

Python Based Web Data Spider Douyin Magnitude of Followers Prediction Model

Binbin Xue^{1*}, Lingyi Zhao¹

¹ School of Economics and Management, Communication University of China, Beijing, China

Email Address

bineahsueh@cuc.edu.cn (Binbin Xue)

*Correspondence: bineahsueh@cuc.edu.cn

Received: 24 July 2023; **Accepted:** 12 August 2023; **Published:** 13 September 2023

Abstract:

This paper mainly uses Selenium, Pandas, and Sklearn modules in Python to capture, analyze, and train the datas of 12 bloggers' followers, the Quantity of "Likes", "Comments", and "Collect" of their videos in Douyin, which are affiliated with the MCN agency of Joy Media. The model is used to predict the possible magnitude of followers of the bloggers who are not famous after they become famous with the help of MCN agencies, and to estimate the future commercial value of the bloggers. The model predicts the following of "10 million followers", "million followers", "100,000 followers" and "10,000 followers". A total of 12 bloggers with 3 bloggers in each of the four levels were trained by data mining, data integration, outlier processing, and decision tree samples, and 30% of the samples were divided into prediction samples. The final accuracy rate was about 80.77% which is good enough. However, no explanation was given for factors such as when the amount of bloggers' followers reached a steady state.

Keywords:

Python, Selenium, Big Data Analysis, Spider, Predictive Model

1. Introduction

In recent years, the short video industry has been advancing rapidly, and the traditional media industry is gradually transforming to the development of new media industry. Jitteritter, fast hand as the head of the short video platform has a huge volume of user groups, and the large-scale data generated by the platform has a very high research value. Currently, the world is the world of traffic, the commercial value of Jittery account is generally directly linked to the amount of fans, and the commercial value of the amount of fans is also high. For MCN organizations, the cost of signing a blogger is also positively correlated with the amount of fans, and it's ideal to sign a blogger at a low amount of fans at a low cost in advance, and after signing a blogger the amount of fans is greatly improved and the commercial value of a large amount of money is greatly prompted, so it's important to study how to Therefore, it is meaningful for MCN organizations to study how to predict and explore bloggers with the potential to become big fans. [1,2]

2. Research Methodology

Most of the Jittery bloggers mainly rely on the works released to attract users' attention, and users often express their emotional tendency through the behaviors of "liking", "commenting" and "favoriting" on the works. Before users pay attention to the blogger, it is generally believed that users "like" and "favorite" a work, which means that users recognize the possibility of the work to become the blogger, so by training the "likes", "comments", "favorites" and "favorites" of a work before the amount of fans of the blogger stabilizes, we can find out how much users have liked the work and how much users have favored the work, and how many users have liked the work. Therefore, by training the three dimensions of "likes", "comments" and "favorites" of a work before the fan base of a ShakeEn blogger stabilizes, we can predict the approximate magnitude of the fan base when it stabilizes by inputting the predicted sample data into the model, e.g., the blogger may become a "10 million-fan blogger". "Ten million fans blogger", "million fans blogger", "100,000 fans blogger", "10,000 fans blogger" etc.

The model mainly studies and analyzes the influence of "likes", "comments" and "favorites" on the works of 12 bloggers under the MCN agency "Worry-free Media". The model analyzes the relationship between the "likes", "comments" and "favorites" of the works of 12 bloggers under the MCN agency "Worry-free Media", and utilizes the decision tree model in the Sklearn module of Python for training and prediction.

3. Hypotheses

By default, it is assumed that the number of followers of a blogger is larger when the number of followers is stabilized than before stabilization. It is assumed by default that the number of "likes", "comments", and "favorites" of a work is higher when stabilizing than before. The default assumption is that "fans", "likes", "comments" and "favorites" will not change during the data collection process. will not change during the data collection process. It is assumed by default that the number of followers of a blogger has stabilized at the time of data collection, i.e., the blogger has reached the maximum number of followers and will not undergo any major changes. [3]

4. Sample Data Sources

4.1. Crawler file construction details

By parsing Douyin web page version of the web page code found that its web page contains async asynchronous loading, so the main use of Python and Web automation testing tools Selenium simulation browser to crawl the data, mainly through the XPATH for the mined data positioning.

(1) In the file program, enter the Jitterbug number to be mined, locate the search box of the homepage of the web version of Jitterbug through XPATH as a form form using the 'zyznJl4l' Class attribute, send the Jitterbug number into the search box and enter and click on the hyperlink of the first search result to enter the homepage of the individual, the Jitterbug number has a unique characteristic, so the default first search results for the user to be collected.

(2) Enter the personal homepage to accurately locate and collect the user name, Jitterbit number, and number of fans, because there is no information such as the

number of comments and collection in the works in the personal homepage, it is also necessary to collect the independent URL links of each work and enter them one by one to collect the information, and the independent URLs of the works in the Jitterbit works are composed of the following: “Jitterbit domain name/video/unique identification number”. [4]

(3) After obtaining the URL, the program uses a for loop to enter one by one to locate and collect information about the title, URL, likes, comments, favorites and release time of the work and store it. After all the works data are collected, use the pandas module to store them.

4.2. Crawler file crawling process details

Using the crawler file on the “10,000 + level”, “100,000 + level”, “1,000,000 + level”, “10,000,000 + level” at all levels of the 3 a total of 12 bloggers for data crawling, and record the relevant information as well as the time of crawling, a total of 1483 crawling works, a total of 1 hours, 58 minutes and 13 seconds, the data generated on May 6, 2022, detailed information as shown in Table 1.

Table 1. Information about the Jitterbug number crawled and when it was crawled.

Magnitude	UserName	UserId	Follower (10,000)	Number of works	Spend time
10 million	Liu Genghong	895373343	6230.8	248	15 min 30 s
	Golden Hair Land Rover	JMLH0606	1990.7	257	25 min 10 s
	Wang Naiying	19171163	1233.0	186	17 min 42 s
1 million	Zhang Xinyao	zxy19941019	939.4	119	9 min 11 s
	One Little Ear	xed1216	662.8	61	3 min 52 s
	Liu Hua Hua	944207538	195.9	108	8 min 26 s
100 thousand	Li Sanshou	SANSUI522	66.9	188	16 min 42 s
	Lin Hui	linhui623	87.1	60	5 min 49 s
	Little Nan Gua	nn21391646	52.7	55	3 min 25 s
10 thousand	Two Fish	22059456	3.3	89	6 min 45 s
	Lin Xiaoxiao	JLdada520	6.4	22	1 min 04 s
	Sweet and Spicy Ran	Ran9030	1.3	90	4 min 37 s

5. Data Cleansing

5.1. Data consolidation

Each account information is stored in a separate csv file after capture in the data analysis before the need to merge the 12 independent csv data files, the use of Python's Pandas library for merging, the core code is as follows:

```
data = pd.concat([df1,df2,df3,df4,df5,df6,df7,df8,df9,df10,df11,df12])
```

5.2. Data type view and preprocessing

The first five rows of data in the merged data set are shown in Table 2.

Table 2. *The first five rows of data in the dataset.*

Index	UserName	UserId	Follower	Title	URL	IssuedTime	Likes	Comments	Collect
1	Liu Genghong	895373343	62.308m	...	...	16/11/2020	1.403m	51k	30k
2	Liu Genghong	895373343	62.308m	...	...	14/7/2020	1.84m	80k	37k
3	Liu Genghong	895373343	62.308m	...	...	29/4/2020	1.038m	25k	16k
4	Liu Genghong	895373343	62.308m	...	...	6/5/2022	21k	3596	2185
5	Liu Genghong	895373343	62.308m	...	...	5/5/2022	88k	4801	5842

In the merged data set did not find the existence of null values, data columns exist in the unit character "m", so you need to determine whether the last character of each piece of data for the "m" if it is required to go to the "m" character and multiply it by 10,000, if not need to be dealt with, and to change the data columns attributes to "int64", the core code is as follows:

```
for indexs in range(1483):
    if str(data.loc[indexs, 'Follower'])[-1] == 'm':
        data.iloc[indexs, 3] = int(float(data.loc[indexs, 'Follower'][:-1])*100000)
```

5.3. Distribution of data

Most of the vlogger's "likes", "comments", "favorites" there are a large number of outliers, data distribution in the two parts of the outliers are fans tend to stabilize the data, the goal of the prediction model is to predict the blogger's fan volume tends to stabilize before the prediction of the fan volume needs to tend to stabilize the volume of the fan, that is, the outliers are excluded, which outliers are eliminated with the help of Excel for the elimination of the outliers. [5]

5.4. Adding a "level" dimension based on follow'er level

The "level" columns are "1" for millions, "2" for millions, "3" for hundreds of thousands, and "4" for tens of thousands, with the following core code:

```
for indexs in range(1483):
    if data.loc[indexs, 'Follower'] > 10000000:
        data.loc[indexs, 'level'] = 1
    elif data.loc[indexs, 'Follower'] > 1000000:
        data.loc[indexs, 'level'] = 2
    elif data.loc[indexs, 'Follower'] > 100000:
        data.loc[indexs, 'level'] = 3
    else:
```

```
data.loc[indices, 'level'] = 4
```

6. Decision Tree Sample Training and Prediction

Decision tree prediction using Sklearn module in Python:

(1) Set training samples X as columns “Likes”, “Comments”, “Collect”.

(2) Y is the “level” column.

(3) Decision tree samples are divided into 30%, min_samples_leaf parameter is 10, max_depth parameter is 10, resulting in a decision tree prediction accuracy of 80.77%, part of the code is as follows:

```
dtc = DecisionTreeClassifier(  
    min_samples_leaf=10,  
    max_depth=10  
)  
dtc.fit(X_train, y_train)  
  
print("The accuracy is: {:.2%}".format(dtc.score(X_test, y_test)))  
  
[0]Output: The accuracy is: 80.77%
```

7. Conclusions and Deficiencies

The final sample prediction accuracy reaches 80.77%, which is a good result, and it can more accurately analyze the number of “likes”, “comments” and “favorites” of the blogger’s works before the fan base stabilizes and predict the approximate size of the fan base after the stabilization. MCN organizations can use this model to estimate the fan volume of bloggers who are interested in signing a contract, and judge the fan volume that may be achieved after signing a contract with the support of the organization’s resources and other aspects, so as to explore the commercial value.

The model does not describe the characteristics of the blogger's fan base stabilization state, and does not accurately mine the work data before fan base stabilization, but defaults to the overall data anomalies as the work data during the stabilization state, and excludes the work data with anomalous values to obtain the pre-stabilization work data, which is an insufficiently rigorous method, and does not divide the work data from the time dimension because it is impossible to determine the critical point of time between the pre-stabilization and stabilization periods. It is not possible to accurately determine whether the blogger’s fan base has reached a stable state. In reality, the blogger's fan base may fluctuate greatly, and the fan base in the stable state may not be the highest value. The model can only roughly determine the amount of bloggers who may have reached a stable state, but not the approximate number of bloggers.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

Funding

This research received no specific grant from any funding agency in the public, commercial or not-for-profit sectors.

References

- [1] Chang, X. Research on Web Data Crawling and Analyzing Based on Python Crawling Techniques. *Digital Technology & Application*, 2017, 09, 35-36.
- [2] Guo, X.Y. Sina Weibo Data Access based on Python and Selenium. *Computer Programming Skills & Maintenance*, 2012, 15, 21-23.
- [3] Feng, Y.R. Python Language in Big Data Analytics. *Computer Knowledge and Technology*, 2020, 16(24), 72-73+80.
- [4] Xu, Y.F.; Xu, B. Python language features and its application in machine learning. *Computer Products & Distribution*, 2019, 12, 142.
- [5] Fang, Y.; Li, Q.; Zhang, R.X. A machine learning based network anomaly traffic analysis system. *Computer Knowledge and Technology*, 2020, 16(04), 24-25.



© 2023 by the author(s); licensee International Technology and Science Publications (ITS), this work for open access publication is under the Creative Commons Attribution International License (CC BY 4.0). (<http://creativecommons.org/licenses/by/4.0/>)