Contents

FINAL PROJECT 2

Introduction: 2

EDA: 3

Method: 6

Result: 7

Outlook: 8

Reference: 8

FINAL PROJECT

Name: Zhujun Shen (140297885)

Title: The Best Platform to Post News

Class: DATA 1030

GitHub: https://github.com/zshen15/data1030_project.git

Introduction:

Social media is one of the best ways to communicate with others and has become a big part of our daily life. By 2019, there are 2.45 billion monthly active users on Facebook and 1 billion on Instagram. I personally don't use Facebook or Instagram as much. The lack of followers results into a small number of comments on all of my posts. Instead, I prefer Wechat where I can gain the highest potential feedback. Whenever I want to post something, I always choose when it is daytime in China. In order to gain more attentions, I care about the following three questions: which platform to post; what to post; when to post.

Same with the news. A successful news is the one that can get the most readers and/or comments. How many users are there on this platform? What time of the day will people read news? Which topic do the users most interested in? This research aims to predict the best platform (among Facebook, GooglePlus, LinkedIn) for posting a certain type of news (among 'Obama', 'Microsoft', 'Economy', 'Palestine') on a given day.

The dataset is retrieved from UCI which includes 93239 observations with 11 features. The target variable will be a class indicates the best platform to post a news. Hence, the project will try to solve a classification problem.

The following will be a table for data description:

	Variable Name	Type	Detailed Description
Features	IDLink	real	Index
	Title	str	title of the news
	Headline	str	headline of the news
	Source	str	source of the news
	Topic	str	'Obama', 'Microsoft','Economy', 'Palestine'
	PublishDate	time	the post time and date for the news
	SentimentTitle	real	sentiment score for the title
	SentimentHeadline	real	sentiment score for the headline
Target Variable	Facebook	real	popularity for each platform
	GooglePlus		
	LinkedIn		

EDA:

The dataset contains no missing values. See the following table for preprocessing:

	Variable Name	Preprocess	Reason
Features	IDLink	drop	not related to the target variable
	Title	drop	same with "SentimentTitle"
	Headline	drop	same with "SentimentHeadline"
	Source	drop	not related to the target variable
	Topic	OneHotEncoder	four parallel categories
	PublishDate	Unix time	under format "yy-mm-dd hh-mm-ss"
	SentimentTitle	MinMaxScaler	range (-1, 1)
	SentimentHeadline	MinMaxScaler	range (-1, 1)
Target Variable	Facebook	Target Variable	"BestPlat" *
	GooglePlus		
	LinkedIn		

* Construct a target variable called "BestPlat". The detailed description can be found below: (F = "Facebook", G = "GooglePlus", L = "LinkedIn")

- BestPlat = 0:
  When F = G = L = -1, it means that the news is not posted on either of the platforms.
- BestPlat = 1:
  When F is the largest, it means that posting on Facebook have the largest popularity.
- BestPlat = 2:
  When G is the largest, it means that posting on GooglePlus have the largest popularity.
- BestPlat = 3:
  When L is the largest, it means that posting on LinkedIn have the largest popularity.
- BestPlat = 4:
  When F = G = L = 0, it means that the news gains 0 popularity on all three platforms. In another word, it is indifferent to post on any platforms.
- BestPlat = 5:
  Other cases. For example, F = G > L

When BestPlat = 0 (Not post), the news is not posted. It is not related to the proposal of this project. In Figure 1 we can see that, there are only 6.2% of the total dataset for BestPlat = 5 (Others) and can be neglected for the sake of convenience. Thus, we drop all the observations when BestPlat equals to 0 and 5. The resulting distribution for BestPlat is Figure 2. After all the preprocess, the dataset now has 82998 observations with 7 features and 1 target variable.

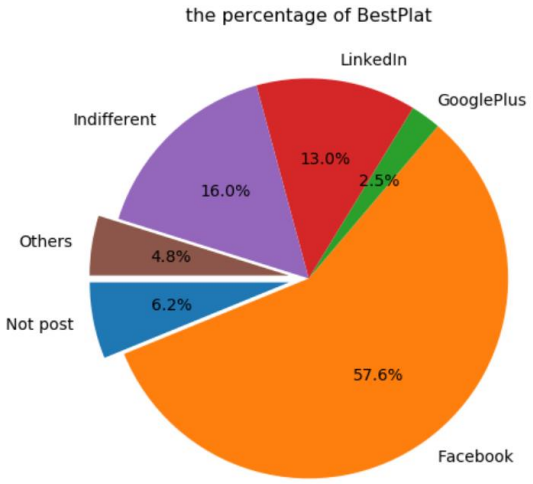


Figure 1: pie chart for BestFlat

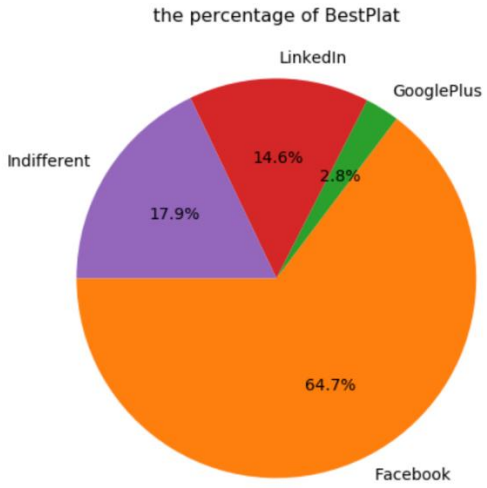
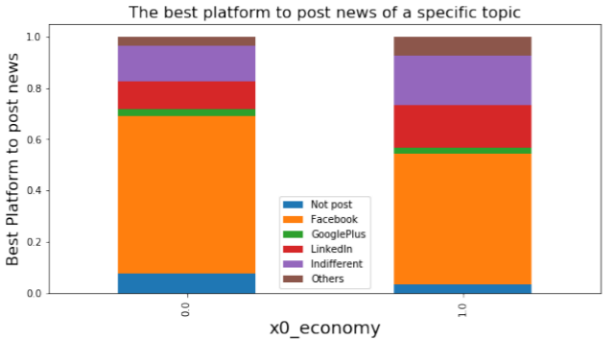
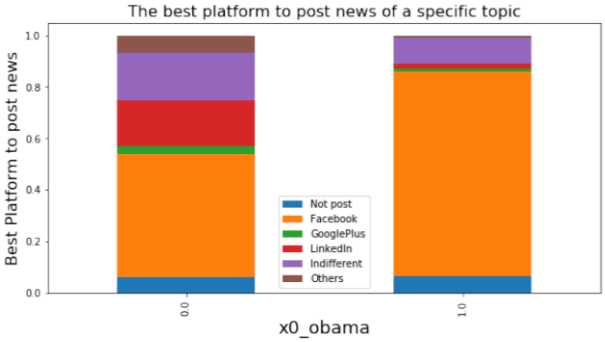
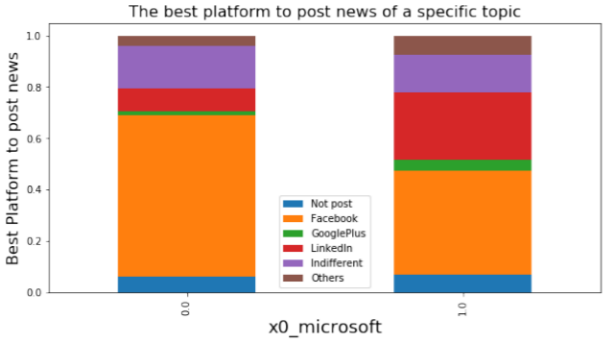


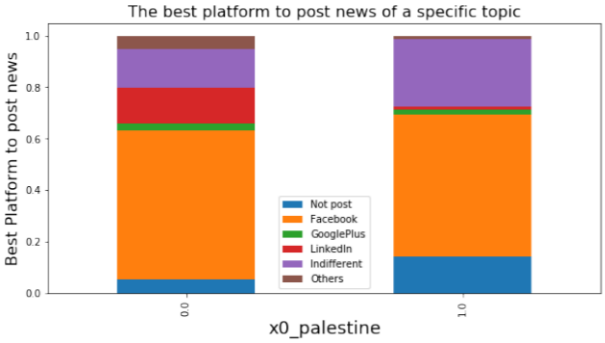
Figure 2: pie chart for BestFlat (drop 0 and 5)

The following plots are the bar plot for each platform given a specific topic. In each plots, we can see the distribution of all values of BestPlat under different topics. To be more specific, in “x0_economy”, it shows that P(choose Facebook as the best platform | topic = economy) < P(choose Facebook as the best platform | topic != economy). Interestingly, compare the four graphs, the orange bar only increaes form left to right in “x0_obama”. It is reasonable to say that the users on Facebook have a greater interest in news about obama in general.

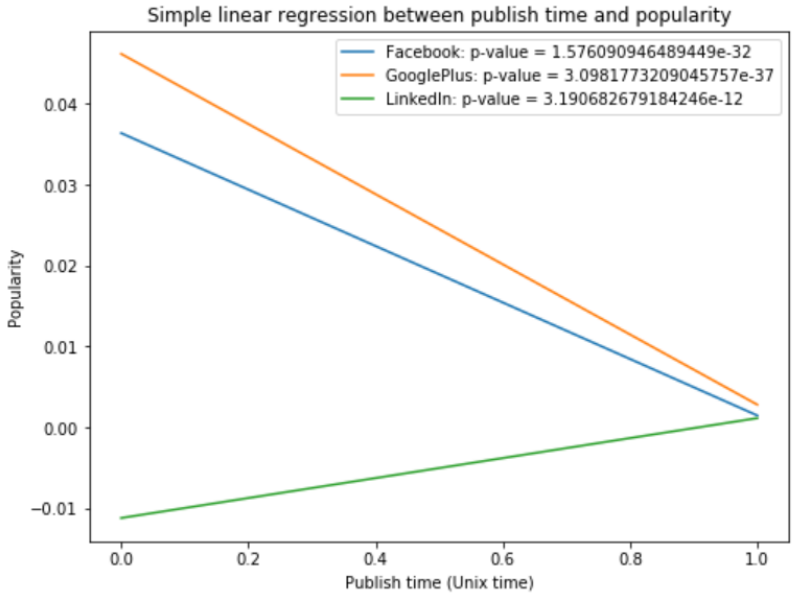








Then, if we run a simple linear regression of PublishDate (x) and the popularity for each platform, we can get Figure4. Since all p-values are greatly less than 0.05, we can conslude that platform popularity is related to publish date to some extend. Moreover, only LinkedIn has a positive relationship with publish time.



Method:

My ML pipeline can be devided into four steps:

1. Cross validation: We first devide out data into testing data and other data. Then we randomly dividing the other data into k groups of approximately equal size by using Stratified K-fold cross validation. We then take (k-1) groups of data and call them training data. The left over group willl be our validation data. That is to say, after k-fold, we can have k pairs of training and validation data.
2. Train: For each pair of training and validation data, we apply our machine learning model with different parameters. By testing the accuracy score, we can find one best parameter as well as its correponding accuracy score. Accuracy score stands for the fraction of predictions our model got right.
3. Multiple trail: Then we can change our random state for different cross validation data. Repeat step 1 and steo 2 n times.
4. Average accuracy score: After step 2, we get k different accuracy score. After step 3, we get n trail. Then we can construct a list contain all k * n accuracy score. Calculate their mean and standard deviation for our final answer. Compare it with the baseline.

In the second step of my ML pipeline, I need to find appropriate models for this project. Since it is a classification problem, I choose to use Logistic Regression, Random Forest Classifier and K Nearest Neighbor. SVC is also good for classification problem. However, since the dataset has larger than 80 thousand observation, SVC is not as efficient as others.

For Logistic Regression, I set “C” = np.logspace(-3,3,10) and use 5-fold wich 3 loops (3 different random state). The mean and standard deviation when penalty is “l1” is 0.656474 +/- 0.000838 and it is 0.656382 +/- 0.000763 for “l2” penalty. Both process took about 688 seconds to finish. Then, I squared all features and run the same model again. Except, I used “C” = 1e-05, penalty='l1', 5-fold and 1 loop. It took 2599 second to finish this process while the accuracy socre dropped to 0.654663 +/- 0.000471. Thus, I quited trying more runs.

For Random Forest classification, I have two parameter “max_depths” and “min_samples_splits” where max_depths = list(range(2,15,3)) and min_samples_splits = [0.05, 0.1, 0.15, 0.2, 0.25]. Then after 5-fold, 3 loops (3 different random state), it took 1116 seconds to get the accuracy score 0.670651 +/- 0.000601.

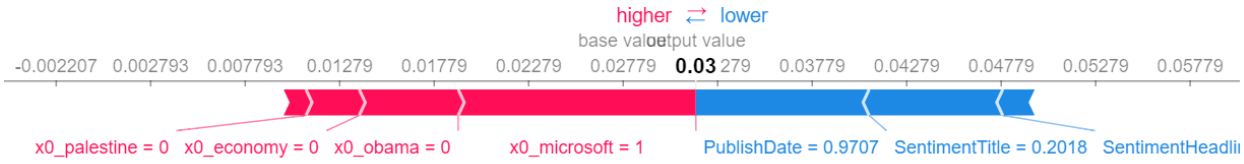
The last model I tried is K Nearest Neighbor. I used kn = [1,10,30,100,300,500,1000], weights="distance", 5-fold and 3 loops (3 different random state). The accuracy score 0.659494 +/- 0.001036 shown up after 807 seconds.

Result:

Model Name	Parameter	k-fold	Random state	Accuracy score	Time (unit time)
LR	C = logspace(-3,3,10) penalty='l1'	5	3	0.656474 +/- 0.000838	688 (15)
	C = logspace(-3,3,10) penalty='l2'	5	3	0.656382 +/- 0.000763	688 (15)
LR (squared)	C = 1e-05 penalty='l1'	5	1	0.654663 +/- 0.000471	2599 (520)
RF	max_depths = [2, 5, 8] min_samples_splits = [0.05, 0.1, 0.15, 0.2, 0.25]	5	3	0.670651 +/- 0.000601	1116 (5)
KNN	kn = [1,10,30,100,300,500] weights="distance"	5	3	0.659494 +/- 0.001036	807 (10)

Our final task is to compare all accuracy score with out baseline. Recall the data in Figure2, 64.7% percent of the data has a value of 1 for Bestplat. In another word, if we assume Facebook is the platform that can give us the highest popularity for every single news, there is 64.7% chance that out guess is correct. A successful model should have an accuracy score at least higher than 64.7%. From the above result we can tell that Logistic Regression and KNN have accuracy scores around 0.66. The improvement is either less than 1% or slightly higher than 1%. My best model, Random Forest, has an accuracy score of 0.67. It is only 2 percentage point higher than the baseline. Moreover, it is the model that process the fastest. Although neither of my model did a significant improvmet, the standard deviation turned out to be very small (close to zero). It shows the stability of all of my models.

For model inspection, I chose to use shap value method. SHAP takes variables' interactions with others into account when measuring their impact. From the graph below we can see that the variable with the highest impact is whether or not it is a news about microsoft. However, we have to notice that even the highest number is around 0.03, which is extremely low. That is too say, the impact is really small. This leads to a question: in real life, what factors are the ones that really impact the popularity of one news?



Outlook:

There are more things I can do in the feature to improve the process. For example, I can use more models such as neural network. Even in the models I have tried already, I can use more parameters such as the value for “C” in Logistic Regression and the value for “kn” in K Nearest Neighbor. What’s more, in the current dataset, the feature I have can be summarized as publish date, topic, title and headline. There are way more factors that may affect the popularity of one news, such as politics activity, stock market, etc. Having too little features is the largest challenge in my project.

Generally, after the model has been developed, we can add more news type as well as other platforms to generalize the result. The model can be used for political activities or business publicity. We can also find several researches online that use the same data. The data can be used to research on topic detection, sentiment analysis and first story detection.

Reference:

https://archive.ics.uci.edu/ml/datasets/News+Popularity+in+Multiple+Social+Media+Platforms

https://www.statista.com/statistics/264810/number-of-monthly-active-facebook-users-worldwide/