

Real-Time Popularity Prediction on Instagram

Deming Chu¹, Zhitao Shen², Yu Zhang³, and Shiyu Yang⁴(✉)

¹ East China Normal University, Shanghai, China
ned_chu@qq.com

² Ant Financial Services Group, Shanghai, China
shenzt@gmail.com

³ eBay Inc. Shanghai, Shanghai, China
webmaster@joe1cafe.com

⁴ The University of New South Wales, Sydney, Australia
yangs@cse.unsw.edu.au

Abstract. Social network services have become a part of modern daily life. Despite explosive growth of social media, people only pay attention to a small fraction of them. Therefore, predicting the popularity of a post in social network becomes an important service and can benefit a series of important applications, such as advertisement delivery, load balancing and personalized recommendation etc. In this demonstration, we develop a real-time popularity prediction system based on user feedback e.g. count of likes. In the proposed system, we develop effective algorithms which utilize the temporal growth of user feedbacks to predict the popularity in real-time manner. Moreover, the system is easy to be adapted for a variety of social network platforms. Using datasets collected from Instagram, we show that the proposed system can perform effective prediction on popularity at early stage of post.

Keywords: Real-Time, Predicting Popularity, Social Network

1 Introduction

With the popularity of mobile devices and lower bandwidth cost, people are more and more connected to each other through the Internet. People not only browse but also share and produce web contents, which leads to flood of information. At the same time, a small fraction of contents attract most of attention from public and bring most of flow out. The identification of potential popular content can help grasp pulse of flow. As a result, the service providers can make more profits out, advertisers can maximize their revenues through better advertisement placement and users can focus on most relevant information. Popularity prediction problem is clearly defined in [3]. And the popularity is usually characterized by number of user feedbacks e.g. count of likes. We also follow this convention in our demonstration. In most of the social networks, especially those focus on short contents e.g. Twitter and Instagram, a post will become popular in a short time (usually less than 24 hours)[4]. Therefore, how to predict the popularity of a post just in a short time after it has been posted becomes increasingly important.

Challenges Due to the diversity and high update frequency property of a post in social network, it is difficult to effectively predict the popularity of a post and it is even harder when considering the real-time requirement. The main challenges are in two folds: Firstly, the popularity of a post is usually affected by many factors and most of them are either difficult to measure or changing frequently. Secondly, most posts loses attention from public days after publication, which means meaningful prediction must be made within several hours or even shorter.

In this demonstration, we propose a real-time prediction system for social network content. The system consists of two main components: back-end and front-end. In the back-end, crawlers continuously crawl target pages from social network platform and several regression model based algorithms are implemented to support prediction task. In terms of front-end, we build user interface upon Django⁵ to visualize recent posts that are expected to be popular and individual prediction on single post as user input. The proposed system is evaluated by datasets which are crawled from Instagram⁶. Our main contributions are summarized as follows:

- A popularity prediction system with novel and effective prediction algorithms which can perform effective real-time prediction on popularity of social network content at early stage.
- A web interface that can present recent posts that are expected to be popular and individual prediction on a post in a user-friendly way.

2 System

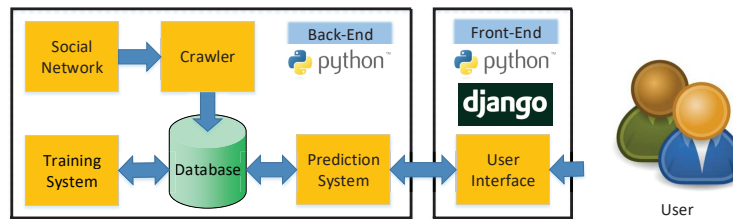


Fig. 1. System Framework

Figure 1 demonstrates framework of our system which consists of two part: back-end part and front-end part. Most of work falls in the back-end part.

2.1 Back-End

Crawler Crawler is built upon Scrapy⁷ which is a popular crawler framework. For a given social platform, homepage of selected bloggers are used as target

⁵ <https://www.djangoproject.com/>

⁶ <https://www.instagram.com>

⁷ <https://scrapy.org/>

pages in advance. In order to predict popularity in real time, the crawler is designed to crawl target pages continuously. The gap between two queries on a certain blogger is no more than tens of minutes to gain sufficient data for prediction. Moreover, we save original data on first visit to a post and ignore data that doesn't change later on. As a result, we collect a set of original data of post and a set of update data of post.

Training and Prediction System There are two important time in prediction problem: indication time t_i and reference time t_r . Prediction algorithm is running at t_i to predict popularity at t_r . In this demonstration, we take $t_r = 24h$, which means the system predict popularity 24 hours after publication. The reference time is chosen based on definition of *effective shelf-life 90%*: time passed between its first visit and the time at which it has received a fraction 90% of the visits it will ever receive.[1]

In this demonstration, we characterize popularity through number of user feedbacks, more specifically, count of likes. Regression-based model is built individually for each blogger upon one's past posts, using data point at t_i and data point at t_r . As popularity of post has strong association with its author, prediction can be well performed.

We propose three prediction algorithms, namely *Normalized Linear Regression*, *Normalized Linear Regression in Log Scale* and *Regression on Coefficient*. We also implement two algorithms as competitors, namely *Linear Regression* and *Linear Regression in Log Scale* are presented in [2]. Coefficient of each model are trained beforehand in training system, so that prediction system can handle queries in real-time. We introduce the idea of each algorithm as following:

- *Linear Regression*. Collect all history posts of a blogger and build linear regression model on popularity at t_i and popularity at t_r .
- *Linear Regression in Log Scale*. This method is almost the same as linear regression, except that linear regression model is built in log scale.
- *Normalized Linear Regression*. Popularity at t_i is normalized based on publication time of post and estimated number of blogger's online followers.
- *Normalized Linear Regression in Log Scale*. Combination of linear regression in log scale and normalized linear regression.
- *Regression on Coefficient*. Fit the growth curve of popularity with power function kx^r where x is hours from publication. Using k and log of popularity at t_r to build linear regression model.

2.2 Front-End

Two scenarios are constructed in front-end part of system to present popularity prediction result in a friendly way. In the first scenario, top 10 posts that are most likely to be popular are collected from posts known to public in the last 24 hours. In the second scenario, individual prediction on a certain post is presented in line chart.

3 Demonstration

3.1 Dataset

The proposed system are trained and evaluated using dataset which is collected from Instagram. Instagram is world's most popular photo/video sharing

social network platform, where users could upload photographs and short videos, follow other users' feeds and share their feeling. Target bloggers are selected from two sources, 100 of them are from a top Instagram accounts list⁸, the rest are randomly picked. In total, the dataset contains about 500 target bloggers, 100,000 posts and 100,000,000 updates. For a given post, gap between two updates is about several minutes.

3.2 User Interface




POST TIME	BLOGGER	BLOGGER NAME	EST POPULARITY	CONTENT
2017-02-02 02:39:00	beyonce	Beyoncé	9884967	 We would like to share our love and happiness. We have been blessed two times over. We are incredibly grateful that our family will be growing by two, and we thank you for your well wishes. - The Carters
2017-05-02 11:32:12	selenagomez	Selena Gomez	7215151	
2017-06-06 08:34:09	selenagomez	Selena Gomez	6174520	date night
2017-06-06 23:45:25	cristiano	Cristiano Ronaldo	4093231	 Proud of you. Congrats to win the copa 2017 and pichichi!

Fig. 2. Top 10 Recent Post

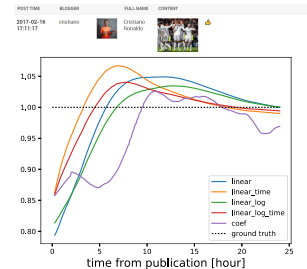


Fig. 3. Historical Prediction

Figure 2 shows the first scenario, where top 10 recent posts that are most likely to be popular are presented. Details of the post are also shown in this scenario, including post time, details of blogger, estimated popularity and content of the post. Figure 3 shows the second scenario for individual post prediction. Users can choose blogger and post, then the historical prediction results of the popularity in 24 hours is presented. In this line chart, x axis means time from publication. Meanwhile, y axis means popularity prediction result at x divided by the real popularity at 24 hours after publication, in other words, relative error to ground-truth. The closer y to 1, the better our prediction algorithm is.

References

1. Castillo, C., El-Haddad, M., Pfeffer, J., Stempeck, M.: Characterizing the life cycle of online news stories using social media reactions. In: Proceedings of the 17th ACM conference on Computer supported cooperative work & social computing. pp. 211–223. ACM (2014)
2. Szabo, G., Huberman, B.A.: Predicting the popularity of online content. Communications of the ACM 53(8), 80–88 (2010)
3. Tatar, A., de Amorim, M.D., Fdida, S., Antoniadis, P.: A survey on predicting the popularity of web content. Journal of Internet Services and Applications 5(1), 8 (2014)
4. Zaman, T., Fox, E.B., Bradlow, E.T., et al.: A bayesian approach for predicting the popularity of tweets. The Annals of Applied Statistics 8(3), 1583–1611 (2014)

⁸ <http://zymanga.com/millionplus/>