Graph Based Classification of Content and Users in BitTorrent

Konstantin Avrachenkov
INRIA Sophia Antipolis, France
K.Avrachenkov@sophia.inria.fr

Paulo Gonçalves
INRIA Rhone-Alpes, France
paulo.goncalves@inria.fr

Arnaud Legout
INRIA Sophia Antipolis, France
arnaud.legout@inria.fr

Marina Sokol
INRIA Sophia Antipolis, France
marina.sokol@inria.sophia.fr

1 Introduction

P2P downloads still represent a large portion of today's Internet traffic. More than 100 million users operate BitTorrent and generate more than 30% of the total Internet traffic [7]. Recently, a significant research effort has been done to develop tools for automatic classification of Internet traffic by application [9, 8, 11]. The purpose of the present work is to provide a framework for subclassification of P2P traffic generated by the BitTorrent protocol. Unlike previous works [9, 8, 11], we cannot rely on packet level characteristics and on the standard supervised machine learning methods. The application of the standard supervised machine learning methods in [9, 8, 11] is based on the availability of a large set of parameters (packet size, packet interarrival time, etc.). Since P2P transfers are based on the same BitTorrent protocol we cannot use this set of parameters to classify P2P content and users. Instead we can make use of the bipartite user-content graph. This is a graph formed by two sets of nodes: the set of users (peers) and the set of contents (downloaded files). From this basic bipartite graph we also construct the user graph, where two users are connected if they download the same content, and the content graph, where two files are connected if they are both downloaded by at least one same user. The general intuition is that the users with similar interests download similar contents. This intuition can be rigorously formalized with the help of graph based semi-supervised learning approach [13].

The main idea of the graph based semi-supervised learning approach is to use the instance smoothness over the graph. Namely, if one data point has many neighbors from some class then it is very likely that this data point belongs to that class. In particular, we have chosen to work with PageRank based semi-supervised learning method [3, 4, 12]. It has been demonstrated in [4] that this method has implementations with quasi-linear complexity and produces robust results with respect to the method's parameters. We would like to emphasize that the graph based semi-supervised learning methods allow one to perform high precision classification using only a very small amount of the labelled data.

Using methodology developed in [7] we were able to use the snapshots of BitTorrent downloads from the whole Internet. Even a snapshot corresponding to half an hour duration represent a huge among of data (more than one million peers and more than 200 thousand content files). Without efficient preprocessing technique, which will be explained in Section 3, we were even not able to operate with the user graph constructed from a single snapshot. The content graph is smaller and we were able to construct an aggregated content graph from several snapshots corresponding to the week-long observation.

We have three goals in the present work. The main goal is to provide a robust graph based semi-supervised learning approach for content and user classification of BitTorrent P2P transfers. The second goal is to demonstrate that the PageRank based semi-supervised learning method, thanks to its quasi-linear complexity, can deal with classification of very large datasets. Some datasets used in the present paper is several orders of magnitude larger than datasets typically used in the literature on graph based semi-supervised learning. The third goal is to test the impact of the choice of the labelled nodes on classification result. In particular, we test the following three options for the choice of the labelled points: randomly chosen labelled points, labelled points with large PageRank values and labelled points with large degrees. We demonstrate that in the context of P2P classification the choice of labeled points with large PageRank values gives good results in the majority of classification tasks.

The work is organized as follows: In the next Section 2 we describe the PageRank based semi-supervised learning method. Then, in Section 3 we give detail description of our datasets. In Section 4 we perform topic based and language based classifications of the whole collection of the P2P traffic based on the content graph and user graph, respectively, and provide conclusions.

2 PageRank based classification

Let us present some basic facts about PageRank based semi-supervised learning method. An interested reader can find more theoretical results in [4] and in related works [3, 12].

Suppose we need to classify N data points into K classes and P data points are labelled. In particular, this means that for a labelled point $i = 1, ..., P$ the function $k(i) \in 1, ..., K$ is defined. Graph based semi-supervised learning approach uses a weighted graph connecting data points. The weight matrix, or similarity matrix, is denoted by W. Here we assume that the weight matrix W is symmetric. Each element $w_{i,j}$ represents a degree of similarity between data points i and j. Denote by D a diagonal matrix with its (i, i)-element equals to the sum of the i-th row of matrix W: $d_{i,i} = \sum_{j=1}^{N} w_{i,j}$.Define $N \times K$ matrix Y as

$$Y_{ik} = \begin{cases} 1, & \text{if } X_i \text{ is labeled as } k(i) = k, \\ 0, & \text{otherwise.} \end{cases}$$

We refer to each column $Y_{.k}$ of matrix Y as labeling function. Also define $N \times K$ matrix F and call its columns $F_{.k}$ classification functions. A general idea of the graph-based semi-supervised learning is to find classification functions so that on the one hand they will be close to the corresponding labeling function and on the other hand they will change smoothly over the graph associated with the similarity matrix. This general idea can be expressed by means of the optimization formulation

$$\underset{F}{\text{argmin}} \sum_{i=1}^{N} \sum_{j=1}^{N} w_{ij} \| \frac{F_{i.}}{d_{ii}} - \frac{F_{j.}}{d_{jj}} \|^2 + \mu \sum_{i=1}^{N} \frac{1}{d_{ii}} \| F_{i.} - Y_{i.} \|^2 \tag{1}$$

where μ is a regularization parameter. In fact, the parameter μ represents a trade-off between the closeness of the classification function to the labeling function and its smoothness.

Proposition 1 *The classification functions for the PageRank based semi-supervised learning are given by*

$$F_{.k} = \frac{\mu}{2+\mu} \left(I - \frac{2}{2+\mu} W D^{-1} \right)^{-1} Y_{.k}, \tag{2}$$

for $k = 1, ..., K$.

Let us now explain why the following framework corresponds to the PageRank based clustering method. Denote $\alpha = 2/(2 + \mu)$ and write $F_{.k}$ in a transposed form

$$F_{.k}^T = (1 - \alpha) Y_{.k}^T (I - \alpha D^{-1} W)^{-1}.$$

If the labeling functions are normalized, this is exactly an explicit expression for PageRank [10]. This expression was used in [3] but no optimization framework was provided.

Note that $D^{-1}W$ represents the transition probability matrix for the random walk on the similarity graph. Then, the (i, j)-th element of the matrix $(I - \alpha D^{-1} W)^{-1}$ gives the expected number of visits to node j starting from node i until the random walk restarts with probability $1 - \alpha$. This observation provides the following probabilistic interpretation for the PageRank based method. In the PageRank based method with normalized labeling functions, F_{ik} gives up to a multiplicative constant the expected number of visits to node i, if the random walk starts from a uniform distribution over the labeled nodes of class k.

The choice of the labelled points can potentially have a significant influence on classification results. Therefore, in the present work we study this influence. Specifically, we consider the following options for the choice of labelled points:

1. randomly chosen labelled points, that is, in each class we take several samples of random labelled points;
2. labelled points are chosen among points with large values of Standard PageRank; (with large values of π_i, $i = 1, ..., N$, where π_i are elements of a solution of the equation $\pi = \pi \alpha D^{-1} W + (1-\alpha)/N \underline{1}^T$);
3. labelled points are chosen among points with large degree (with large values of $d_{i,i}$).

3 Datasets and method implementation description

We have several snapshots of the Torrents collected from the whole Internet using methodology described in [7]. Each snapshot contains half an hour of P2P transfers. In total, we have about one week of observations. We have also an aggregate representing the transfers observed during the whole week. To test the effect of NATs, to save memory and to reduce information noise, the following filtering has been applied which we denote by $g(X, Y)$: we filter out all IP addresses with more than or equal to X ports (X = 0 means no filtering), and we filter out all contents with less than or equal to Y IP addresses seen downloading the content (Y = 0 means no filtering). Two users with the same IP addresses but with different ports could be the same user. So the filtering by ports helps us to reduce the influence of counting the same user as different ones. The second filter by IP address helps to remove unpopular contents which were downloaded by less than or equals to Y different addresses. We use the whole aggregate to create the content graph. Some files are tagged with information about name, language, topic, login of the person who inserted these files. Those tags correspond to the classification made by popular torrent sites like ThePirateBay [7]. If two files are downloaded by the same user, we create an edge between these two files. The weight of the edge shows how many users downloaded these two files.

Table 1: The content graphs after preprocessing.

Graph	# nodes	# edges
g(2,10)	200 413	50 726 946
g(0,10)	200 487	174 086 752
g(2,0)	624 552	92 399 318

We start with the smallest aggregated dataset $g(2, 10)$ which contain information with small noise. To evaluate the impact of the noise with respect to user identification we have also made experiments with datasets $g(0, 10)$ and $g(2, 0)$. The graph for $g(2, 0)$ dataset contains three times more nodes and two times more edges than the dataset $g(2, 10)$. The graph for $g(0, 10)$ dataset contains three times more edges than the dataset $g(2, 10)$.

Let us now describe how we construct the user graph. The user graph is constructed with the help of HADOOP realization of MapReduce technology [1] from the basic user-content bipartite graph from a single half an hour snapshot. The aggregated user graph is too large to work with.

The snapshot contains information on which content was downloaded by whom. In the user graph an edge with the weight M signifies that two users download M same files. The user graph has 3 228 410 nodes and 3 436 442 577 edges. The number of edges with weight one is equal to 3 309 965 972. Also we have noticed that some users downloaded much more files than a normal user would do. One user who has downloaded 655 727 files for sure is a robot. Thus, we have decided remove all edges with weight one and the user-robot. The modified user graph has 1 126 670 nodes and 124 753 790 edges. This filtering significantly reduces required computing and memory resources. Without this filtering even the PageRank based method with quasi-linear complexity cannot be applied on a standard desktop computer. In fact, by doing this filtering we also remove some

Table 2: The quantity of language base line expert classifications.

Language	# content	# users
English	36 465	57 632
Spanish	2 481	2 856
French	1 824	2 021
Italian	2 450	3 694
Japanese	720	416
Unknown	156 473	1 060 051

Table 3: The quantity of topic base line expert classifications.

Topic	# content	# users
Audio Music	23 639	13 950
Video Movies	20 686	43 492
TV shows	12 087	27 260
Porn movies	8 376	7 082
App. Windows	4 831	2 874
Games PC	4 527	8 707
Books Ebooks	1 185	281
Unknown	125 082	1 023 024

information noise. If two users download only one common item it could be by pure chance, if they both download more than two same files - it is more likely that they share same interests.

We classify contents and users by both language and topics. The considered languages and topics are given in Tables 2 and 3.

Our base line expert classification is based on P2P content tags if they are available. For instance, in the case of classification by language we consider that the content is in English if it has only tag "English". And we consider a user to be an English language user, if he or she downloads only English language content.

We have implemented PageRank based classification method in the WebGraph framework [6]. The WebGraph framework has a very efficient graph compression technique which allows us to work with very large graphs.

4 Results and conclusions

Using PageRank based classification method, we have performed four classification experiments. We have used the aggregated graph of content $g(2, 10)$ to classify the content into 5 classes according to the languages (given in Table 2) and into 7 classes according to the content topics (given in Table 3). The classification of the aggregated content graph has taken approximately 15 minutes on a 64-bit computer with Intel-Core7i processor and 6GB RAM. The results of the classification evaluated in terms of accuracy are presented in Tables 4 and 5. Then, we have performed the classification of users also into 5 classes of the languages and into 7 classes of the content preferred by users (see Tables 6 and 7). It has taken about 20 minutes on the same computer. However, the preprocessing of a single snapshop of the user graph was much more demanding than the preprocessing of the aggregated content graph. Our main conclusion is that the PageRank based classification method scales remarkably well with large volumes of data. Then, our second important observation is that by using a very little amount of information, we are able to classify the content and users with high accuracy. For instance, in the dataset of 1 126 670 users, using only 50 labelled points for each language (which is only 0.02% of the whole data), we are able to classify the users according to their preferred language with 88% accuracy (see Table 6).

In all four classification experiment, we have tried three different options for the choice of the labelled points. We have chosen the labelled points: (a) with largest standard PageRank values; (b) with largest degree; and (c) randomly. When evaluating the performance with the randomly chosen labelled points we have averaged the accuracy over 10 random samples (because of the size of the data, making more than 10 samples for each of many experimental setups was very time demanding) and we have also reported the worst (rand min column) and the best (rand max column) accuracy. With respect to the choice of the labelled points, our conclusion is that in the majority of cases the labelled points with large values of the standard PageRank are the best picks (see topPR columns). In the case of classification with the aggregated content graph, the labelled points with large degrees give results comparable with the results obtained with the labelled points chosen according to PageRank. However, it was interesting to observe that in the case of the classification of users, the classification based on the labelled points with large degrees does not perform well at all. Our explanation is that in that dataset the nodes with very large degrees are not representative. There is an independent confirmation of this idea given in [5].

Finally, we have observed that the classification using $g(2, 10)$ filtering is one or two percent better in terms of accuracy than the classification using $g(0, 10)$ filtering. Thus, by doing the filtering we not only reduce the amount of data required for processing, but also we reduce the information noise.

Acknowledgement

The work has been supported by the joint INRIA Alcatel-Lucent Laboratory.

Appendix 1: Tables

Table 4: Accuracy of the classifications for the $g(2, 10)$ dataset by languages.

# labeled points	topPR	topDegree	rand (10Exp)	rand min	rand max
5	0.579	0.573	0.51	0.44	0.578
50	0.663	0.647	0.634	0.614	0.649
500	0.688	0.676	0.658	0.653	0.663

Table 5: Accuracy of the classifications for the $g(2, 10)$ dataset by topics

# labeled points	topPR	topDegree	rand(10Exp)	rand min	rand max
5	0.504	0.51	0.48	0.36	0.546
50	0.6344	0.6276	0.6278	0.604	0.645
500	0.7279	0.7182	0.6562	0.6525	0.6595

Table 6: Accuracy of the classifications for the user dataset by languages

# labeled points	topPR	topDegree	rand (10Exp)	rand min	rand max
5	0.788	0.765	0.732	0.613	0.817
50	0.88	0.78	0.834	0.82	0.85
500	0.853	0.535	0.901	0.896	0.907

Table 7: Accuracy of the classifications for the user dataset by topics.

# labeled points	topPR	topDegree	rand(10Exp)	rand min	rand max
5	0.683	0.399	0.631	0.563	0.678
50	0.752	0.477	0.767	0.752	0.777
500	0.789	0.52	0.86	0.858	0.865

References

[1] Hadoop mapreduce software framework, http://hadoop.apache.org/mapreduce/. 2011.

[2] Wikipedia article “bittorrent (protocol)”, http://en.wikipedia.org/wiki/bittorrent_(protocol). 2011.

[3] Konstantin Avrachenkov, Vladimir Dobrynin, Danil Nemirovsky, Son Kim Pham, and Elena Smirnova. Pagerank based clustering of hypertext document collections. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR ’08, pages 873–874. ACM, 2008.

[4] Konstantin Avrachenkov, Paulo Gonçalves, Alexey Mishenin, and Marina Sokol. Generalized optimization framework for graph-based semi-supervised learning. *Submitted for publication, available upon request*, 2011.

[5] Brian Ball, Brian Karrer, and M. E. J. Newman. Efficient and principled method for detecting communities in networks. *Phys. Rev. E*, 84:036103, Sep 2011.

[6] Paolo Boldi and Sebastiano Vigna. The webgraph framework i: compression techniques. In *Proceedings of the 13th international conference on World Wide Web*, WWW '04, pages 595–602, New York, NY, USA, 2004. ACM.

[7] Stevens Le Blond, Arnaud Legout, Fabrice Lefessant, Walid Dabbous, and Mohamed Ali Kaafar. Spying the world from your laptop: identifying and profiling content providers and big downloaders in bittorrent. In *Proceedings of the 3rd USENIX conference on Large-scale exploits and emergent threats: botnets, spyware, worms, and more*, LEET'10, pages 4–4, Berkeley, CA, USA, 2010. USENIX Association.

[8] Wei Li, Marco Canini, Andrew W. Moore, and Raffaele Bolla. Efficient application identification and the temporal and spatial stability of classification schema. *Comput. Netw.*, 53:790–809, April 2009.

[9] Wei Li and Andrew W. Moore. A machine learning approach for efficient traffic classification. In *Proceedings of the 2007 15th International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, pages 310–317, Washington, DC, USA, 2007. IEEE Computer Society.

[10] Cleve B. Moler. *Numerical Computing with MATLAB*. 2004.

[11] Marcin Pietrzyk, Jean-Laurent Costeux, Guillaume Urvoy-Keller, and Taoufik En-Najjary. Challenging statistical classification for operational usage: the adsl case. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference*, IMC '09, pages 122–135, New York, NY, USA, 2009. ACM.

[12] Dengyong Zhou, Olivier Bousquet, Thomas Navin Lal, Jason Weston, and Bernhard Schölkopf. Learning with local and global consistency. In *Advances in Neural Information Processing Systems 16*, pages 321–328. MIT Press, 2004.

[13] Xiaojin Zhu. Semi-supervised learning literature survey, technical report 1530, department of computer sciences, university of wisconsin, madison, 2005.