Predicting Peer Offline Probability in BitTorrent Using Nonlinear Regression

Dongdong Nie[1], Qinyong Ma[2], Lizhuang Ma[1], and Wuzheng Tan[1]

[1] Computer Science & Engineering Department, Shanghai Jiao Tong University, Shanghai
[2] Computer Science & Engineering Department, Zhejiang University, Hangzhou
niedd.mail@gmail.com, mqyray@163.com, ma-lz@cs.sjtu.edu.cn, tanwuzheng@sjtu.edu.cn

Abstract. BitTorrent is a popular and scalable P2P content distribution tool. This study attempts to analyze the factors that affect the offline probability of BitTorrent peer, and express the probability using these factors. We first collect large data set of BitTorrent peers' activities. Then we use nonlinear least-squares regression to determine the probability distribution function for each of the three factors (download percent, download speed, and local time) and the joint probability distribution function of the three factors, and use another large data set to verify the prediction results.

Keywords: BitTorrent; Offline; Probability Distribution; Regression.

1 Introduction

BitTorrent [1] is a popular and scalable P2P content distribution tool. In the recent years, many research efforts [5, 6, 7, 8] have focused on P2P systems. Some works on the measurements of P2P systems have studied the distributions of peer's session time. Saroiu, et al. [9] present the session durations of Napster and Gnutella. Chu, et al. [10] present the first study of the popularity of files stored and transferred among peers in Napster and Gnutella over month-long periods, and propose that the distribution of session lengths follows a log-quadratic function. Stutzbach, et al. [2, 11] show that peer uptime follows a power-law distribution rather than the commonly assumed Poisson distribution. Besides session time, some studies [3, 9, 10] find that peer's availability varies with the hour of the day. We are inspired by this finding to express offline probability distribution by local time.

If peer's offline probability can be expressed clearly, it would be helpful for BitTorrent to improve its downloading strategy or to implement some interesting functions. In this paper, we try to analyze the factors that affect the offline probability of BitTorrent peer, and describe the probability using these factors. We find that besides uptime, there are still some other important factors: download percent, download speed, and local time. We regard each factor as a random variable. From the large data set of peer activities obtained in a BitTorrent network, we get these variables' values when each peer enters offline state. Based on these values, we get the

L. Ma, R. Nakatsu, and M. Rauterberg (Eds.): ICEC 2007, LNCS 4740, pp. 339–344, 2007.

distribution data for each variable, and determine the expressions to represent these distributions by performing a nonlinear least-squares regression (NLSR) on each distribution data. Then we try to use three main factors (download percent, download speed, and local time) to express the offline probability together, and use another set of BitTorrent log data to verify the expressions.

2 Data Set

The data set consists of two parts. The first part, which is called the training set, contains the data that is used for regression analysis. We acquire the training set with the same method as used in [3]. We use an application to acquire the data in four steps. First, it monitors, gathers, and parses the HTML pages of the bt.5qzone.net, a famous BitTorrent website in CERNET (China Education and Research Network), and downloads some new .torrent files randomly. Second, parses the downloaded .torrent file to get corresponding tracker URLs. Third, links to each tracker to get the list of all the peers downloading the files. Fourth, link to each non-firewalled downloading peer, and begin to record its session data. A number of 219063 valid session data was acquired during the period from February 2006 to April 2006. We use this data set to determine the probability distribution function (PDF) of each variable and the joint probability distribution function (JPDF) of three variables.

The second part, which is called the test set, is the BitTorrent tracker log described in [4]. The authors obtained a RedHat torrent's tracker log on a five months long period. The log contains statistics for about 180,000 clients, and it clearly exhibits an initial flash-crowd period with more than 50,000 clients initiating a download in the first five days. We use the filtered 144,196 valid session data as the test set to verify the determined distribution functions.

In the training set and the test set, we only care about the offline data, which is the data related to peer's offline state. The offline data includes the values of the following variables when each peer enters offline state: uptime (equals to session time), download percent, download speed, and local time.

The uptime values can be obtained from the data set directly. A peer's session time is a combination of the time it spends to download the file (the download time) and the additional time it keeps running after the download is complete (the lingering time).

We let a peer's download percent value could be larger than 100%. Its value would exceed 100% if the lingering time is larger than zero. Let T_d be the download time. Let T_l be the lingering time, and set $T_l = 0$ if the download has not be completed. Let S_d be the size of the data downloaded by the peer in T_d. Let S_f be the total size of the files. The download percent P_d is calculated as: $P_d = S_d / S_f + T_l / T_d$.

A peer's download speed value is calculated from the last two known sizes of the data downloaded by the peer. The calculated download speed will be zero if the peer has downloaded all the files.

To calculate a peer's local time, we first get the its time zone from its IP address, then converts the recorded offline time to its time zone's corresponding local time.

For the offline data obtained above, we define the following intervals as the analysis domain: uptime is limited to the interval: [0, 1005) minutes, download percent is limited to the interval: [0, 201) percent, and download speed is limit to the interval:

[0, 301500) Bps. The domain covers about 85% of the total offline data. The skipped data covers a large span, while its proportion is small. For instance, the proportion of the sessions with uptimes longer than 1005 minutes is less than 4% of total sessions, but their values covers a large span of time, some peers' uptimes even exceed 100000 minutes. Since peer uptimes follow a power-law distribution, it has some sessions with much longer durations, as well as has a much larger fraction of sessions with short durations. The offline probability of the skipped sessions is very low, so that the limited domain let us concentrate on the sessions with higher offline probabilities, while it has little influence on the sessions with low offline probabilities. For instance, when we compare two peers' offline probabilities, 30% and 10% have much difference, but the difference between 0.01% and 0.03% is little.

The data types of uptime, download percent, and download speed are all float. To facilitate calculation, we divide each variable's domain interval into 100 equal subintervals by rounding to the nearest integer. For instance, uptime's domain interval [0, 1005) is divided into 100 equal subintervals, then the value 3 represents the interval [25, 35). The data type of local time is integer, and the time unit is hour.

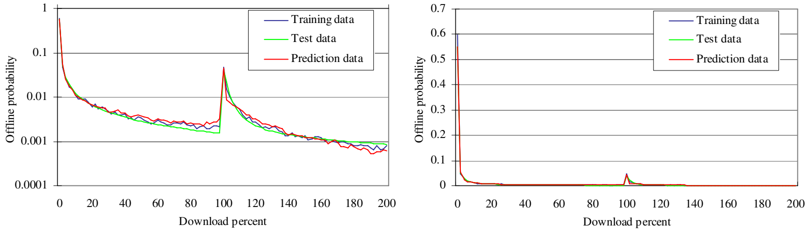
3 Determine the Distributions of the Variables

Here we first discuss the limitation if we only use the uptime to express peer's offline probability. Consistent with [2], we confirm that the uptime follows a power-law distribution. But we find that offline probability is correlated with other factors. Peer's offline probability varies significantly with its download speed when uptime is fixed. So the accuracy is often limited if we only use uptime to express peer's offline probability. The phenomenon can also be found by the offline probability distribution of download percent described blow. A peer's offline probability would increase suddenly when it just has completed the downloading.

We first get the offline probability distribution of download percent from training set, then attempt to determine its PDF by NLSR. We attempt to use some curve equations to perform the NLSR, and select the equation with best fitting for larger value as its PDF. Let x_p be the download percent. The offline PDF of download percent can be expressed as:

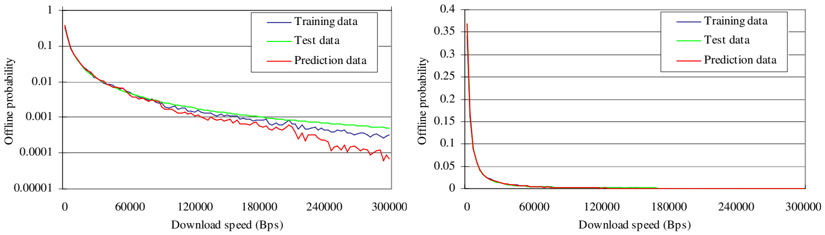
$$
\begin{aligned}
y_p &= \frac{1}{17.42 \times x_p^{0.93218} + 1.8105} \qquad (x_p < 50) \\
y_p &= \frac{1}{17.42 \times x_p^{0.93218} + 1.8105} + \frac{1}{22.498 \times (x_p - 50)^{1.6413} + 25.6} \qquad (x_p \geq 50)
\end{aligned}
\tag{1}
$$

We use the equation to calculate the prediction value, and verify it with the test set. Fig. 1 illustrates the result, in which (a), (b) show the same data; the only difference is that the *y*-axis in (a) is in log scale. As shown in the figure, a peer's offline probability would increase suddenly when it just has completed the downloading. As can be observed better in (b), the obtained PDF has higher accuracy for larger offline probabilities. It is because of our preference for larger offline probabilities.



(a) (b)

Fig. 1. Download percent's offline probability: (a) Y-axis is in log scale; (b) Y-axis is in linear scale



(a) (b)

Fig. 2. Download speed's offline probability: (a) Y-axis is in log scale; (b) Y-axis is in linear scale

Using the above method, we obtain the PDF of download speed:

$$y_s = \frac{1}{3.1308 \times x_s^{1.3982} + 2.7078} \tag{2}$$

where x_s is the download speed. The result is illustrated in Fig. 2.

Setting x_t as the local time, we obtain the PDF of local time similarly:

$$\begin{aligned} y_t = {} & 0.043043 - 0.008794 \times \sin(0.23499 \times x_t + 6.4713) + \\ & 0.0085267 \times \sin(0.3914 \times x_t + 2.4461) \end{aligned} \tag{3}$$

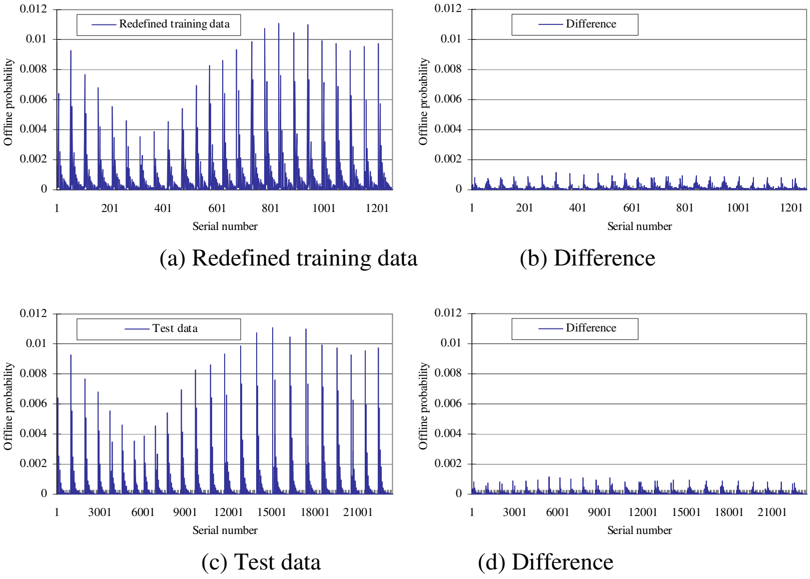
Now we try to use the three variables (download percent, download speed, and local time) to express the offline probability together. The value of the offline probability expressed by three variables together is relative small. Since we prefer larger offline probability, the analysis domain is redefined to the following intervals: download percent is limited to the intervals: [0, 6] & [100, 102] percent, and download speed is limit to the interval: [0, 30150} Bps, while local time is unlimited. Then we perform a multiple least-squares nonlinear regression on the data by the Gauss-Newton method. We obtain the following JPDF:

$$
\begin{aligned}
y_p &= \frac{1}{2.206 \times {x_p}^{1.1437} + 0.11516} \quad (x_p < 50) \\
y_p &= \frac{1}{2.206 \times {x_p}^{1.1437} + 0.11516} + \\
&\quad 0.037397 + 0.59592 \times \exp(0.60185 - 1.3989 * (x_p - 50)) \quad (x_p \geq 50)
\end{aligned} \tag{4}
$$

$$
y_s = \frac{1}{1.2357 \times {x_s}^{1.816} + 2.2582} \tag{5}
$$

$$
\begin{aligned}
y_t = {} & (0.0021369 + 0.00083506 \times \sin(0.27575 \times x_t + 2.9867) + \\
& 0.00026875 \times \sin(0.5096 \times x_t + 1.3564))
\end{aligned} \tag{6}
$$

$$
y = y_p \times y_s \times y_t \tag{7}
$$



(a) Redefined training data (b) Difference

(c) Test data (d) Difference

Fig. 3. Triple set's offline probability

The result is illustrated in Fig. 3, in which *X*-axis is the serial number of the triple (download percent, download speed, local time) set; (a) illustrates the distribution of the redefined training data; (b) illustrates the difference between the prediction data and the redefined training data; (c) illustrates the distribution of the test data; (d) illustrates the difference between the prediction data and the test data.

4 Conclusions

This study attempts to predict peer's offline probability in BitTorrent. We first collect large data set of BitTorrent peers' activities. Then we use nonlinear least-squares regression to obtain the probability distribution functions and the joint probability distribution functions for three variables, and compare the prediction values with another large data set. The result shows that the PDFs are relative accurate for larger offline probabilities.

We will collect other popular P2P systems' activity data in the future, and use regression to analyze the offline probabilities of other P2P systems. We will also seek other ways to predict peer's offline probability.

Acknowledgement. This work is supported by National Science Foundation of China No.60403044, No. 60573147 and partly funded by the National Key Basic Research and Development Plan of China ("973"Projects)(2006CB303105).

References

1. BitTorrent: http://www.bittorrent.com/
2. Stutzbach, D., Rejaie, R.: Characterizing Churn in Peer-to-Peer Networks. Technical report CIS-TR-05-03 (2005)
3. Pouwelse, J., Garbacki, P., Epema, D., Sips, H.: The BitTorrent P2P File-Sharing System: Measurements And Analysis. In: Castro, M., van Renesse, R. (eds.) IPTPS 2005. LNCS, vol. 3640, pp. 205–216. Springer, Heidelberg (2005)
4. Izal, M., Urvoy-Keller, G., Biersack, E., Felber, P., Al Hamra, A., Garces-Erice, L.: Dissecting BitTorrent: Five Months in a Torrent's Lifetime. In: Barakat, C., Pratt, I. (eds.) PAM 2004. LNCS, vol. 3015, pp. 1–11. Springer, Heidelberg (2004)
5. Stoica, I., Morris, R., Karger, D., Frans Kaashoek, M., Balakrishnan, H.: Chord: A Scalable Peer-To-Peer Lookup Service for Internet Applications. In: ACM SIGCOMM 2001, San Deigo, CA, pp. 149–160 (2001)
6. Ratnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S.: A Scalable Content-Addressable Network. In: ACM SIGCOMM 2001, San Diego, CA, pp. 161–172 (2001)
7. Mischke, J., Stiller, B.: Rich and Scalable Peer-to-Peer Search with SHARK. In: AMS'03, pp. 112–122. IEEE Press, Washington (2003)
8. Zhang, C., Krishnamurthy, A., Wang, R.: Brushwood: Distributed Trees in Peer-to-Peer Systems. In: Castro, M., van Renesse, R. (eds.) IPTPS 2005. LNCS, vol. 3640, pp. 47–57. Springer, Heidelberg (2005)
9. Saroiu, S., Gummadi, P., Gribble, S.: A Measurement Study of Peer-to-Peer File Sharing Systems. In: Kienzle, M.G. (ed.) Multimedia Computing and Networking, pp. 156–170. SPIE, San Jose (2002)
10. Chu, J., Labonte, K., Levine, B.: Availability and Locality Measurements of Peer-to-Peer File Systems. In: ITCom: Scalability and Traffic Control in IP Networks, Boston, pp. 310–321 (2002)
11. Stutzbach, D., Rejaie, R.: Towards a Better Understanding of Churn in Peer-to-Peer Networks. Technical Report CIS-TR-04-06 (2004)