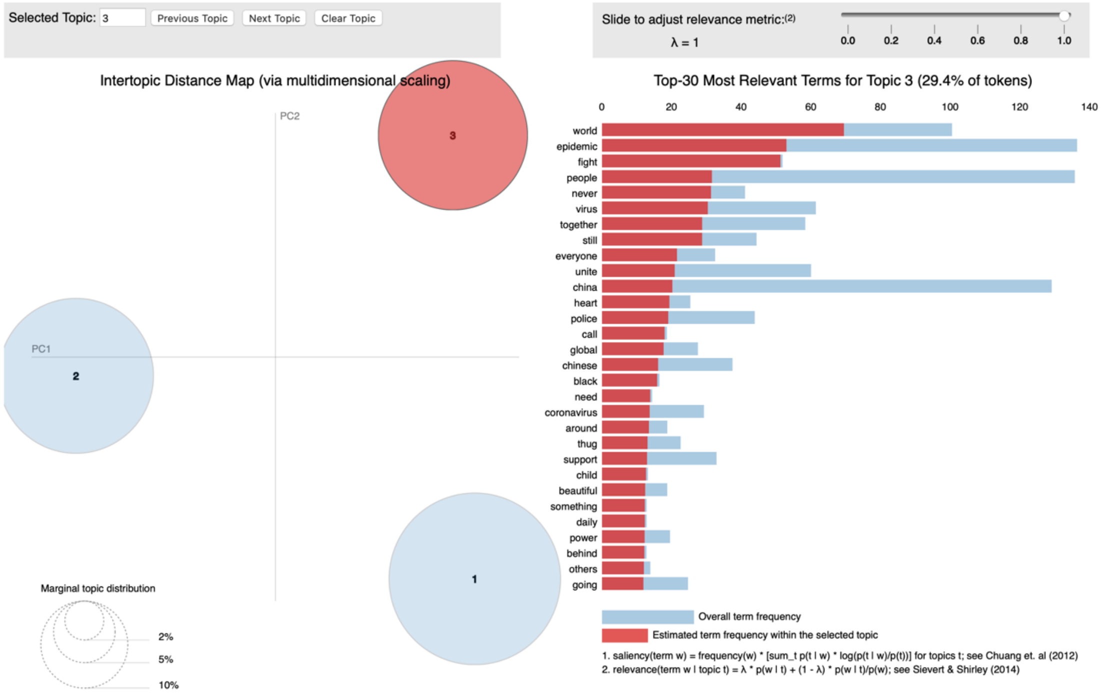
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)

2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)


Selected Topic: 2

λ = 1

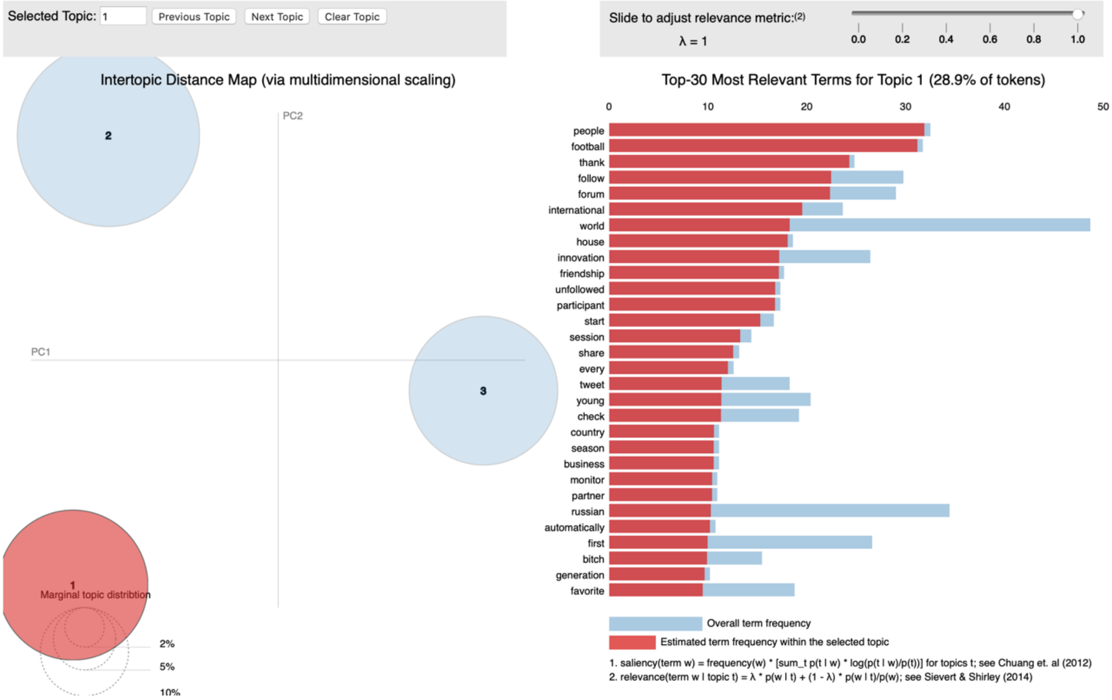
1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)


Selected Topic: 3

λ = 1

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)

9.2 Appendix: Results of Topic Analysis in Russian Datasets



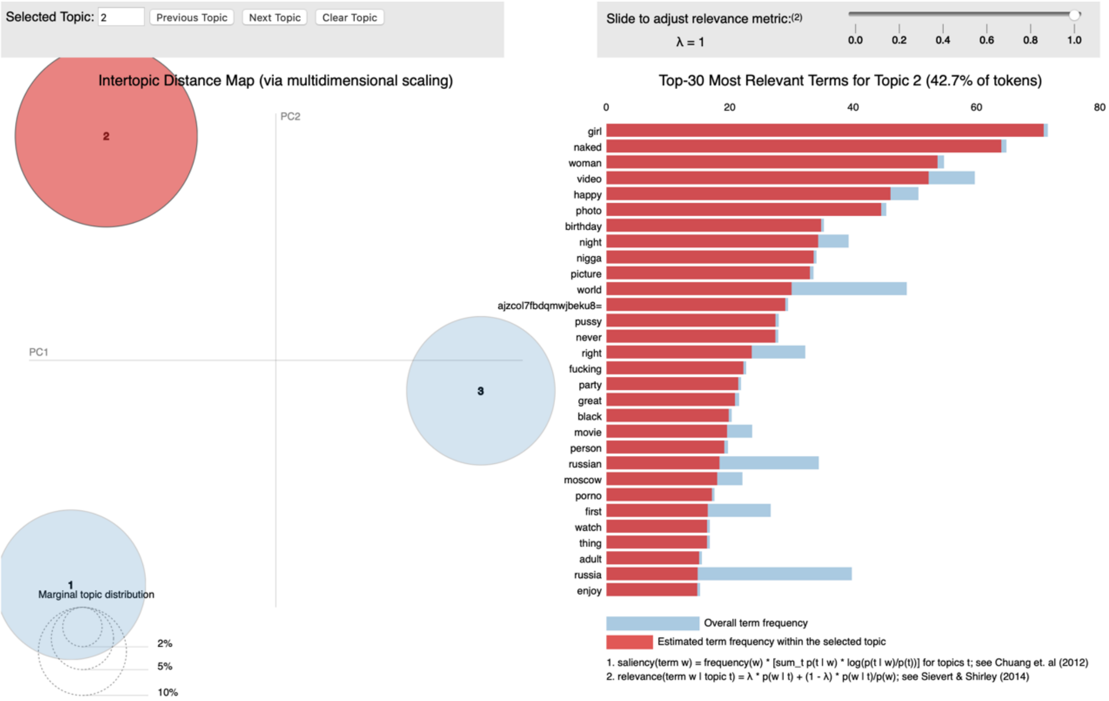
Selected Topic: 1

Slide to adjust relevance metric:(2)

λ = 1

Top-30 Most Relevant Terms for Topic 1 (28.9% of tokens)

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)



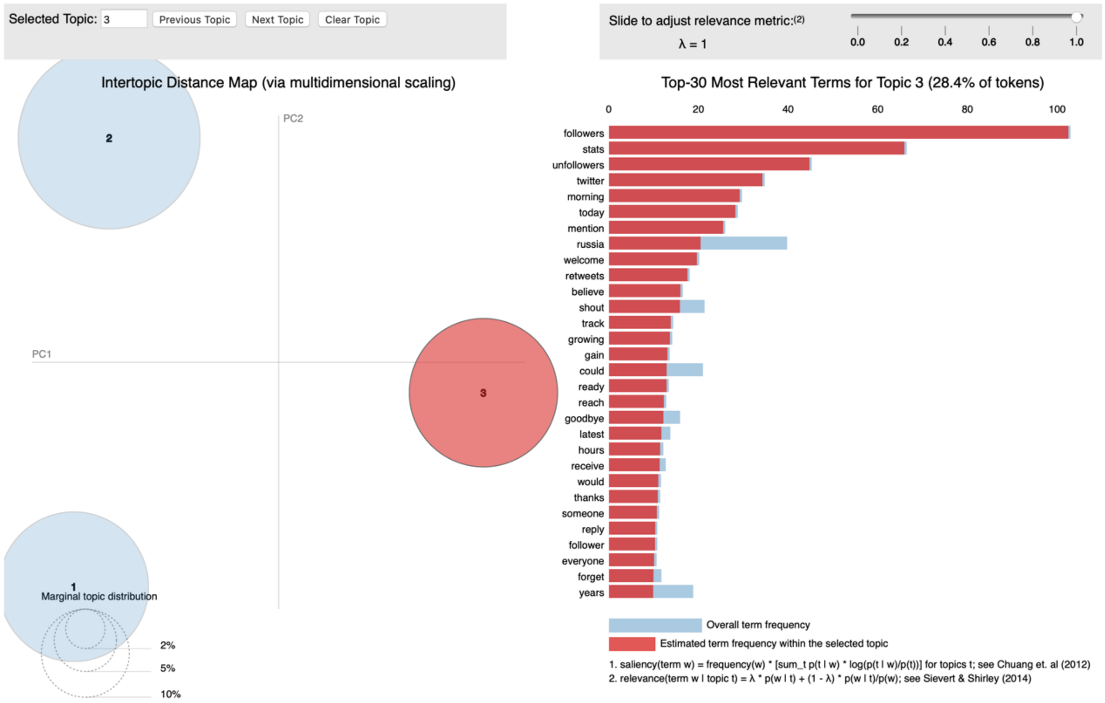
Selected Topic: 2

Slide to adjust relevance metric:(2)

λ = 1

Top-30 Most Relevant Terms for Topic 2 (42.7% of tokens)

1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)



1. saliency(term w) = frequency(w) * [sum_t p(t | w) * log(p(t | w)/p(t))] for topics t; see Chuang et. al (2012)
2. relevance(term w | topic t) = λ * p(w | t) + (1 - λ) * p(w | t)/p(w); see Sievert & Shirley (2014)